**SOFTWARE INTERNATIONALIZATION: A FRAMEWORK VALIDATED AGAINST
INDUSTRY REQUIREMENTS FOR COMPUTER SCIENCE AND
SOFTWARE ENGINEERING PROGRAMS**

A thesis presented to the faculty of
California Polytechnic State University
at
San Luis Obispo, California

In partial fulfillment of the requirements for the degree
Master of Science
in
Computer Science

by
John Huan Vu

March 2010

© 2010

John Huan Vu

All Rights Reserved

COMMITTEE MEMBERSHIP

TITLE: **Software Internationalization: A Framework Validated Against Industry Requirements for Computer Science and Software Engineering Programs**

AUTHOR: **John Huan Vu**

DATE SUBMITTED: **March 1, 2010**

COMMITTEE CHAIR: **Dr. Clark S. Turner, J.D. Ph.D.**
Thesis Advisor
Software Engineering Program Coordinator
Computer Science Professor
Computer Science Department
California Polytechnic State University

COMMITTEE MEMBER: **Dr. Beth L. Chance, Ph.D.**
Statistics Advisor
Statistics Professor
Statistics Department
California Polytechnic State University

COMMITTEE MEMBER: **Dr. Ignatios E. Vakalis, Ph.D.**
Department Chair
Computer Science Professor
Computer Science Department
California Polytechnic State University

COMMITTEE MEMBER: **Dr. Franz J. Kurfess, Ph.D.**
Computer Science Professor
Computer Science Department
California Polytechnic State University

ABSTRACT

Software Internationalization: A Framework Validated Against Industry Requirements
for Computer Science and Software Engineering Programs

John Huan Vu

In 2001, the ACM and IEEE Computing Curriculum stated that it was necessary to address “the need to develop implementation models that are international in scope and could be practiced in universities around the world” [1]. With increasing connectivity through the internet, the move towards a global economy and growing use of technology places software internationalization as a more important concern for developers [2]. However, there has been a “clear shortage in terms of numbers of trained persons applying for entry-level positions” in this area [3]. Eric Brechner, Director of Microsoft Development Training, suggested five new courses to add to the computer science curriculum due to the growing “gap between what college graduates in any field are taught and what they need to know to work in industry” [4]. He concludes that “globalization and accessibility should be part of any course of introductory programming,” stating [4]:

A course on globalization and accessibility is long overdue on college campuses. It is embarrassing to take graduates from a college with a diverse student population and have to teach them how to write software for a diverse set of customers. This should be part of introductory software development. Anything less is insulting to students, their family, and the peoples of the world.

There is very little research into how the subject of software internationalization should be taught to meet the major requirements of the industry [4]. The research question of the thesis is thus, “*Is there a framework for software internationalization that has been validated against industry requirements?*” The answer is no. The framework “would promote communication between academia and industry ... that could serve as a common reference point in discussions” [3]. Since no such framework for software internationalization currently exists, one will be developed here. The contribution of this thesis includes a provisional framework to prepare graduates to internationalize software and a validation of the framework against industry requirements. The requirement of this framework is to provide a portable and standardized set of requirements for computer science and software engineering programs to teach future graduates.

KEYWORDS

Software Internationalization, Software Localization, Software Globalization, Computer Science Education, Software Engineering Education, Industry Survey, Industry Questionnaire, Industry Case Study, Statistics, Hypothesis Testing.

CURRICULUM VITAE

John Huan Vu

Master's of Science in Computer Science

Bachelor's of Science in Computer Science

Bachelor's of Science in Mathematics

California Polytechnic State University, San Luis Obispo, California

March 2010

PUBLICATIONS

Vu, John Huan, Niklas Frojd, Clay Shenkel-Therolf, & David S. Janzen (2009). Evaluating Test-Driven Development in an Industry-sponsored Capstone Project. *International Conference on Information Technology: New Generations*. IEEE.

PERSONAL ACKNOWLEDGEMENTS

God – Thank you for a *Purpose-Driven Life* after my freshmen year and the people you placed in my life to be a better person. I am dedicating this thesis *Ad Majorem Dei Gloriam Inque Hominum Salutem*.

Gregory A. Ripa – Thank you for being my best friend, my true love, and my future partner. I am thankful for God's divine intervention for our friendship, our relationship, and our future.

Michelle K. Vu – Thank you for being my mother who has raised me without a father since I was one. I am forever thankful for the love and the blessings you have given me these past years.

Sonia M. Martin – Thank you for being my second mom and the best high school teacher I ever had. I am thankful that you never gave up on me and that you will always set time aside to give advice and help.

Dr. Clark S. Turner – Thank you for being my thesis advisor who challenged me to be the best I can. Thank you for being an honest and loving person who strives to make everyone feel welcomed.

Dr. Ignatios E. Vakalis – Thank you for being the greatest Computer Science Department Chair in my years at Cal Poly. I am thankful for all your help, support, and advice.

Dr. Franz J. Kurfess – Thank you for being one of the most approachable professors I have ever met. I am thankful that you are always willing to be supportive and helpful in anything and everything.

Dr. Beth L. Chance – Thank you for never giving up on me in order to help me become a better statistician. I am thankful that you were always available for advice.

Dr. Harvey C. Greenwald – Thank you for believing in me and inspiring me to double major in mathematics. I am thankful that I have someone who always challenges me to never give up.

Dr. Philip L. Nico – Thank you for being the advisor who challenged me to stay within my major. I am thankful that you believed in my potential and I sincerely hope that I have made you proud.

Dr. Zoë J. Wood – Thank you for being my phenomenal computer graphics professor who continues to believe in me and who continues to challenge me to be someone greater.

Fr. John M. Ulrich S.M. – Thank you for changing my life when I first opened up to you. I am thankful that you formed me into a more mature, more loving, and more Christ-like person.

Fr. Kevin J. Duggan S.M. – Thank you for increasing my knowledge about our Catholic Christian faith. I am thankful for the energy and time you put into our Newman Catholic Community.

Linda M. Garcia-Inchausti – Thank you for being my mother at my home away from home. I am thankful that I am able to open up to you and receive a truthful and honest answer in reply.

Yvonne Ramos – Thank you for not only helping me in the financial aid office concerning scholarships but also being a friend with whom I can talk and hug.

José A. Millán – Thank you for being my advisor from the beginning of college who never hesitated to give the time and place to help me succeed in college.

Ryan A. Bridge – Thank you for being a great friend to me throughout the years. Thank you for documenting one of my greatest achievements in my life and giving me the best hugs in the world.

Rajani Gudlavalleti – Thank you for being the friend who always make me feel like a special person in this world. Thank you for witnessing my growth and challenging me to be a bigger and better person.

Megan J. Stallman – Thank you for being my loving friend whom I met at Newman. I am thankful that I could turn to you whenever I needed a hug, a shoulder, or support.

Tina M. Woolf – Thank you for being my loving friend throughout mathematics and statistics. Thank you for your kindness, sense of humor, patience, and being my permanent temporary girlfriend.

Kaitlin E. Hill – Thank you for being that one friend whom I can come to and know that I can come out better and stronger. I sincerely hope others will love you and appreciate you for everything that you do.

TABLE OF CONTENTS

LIST OF TABLES	X
LIST OF FIGURES	XI
CHAPTER 1. INTRODUCTION.....	1
CHAPTER 2. IS SOFTWARE INTERNATIONALIZATION IMPORTANT?	6
2.1. Is software internationalization a trivial process?	7
2.2. Is there market potential for software internationalization?	8
CHAPTER 3. WHAT IS THE PRESENT STATE OF EDUCATION FOR SOFTWARE INTERNATIONALIZATION?.....	9
3.1. Foreword	9
3.2. Educational architectural models	11
3.2.1. <i>Integration into existing courses</i>	12
3.2.2. <i>Develop an entire course</i>	12
3.2.3. <i>Study abroad for experience</i>	14
3.2.4. <i>Conclusion</i>	14
3.3. Definitions and standards	15
3.4. Development practices	16
3.5. Translation and documentation	17
3.5.1. <i>Direction</i>	18
3.5.2. <i>Sorting</i>	19
3.5.3. <i>Date and time</i>	19
3.5.4. <i>Number and currency</i>	20
3.6. Social responsibilities and ethics	21
3.7. Cultural concerns.....	22
3.7.1. <i>Colors</i>	23
3.7.2. <i>Visual elements</i>	24
3.7.3. <i>Noise elements</i>	25
3.7.4. <i>Laws and customs</i>	26
3.8. Summary of proposed requirements	27
3.9. Synopsis	28
CHAPTER 4. IS THERE AN ADEQUATE FRAMEWORK FOR SOFTWARE INTERNATIONALIZATION?.....	30
4.1. LISA Education Initiative Taskforce (LEIT)	30
4.2. LEIT's Market Needs Survey.....	32
4.3. Case studies evaluating students' perceptions	32
4.4. Research question.....	33
CHAPTER 5. INDUSTRY SURVEY QUESTIONNAIRE DESIGN	35
5.1. Foreword	35
5.2. The plan.....	35
5.3. Goals	36
5.4. Summary of hypotheses	37

5.5. Method of data observations and hypothesis tests.....	39
5.5.1. <i>P-value and significance level</i>	39
5.5.2. <i>Check for sample size</i>	40
5.5.3. <i>Calculating the averages</i>	42
5.5.4. <i>Test for equal proportions</i>	42
5.5.5. <i>Test for significant majority</i>	42
5.6. Subjects	43
5.7. Survey design and procedure.....	44
5.7.1. <i>Tools for gathering data</i>	44
5.7.2. <i>Human Subjects Committee and consent web page</i>	44
5.7.3. <i>Survey questionnaire</i>	45
5.7.4. <i>Procedure</i>	45
CHAPTER 6. INDUSTRY SURVEY QUESTIONNAIRE ANALYSIS	47
6.1. Response rate.....	47
6.2. Demographics.....	47
6.2.1. <i>Participating companies</i>	47
6.2.2. <i>Locations</i>	48
6.2.3. <i>Years working in the industry</i>	48
6.2.4. <i>Titles</i>	49
6.2.5. <i>Highest level of education</i>	49
6.2.6. <i>Group</i>	50
6.3. Calculating the averages.....	51
6.4. Check for sample size.....	52
6.5. Test for equal proportions	52
6.6. Test for significant majority	53
CHAPTER 7. FRAMEWORK FOR SOFTWARE INTERNATIONALIZATION VALIDATED AGAINST INDUSTRY REQUIREMENTS	56
CHAPTER 8. CONCLUSION	59
8.1. Summary	59
8.2. Lessons learned	60
8.3. Threats to validity.....	60
8.4. Future work	60
8.5. Contributions	61
CHAPTER 9. BIBLIOGRAPHY	62
APPENDIX A. GLOSSARY AND ACRONYMS.....	67
A.1. ACM (Association for Computing Machinery).....	67
A.2. ASCII (American Standard Code for Information Interchange)	67
A.3. Cal Poly	67
A.4. g11n (Globalization).....	68
A.5. GILT (Globalization, Internationalization, Localization and Translation).....	68
A.6. i18n (Internationalization)	68
A.7. ICU	68

A.8. IEEE	68
A.9. ISO	68
A.10. I10n (Localization)	68
A.11. LISA	68
A.12. Unicode	69
APPENDIX B. CAL POLY INFORMATION.....	70
B.1. Members of the Computer Science Department Industrial Advisory Board (Spring 2009)	70
APPENDIX C. STATISTICS OF INTERNET USERS.....	71
C.1. Internet users ranked by top ten languages	71
C.2. Internet users ranked by region.....	72
C.3. Synopsis.....	73
APPENDIX D. STUDENTS' CURRENT PERCEPTIONS	74
D.1. Foreword	74
D.2. Goals.....	74
D.3. Subjects	74
D.4. Survey design and procedure.....	75
<i>D.4.1. Human Subjects Committee and consent form</i>	<i>75</i>
<i>D.4.2. Survey.....</i>	<i>75</i>
<i>D.4.3. Procedure.....</i>	<i>76</i>
D.5. Data results and analysis	76
<i>D.5.1. Data filtering</i>	<i>76</i>
<i>D.5.2. Demographics</i>	<i>77</i>
<i>D.5.3. Summary of data results</i>	<i>77</i>
D.6. Threats to validity.....	78
D.7. Future work	79
D.8. Synopsis.....	79
APPENDIX E. STUDENTS' PERCEPTIONS ON A LABORATORY ASSIGNMENT	81
E.1. Foreword.....	81
E.2. Goals and research questions	81
E.3. Subjects.....	82
E.4. Survey design and procedure	82
<i>E.4.1. Human Subjects Committee and consent form.....</i>	<i>82</i>
<i>E.4.2. Pre-lab survey questionnaire and procedure.....</i>	<i>82</i>
<i>E.4.3. Internationalization laboratory assignment and procedure</i>	<i>84</i>
<i>E.4.4. Post-lab survey questionnaire and procedure</i>	<i>85</i>
E.5. Summary of hypotheses	86
E.6. Data results	87
<i>E.6.1. Demographics.....</i>	<i>87</i>
<i>E.6.2. Survey questionnaire.....</i>	<i>88</i>
E.7. Data analysis	91
<i>E.7.1. Calculating changes in perceptions.....</i>	<i>91</i>
<i>E.7.2. Changes in perceptions.....</i>	<i>91</i>
<i>E.7.3. Nonparametric one-sample signed test.....</i>	<i>94</i>

E.8. Results.....	94
E.9. Threats to validity	94
E.10. Future work.....	95
E.11. Synopsis	95
APPENDIX F. STUDENTS' CURRENT PERCEPTIONS DOCUMENTS AND DATA	97
F.1. Figures.....	97
F.2. Tables.....	99
APPENDIX G. STUDENTS' PERCEPTIONS THROUGH A LABORATORY ASSIGNMENT DOCUMENTS AND DATA	100
G.1. Figures	100
G.2. Tables	105
APPENDIX H. INDUSTRY SURVEY QUESTIONNAIRE DOCUMENTS AND DATA.....	108
H.1. Figures	108
H.2. Tables	115
H.3. Titles.....	120
H.4. Participating companies.....	122
H.5. Survey results	123
H.6. Graphs of the survey results	124

LIST OF TABLES

Table 1 - Cultural associations of color [35].	24
Table 2 - Abbreviations for the statements used in the industry survey questionnaire.....	38
Table 3 – Tracing each of the abbreviations to the sections in Chapter 3	39
Table 4 - Averages of the Likert scale values of each of the statements	51
Table 5 – Summary of results from test for significant majority.....	53
Table 6 - Framework of requirements the industry overall agree listed in order of importance.	57
Table 7 - Summary of results from the survey questionnaire on students' perceptions.....	77
Table 8 - Summary of hypotheses for the research questions.	86
Table 9 - Summary of results from the first three questions of the pre- and post-survey questionnaire.	89
Table 10 - Summary of results from the last three questions of the pre- and post-survey questionnaire.	89
Table 11 - Assigning values to the Likert scale used in the survey questionnaire.	91
Table 12 - Summary of p-values after conducting a nonparametric one-sample signed test.....	94
Table 13 - Survey questionnaire responses.	99
Table 14 - Pre-lab survey results from the survey questionnaire.	105
Table 15 - Post-lab survey results from the survey questionnaire.....	106
Table 16 - Change in survey results from the survey questionnaire.....	107
Table 17 – Results calculated for the power and effect size of the survey results.....	115
Table 18 - Locations of participants of the industry survey questionnaire in South America.....	116
Table 19 - Locations of participants of the industry survey questionnaire in Asia.....	116
Table 20 - Locations of participants of the industry survey questionnaire in Europe.	116
Table 21 - Locations of participants of the industry survey questionnaire in North America.....	117
Table 22 - Results for Test for Equal Proportions.....	118
Table 23 - Results of Z Values for Test for Significant Majority.....	118
Table 24 - Results of P-Values for Test for Significant Majority	119
Table 25 - Industry responses on number of years working in the industry.....	119
Table 26 - Summary of survey results.....	123

LIST OF FIGURES

Figure 1 - An image of a standard folder icon.....	24
Figure 2 - An image of a mailbox with a flag.	25
Figure 3 - An image of a trash bin.....	25
Figure 4 - Hand gesture meaning an affirmation in the United States.....	25
Figure 5 - Bonferroni Adjustment	40
Figure 6 - Test for sample size with effect size value.	41
Figure 7- Non-centrality parameter for a Chi-Square distribution.	41
Figure 8 - Minimal sample size with the non-centrality parameter at 15.	41
Figure 9 - Test for equal proportions with Chi-Squared Goodness of Fit test.....	42
Figure 10 - Test for significant majority with one proportion z-test on "Overall Agree."	43
Figure 11 - Test for significant majority with one proportion z-test on "Strongly Agree."	43
Figure 12 - Test for significant majority with one proportion z-test on "Overall Disagree."	43
Figure 13 - Test for significant majority with one proportion z-test on "Strongly Disagree."	43
Figure 14 - Industry responses by location separated by continent.	48
Figure 15 - Industry responses to the number of years working in the industry.....	49
Figure 16 - Industry responses to "Highest Level of Education."	50
Figure 17 - Industry responses to "Group."	50
Figure 18 - Number of internet users ranked by the top 10 languages on June 30, 2009.....	71
Figure 19 - Growth of internet users ranked by the top 10 languages from 2000 to June 30, 2009.	72
Figure 20 - Number of internet users ranked by region on June 30, 2009.	72
Figure 21 - Growth of internet users ranked by region from 2000 to June 30, 2009.....	73
Figure 22 - Pie chart of participants of the survey questionnaire by major.	77
Figure 23 - Summary of results from the survey questionnaire on students' perceptions.	78
Figure 24 - Pie chart of participants of the case study by major.....	87
Figure 25 - Pie chart of participants of the case study by year of study.	87
Figure 26 - Pie chart of participants of the case study by number of languages.	88
Figure 27 - Summary of results from the first part of the pre- and post-survey questionnaire.....	89
Figure 28 - Summary of results from the second part of the pre- and post-survey questionnaire.	90
Figure 29 - Equation to calculate change in student's perceptions.	91
Figure 30 - Histogram of change of "i18n is a familiar topic to me."	92
Figure 31 - Histogram of change of "i18n will be important to my future career."	92
Figure 32 - Histogram of change of "i18n should be taught more at Cal Poly."	92
Figure 33 - Histogram of change of "Learning i18n."	93
Figure 34 - Histogram of change of "Implementing, deploying software that supports i18n."	93
Figure 35 - Histogram of change of "Fixing, maintaining software that supports i18n."	93
Figure 36 - Informed consent form for a survey on students' perception.	97
Figure 37 - Internationalization survey questionnaire on students' perception.	98
Figure 38 - Informed consent form for a case study on the change of students' perception.	100

Figure 39 - Pre-lab internationalization survey questionnaire on students' perception.....	101
Figure 40 - First page of the Internationalization Laboratory Assignment.	102
Figure 41 - Second page of the Internationalization Laboratory Assignment.	103
Figure 42 - Post-lab internationalization survey questionnaire on students' perception.	104
Figure 43 - Informed consent page of the industry survey.....	108
Figure 44 - Demographics page of the industry survey.....	109
Figure 45 - First page of the industry survey.....	110
Figure 46 - Second page of the industry survey.	111
Figure 47 - Confirmation page of the industry survey.	112
Figure 48 - Discussion and details with Dr. Karen McGaughey (1 of 3).....	113
Figure 49 - Discussion and details with Dr. Karen McGaughey (2 of 3).....	114
Figure 50 - Discussion and details with Dr. Karen McGaughey (3 of 3).....	115
Figure 51 - Industry responses to "Learn."	124
Figure 52 - Industry responses to "Apply."	124
Figure 53 - Industry responses to "Integrate."	124
Figure 54 - Industry responses to "Course."	124
Figure 55 - Industry responses to "Abroad."	125
Figure 56 - Industry responses to "Technology-Aware."	125
Figure 57 - Industry responses to "Technology-Apply."	125
Figure 58 - Industry responses to "Definitions."	125
Figure 59 - Industry responses to "Standards."	126
Figure 60 - Industry responses to "Colors."	126
Figure 61 - Industry responses to "Visual."	126
Figure 62 - Industry responses to "Noise."	126
Figure 63 - Industry responses to "Laws."	127
Figure 64 - Industry responses to "Translation-Aware."	127
Figure 65 - Industry responses to "Translation-Apply."	127
Figure 66 - Industry responses to "Direction-Aware."	127
Figure 67 - Industry responses to "Direction-Apply."	128
Figure 68 - Industry responses to "Collation-Aware."	128
Figure 69 - Industry responses to "Collation-Apply."	128
Figure 70 - Industry responses to "Dates-Aware."	128
Figure 71 - Industry responses to "Dates-Apply."	129
Figure 72 - Industry responses to "Currency-Aware."	129
Figure 73 - Industry responses to "Currency-Apply."	129
Figure 74 - Industry responses to "Ethics."	129

CHAPTER 1. INTRODUCTION

The earliest software products were developed by Americans for Americans [5]. Similarly, other software was also developed by people within a single culture or language for others in the same culture or language [6]. Not until the 1980s did the software industry start to see the need for software internationalization among the customers of other nations and cultures [7]. Various factors led up to this moment, including an increase in the number of computers around the world, an increased demand for software outside the United States, and the development of the internet [5] [8].

Software internationalization (i18n) is the process of producing an application that can be localized for a particular country without any changes being made to the program code [9]. Localization (l11n) is the process of adapting software for a particular geographical region or locale [10]. Software internationalization is a precursor to localization, which lowers the effort and cost of localization while increasing the speed of accuracy of localization [8]. As well, localization is the process of “plugging in” locale-specific data into the internationalized structure [2].

Software internationalization did not begin at the same time as software or computers. One underlying inquiry motivating this thesis is:

Is software internationalization important?

One study concluded that “[software] internationalization is one of the most important aspects of software development with respect to the globalization process in the world” [11] [12]. The globalization process is defined as making all the necessary technical, financial, managerial, personnel, marketing and other enterprise decisions necessary to facilitate international business [10]. This includes different forms of social interaction in the world, such as an e-business framework, web-based customer relationship management, and a web-based banking interface [13]. As countries become more dependent on each other, their economic, political, and social stability are due to the advances of technology that help foster these kinds of relationships [1].

Software internationalization sales constitute more than half of the revenues of the top 100 United States software companies [5] [14]. The individual revenues of the 2006 Global Fortune 500 companies yield an annual combined revenue of approximately \$5.9 trillion for internationalized software with profits of \$365 billion [15]. Microsoft, the leading developer of personal computer software, states that its foreign

sales through software internationalization constitute more than 60 percent of its total revenue, which exceeds five billion dollars [15] [16]. Sun Microsystems, a manufacturer of workstations, servers, and software, stated that 6 to 20 percent of its profit was at stake if it did not globalize [17] [18]. In 2001, the additional revenue averaged \$10 for every dollar spent on localization, compared to today's average which increased to \$25 for every dollar spent on localization [15]. Overall, software internationalization brings huge savings, increased revenues and profits, and shorter product-time to market with little or no modification [5].

In 2001, the ACM and IEEE Computing Curriculum stated that it is necessary to address “the need to develop implementation models that are international in scope and could be practiced in universities around the world” [1]. With increasing connectivity through the internet, the move towards a global economy and growing use of technology places software internationalization as an increasingly important concern for developers [2]. However, there has been a “clear shortage in terms of numbers of trained persons applying for entry-level positions” in this area [3]. Given this, the demand for computing professionals with the knowledge of globalization has increased over time as companies acquired more foreign clients, off-shore programming practices, and culturally diverse companies [1]. Developers will face challenges such as language barriers and distributed software development teams [1].

One paper states that understanding internationalization and localization issues “is essential for companies seeking to go global or who are already global” and for the graduates whom they employ [11]. The paper also states that at least “three ingredients [should] be in place” for software internationalization to be successful: an adequate development environment, maintenance of resources and tools, and access to quality or skilled graduates [11]. Having quality or skilled graduates brings up the next question motivating this thesis:

Are computer science and software engineering students adequately learning about software internationalization?

Eric Brechner, Director of Microsoft Development Training, wrote a paper entitled, “Things They Would Not Teach Me of in College: What Microsoft Developers Learn Later” [4]. Brechner suggested five new courses to add to the computer science curriculum in universities due to the growing “gap between what college graduates in any field are taught and what they need to know to work in industry” [4]. He

concludes that his suggestion is actually four new courses since “globalization and accessibility should be part of any course of introductory programming,” stating [4]:

A course on globalization and accessibility is long overdue on college campuses. It is embarrassing to take graduates from a college with a diverse student population and have to teach them how to write software for a diverse set of customers. This should be part of introductory software development. Anything less is insulting to students, their family, and the peoples of the world.

Eric Brechner used some strong words and has shown that a topic such as globalization is just not any ordinary academic topic but a theory in practice. This relates to the “Learn by Doing” philosophy at California Polytechnic State University (Cal Poly) [19]. Cal Poly prides itself on about 90 percent of its graduates reporting “working full-time or attending graduate school within one year of graduation” for the past decade [20]. To ensure that Cal Poly graduates are the “cream of the crop,” the colleges and respective departments “foster relationships with industry, scientific and professional communities, and government” [21]. At Cal Poly, the Computer Science Department seeks advice from members of the Industrial Advisory Board (Appendix B) [22]. By fostering these relationships, the colleges and respective departments gather feedback to improve their programs and curriculum each and every year. This is essential to keep graduates on the edge of the computer science and software engineering fields.

Following the spirit of Cal Poly’s “Learn by Doing” philosophy, gathering feedback from industry on how software internationalization is taught is important and necessary. Deriving feedback from industry requirements could go a long way to support student coursework and training in software internationalization. This could be a validation process for any attempts to form an academic framework for software internationalization. Unfortunately, only eleven universities identified offering some relevant program in localization, software internationalization, or globalization in 1998, three of which had an academic software internationalization course [3]. Since the report was over ten years old, a brief search on Google recently revealed only seven colleges or universities today cover software internationalization [11] [23] [24] [25] [26] [27] [28].

Universities find “little incentive to add anything” to computer science curricula, since the curricula are already rigorous [3]. Although the ACM and IEEE Computing Curriculum stated that it was necessary to address “the need to develop implementation models that are international in scope and could be practiced in universities around the world,” ACM has no provision for internationalization training

[1][3]. However, some universities state that they might focus on internationalization issues if they were convinced that it would help their students find jobs [3]. Given this, there is a need for a “standard set of skills and knowledge that could be taught by any knowledgeable instructor” [3].

Among the universities identified, the courseware was dependent on the instructor who had developed the materials [3]. Although a typical course involved presentations, handouts, and photocopied articles, the materials and syllabus were “neither standardized, nor easily portable” for instructors [3]. A syllabus, defined as an outline course of study, could only scratch the surface of software internationalization with no guarantee that it has been validated against industry requirements [29]. To make broader changes, it is important to develop an underlying set of requirements that can be taught by other instructors and then validate them against industry requirements of today.

There is very little research into how the subject of software internationalization should be taught to address the major requirements of the industry [4]. Addressing this unresolved problem, the LISA Education Initiative Taskforce (LEIT) was formed in March 1998 to evaluate the state of education in software internationalization, localization, and globalization [3]. LEIT’s intermediate plan of action was the development of a model curricula or a framework that “would provide a standard set of skills and knowledge that could be taught by any knowledgeable instructor” [3]. Without loss of generality, the term “framework” in this thesis will be known as “a standard set of skills and knowledge that could be taught by any knowledgeable instructor” [3]. However, LEIT stated that such a framework “would also need to be based on educational consensus and the approval of business contacts who could affirm that the skills and knowledge taught met real demands both from an educational perspective and from a business perspective” [3]. The “educational consensus and the approval of business contacts” is known as a validation step to the overarching requirement. Overall, the framework “would promote communication between academia and industry ... that could serve as a common reference point in discussions” [3].

The research question of the thesis is thus,

Is there a framework for software internationalization that has been validated against industry requirements?

The answer is no. Since no such framework for software internationalization currently exists, one will be developed here. The contribution of this thesis includes a provisional framework to prepare graduates to internationalize software and a validation of the framework against industry requirements. The general

hope of this framework is to provide a portable and standardized set of requirements for computer science and software engineering programs to teach future graduates.

The next few chapters of the thesis will not only attempt to answer the research question, but also to discuss the topic of software internationalization. Chapter 2 provides insight into why software internationalization is important. Chapter 3 explores the present state of software internationalization education, and includes a discussion of the underlying set of requirements for software internationalization. The definition of a “requirement” and how it pertains to the thesis is introduced in Chapter 3. The research question is presented in Chapter 4, which concludes whether or not there is an adequate framework for software internationalization. Chapter 5 provides a discussion on the design of the industry survey questionnaire to validate the underlying set of requirements discussed in Chapter 3. The results of the industry survey questionnaire and analysis of the hypothesis testing are presented in Chapter 6. Chapter 7 includes a discussion of the framework for software internationalization validated against industry requirements. Lastly, Chapter 8 presents the conclusions, contributions, and future work of the thesis.

CHAPTER 2. IS SOFTWARE INTERNATIONALIZATION IMPORTANT?

The earliest software products were developed by Americans for Americans [5]. In the 1960s, IBM produced 70 percent of the world's computers, of which 80 percent were used in the United States [30]. This meant there was little or no demand for software or computers outside the United States [5]. In 1969, software became a commodity when the Department of Justice ruled that IBM must sell its software separate from its machines [17] [31] [32]. This ruling paved the way for the software industry as it sought this new source of revenue [32].

Not until the 1980s did the software industry start to see the need for software internationalization through a diverse set of customers within various nations and cultures [7]. In the 1990s, the software industry then developed software architectures to adapt to software internationalization [13]. Various factors led up to this moment, including an increase in the number of computers around the world, an increased demand for software outside the United States, and the development of the internet [5] [8]. A paper by Jose Coronado and Carrie Livermore on globalization states [33]:

The bottom line is that companies cannot sell English products throughout the non-English-speaking world. In order to maximize profits overseas, companies must be willing to face the challenges inherited with globalization of their products.

Software internationalization (i18n) is the process of producing an application that can be localized for a particular country without any changes being made to the program code [9]. The first attempt toward software internationalization involved localization [5]. Localization (l11n) is the process of adapting software for a particular geographical region or locale [10]. Software internationalization is a precursor to localization to lower the effort and cost of localization while increasing the speed of accuracy of localization [8]. Localization is the process of “plugging in” locale-specific data into the internationalized structure [2]. This process involves modifications to formatting, display of numbers, and date and time formats, as well as displaying character sets [23].

One question necessary to this work:

Is software internationalization important?

One study concluded that “[software] internationalization is one of the most important aspects of software development with respect to the globalization process in the world” [11] [12]. The globalization process is defined as making all the necessary technical, financial, managerial, personnel, marketing and other

enterprise decisions necessary to facilitate international business [10]. This includes different forms of social interaction in the world, such as an e-business framework, web-based customer relationship management, and a web-based banking interface [13]. Software is “developing and expanding to support global activities that require fast communication, access to distributed data, and advanced security features” [1].

2.1. Is software internationalization a trivial process?

As countries become more dependent on each other, their economic, political, and social stability is due to the advances of technology that help foster these relationships [1]. Although some project managers and programmers believe that software internationalization is just language translation, they overlook “multilingual project management, online help engineering, memory alignment management, and multilingual product support” [1]. Some fail to see that software is more than just algorithms when it also “involves the user interface, online help, button and dialog box labels, money and business plans, marketing, and of course, unpredictable users” [4].

To understand the diversity of people from around the world, Appendix C includes a discussion about the statistics of the internet users on June 30, 2009. The conclusion that can be reached is that the English language does not constitute the majority of the internet market, but rather that the Arabic and Chinese languages along with languages from the regions of the Middle East and Asia are taking precedence; these geographic areas are growing markets for globalization. The internet is of particular interest because people from all over the world are connecting with the assistance of technology through the integration of cell phones, the internet, and many software applications [2] [34]. The internet increases communication between countries, disseminates vast amounts of information and aids global businesses in all areas of the world [1]. There are about 3,000 spoken languages throughout the world with slightly more than 100 written languages [17]. A study has shown that people who interact with computers in their native language learn faster while being more skilled and capable of avoiding mistakes [5] [8].

Another significant benefit of software internationalization is that it encourages user-centered development by forcing developers to take into account a greater variety of potential users [6]. This may lead to better structural design of the software including the interaction aspects [6]. However, the potential

drawback is that such a design may be less targeted at individual constituencies, and they may have a sub-optimal user experience [6].

2.2. Is there market potential for software internationalization?

Many United States manufacturers “derive a large percentage of their revenues and profiles from international sales” [8]. Software internationalization sales constitute more than half of the revenues of the top 100 United States software companies [5] [14] [35]. The individual revenues of the 2006 Global Fortune 500 companies yielded an annual combined revenue of approximately \$5.9 trillion for internationalized software with profits of \$365 billion [15]. Microsoft, the leading developer of personal computer software, stated that its foreign sales through software internationalization constituted more than 60 percent of its total revenue, which exceeded five billion dollars [15] [16]. Sun Microsystems, a manufacturer of workstations, servers, and software, stated that 6 to 20 percent of its profit was at stake if it did not choose to globalize [17] [18]. Although the software industry was booming in the United States, companies such as Germany’s SAP (Systems Applications and Products in Data Processing) and Canada’s Corel became leaders in their respective fields of database and graphics software [11]. Eventually, India developed a billion-dollar software industry and Japan became the second-largest software-producing country, only behind the United States [11]. In 2001, the additional revenue averaged \$10 for every dollar spent on localization, compared with today’s increased average of \$25 for every dollar spent on localization [15]. Overall, software internationalization renders huge savings, increased revenues and profits, and shorter product time to market with little or no modification [5] [36].

CHAPTER 3. WHAT IS THE PRESENT STATE OF EDUCATION FOR SOFTWARE INTERNATIONALIZATION?

3.1. Foreword

In 2001, the ACM and IEEE Computing Curriculum stated that it was necessary to address “the need to develop implementation models that are international in scope and could be practiced in universities around the world” [1]. With increasing international connectivity through the internet, the move towards a global economy and growing use of technology places software internationalization as an increasingly important concern for developers [2]. Computing professionals are affected by the “economic, political, legal, and social issues of the new international work environment” [1]. However, there has been a “clear shortage in terms of numbers of trained persons applying for entry-level positions” in this area [3]. The translation and customization of software requires “a variety of specialists, such as programmers, localization engineers, quality assurance (QA) specialists and project managers” [8]. Developers should consider that eliciting “requirements on the software to be multilingual is increasing heavily” [37].

The demand for computing professionals with the knowledge of globalization has increased over time as companies have acquired more foreign clients, off-shore programming practices, and cultural diversity [1]. Developers will face challenges such as language barriers and software development teams distributed throughout the world [1]. One university developed a course for software internationalization because it believed that a software internationalization project involves a significant amount of new theory and novel practice, and thus warrants a course in its own right [11]. One paper states that understanding internationalization and localization issues “is essential for companies seeking to go global or who are already global” and for the graduates whom they employ [11]. The same paper also requires at least “three ingredients to be in place” for software internationalization to be successful: adequate development environment, maintenance of resources and tools, and access to quality or skilled graduates [11]. Having quality or skilled graduates brings up the next question motivating this thesis:

What is the present state of education for software internationalization?

Eric Brechner, Director of Microsoft Development Training, wrote a paper entitled, “Things They Would Not Teach Me of in College: What Microsoft Developers Learn Later” [4]. Brechner suggested five new courses to add to the computer science curriculum due to the growing “gap between what college

graduates in any field are taught and what they need to know to work in industry” [4]. According to Brechner, the requirements to satisfy the industry needs or objectives are incomplete and infeasible. He concludes that his suggestion is actually four new courses since “globalization and accessibility should be part of any course of introductory programming,” stating [4]:

A course on globalization and accessibility is long overdue on college campuses. It is embarrassing to take graduates from a college with a diverse student population and have to teach them how to write software for a diverse set of customers. This should be part of introductory software development. Anything less is insulting to students, their family, and the peoples of the world.

The state of education for software internationalization can be viewed in the context of a requirements problem. Karl Wieggers defines a requirement as [38]:

A statement of a customer need or objective, or of a condition or capability that a product must possess to satisfy such a need or objective. A property that a product must have to provide value to a stakeholder.

Characteristics of an excellent requirement are those that are complete, correct, feasible, necessary, prioritized, unambiguous, and verifiable [38].

The problem addressed in this thesis can be viewed as one of the requirements. The industry acts in part as a customer with a need for qualified computer science and software engineering students with the knowledge of software internationalization. The universities must provide graduates with sufficient knowledge and training in software internationalization. In the context of the requirements definition, the customers are those in the industry, the products are the qualified computer science and software engineering students, and the customer need or objective is for students to have the knowledge of software internationalization. Therefore, the overarching requirement within this thesis is for universities to provide qualified computer science and software engineering students with the knowledge of software internationalization to the industry. The framework should “promote communication between academia and industry ... that could serve as a common reference point in discussions” [3].

In an attempt to meet the customer’s needs and objectives for students with knowledge of software internationalization, this chapter will break down the one overarching requirement to other requirements developed from a literature review. This chapter will discuss an underlying set of requirements that embodies the state of education in software internationalization.

First, the chapter will discuss three educational architectural models (Section 3.2) proposed by Paula Gabbert as proposed solutions to the overarching requirement [1]:

- Integration into Existing Courses (Section 3.2.1)
- Develop an Entire Course (Section 3.2.2)
- Study Abroad for Experience (Section 3.2.3)

The three educational architectural models can be used to derive requirements as proposed solutions to the problem.

Second, the chapter will discuss many of the requirements of education for software internationalization broken down and derived from the customer's overarching requirement. The following requirements were compiled and supported from various authors including, Elvis Hau and Manuela Aparício [8], Patricia Russo and Stephen Boor [35], Tony Fernandes [39], Erica Young [17], and Steffen Gross [37]:

- Definitions and Standards (Section 3.3)
- Development Practices (Section 3.4)
- Translation and Documentation (Section 3.5)
- Social Responsibilities and Ethics (Section 3.6)
- Cultural Concerns (Section 3.7)

The three educational architectural models and number of requirements will be discussed and summarized in Section 3.8.

3.2. Educational architectural models

The first part of the chapter analyzes how students should learn about software internationalization. There are three possible educational architectural models of software internationalization as proposed solutions to the overarching requirement [1]:

- Integrating software internationalization into existing computing courses (Section 3.2.1)
- Developing an entire course devoted to software internationalization (Section 3.2.2)
- Studying abroad for practical experience with software internationalization (Section 3.2.3)

Each of the educational models for software internationalization will be discussed in detail. Whichever model is used, a framework should still require social, economic, political, and legal issues to be integrated into the computing curriculum [1].

3.2.1. Integration into existing courses

Today, most computing programs employ the model of integrating software internationalization into existing computing courses [1]. This makes sense in allowing students to see how software internationalizations are at work with other fields of computer science and software engineering. The ACM and IEEE Computing Curriculum 2001 identified the following courses using this educational architectural model, including ethics or social issues courses, capstone courses, and team-based courses [1]. In the survey questionnaire, students indicated software engineering, databases, graphical user interface design, systems programming, professional responsibilities, and general education courses that covered software internationalization, detailed in Appendix D and Appendix E [40].

The tradeoff to this model is that it does not imply that topics of software internationalization were discussed in depth or applied to projects or laboratory assignments [1]. A discussion in Appendix E shows one simple attempt to determine if students' perceptions changed through an internationalization laboratory assignment. The model can only at best cover an overview of software internationalization and have "computer science students aware of some of the issues of globalization" [1]. Lastly, the model would have difficulty integrating all the requirements of software internationalization by showing relationships between each aspect sufficiently [1]. This tradeoff shows that students cannot completely fulfill the customer's overall requirement that is complete, feasible, and correct.

3.2.2. Develop an entire course

It is undoubtedly rare for any computing program to adopt the model of developing an entire course for software internationalization [34]. In 1998, only eleven universities identified offering some relevant program in localization, software internationalization, or globalization, and only three had an academic software internationalization course [3]. Since the report is over ten years old, a brief search on Google recently revealed only seven colleges or universities today cover software internationalization. The colleges and universities are Austin Community College [23], Chico State University [24], Elon University

[25], Queensland University of Technology [11], Rensselaer Polytechnic Institute [26], San Diego State University [27], University of Mississippi [28].

This model is neither feasible nor cost-effective at this time because most computer science curricula are already rigorous and universities find “little incentive to add anything” [3]. In addition, this model would be challenging for professors whose teaching material and textbooks all stem from a variety of different sources instead of a central location [1]. Another consequence of this model is that students may graduate with little perspective of how software internationalization plays into other requirements of computer science and software engineering [1]. An entire course devoted to software internationalization may result in students having a “lack of small-scale real-world problems suitable for use as hands-on exercises in the classroom environment” [11]. However, it is typical for organizations such as LISA to provide consultants to teach a course or a workshop to companies in the industry which may fulfill this requirement or part of it [41].

The tradeoff for adopting this model is that students who are interested in an internationally-scoped career will greatly benefit from this course [1]. Queensland University of Technology in Australia has an entire course devoted to software internationalization [11]. It was a 13-week graduate course that “introduces students to the strategies, technologies, techniques and current development associated with” the growing software development for the world [11]. The course included lecture discussion topics and practical laboratory activities with an “application-driven laboratory examination” [11]. Students learned about what is involved in planning and managing a software internationalization project as well as designing, building, and using a software internationalization application [11]. Some topics included in the course were translation, localization, modularization, resource bundles, documentation and online help, web services, tools and standards, and other contexts such as cultural, law, and finance concerns [11]. The university surveyed how the students felt: 76 percent felt the course was successful and timely, 86 percent of the students stated they learned something valuable, and 72 percent of them would recommend the course to a friend [11].

Although this model satisfies the completeness and correctness of the overall requirement of the customer’s objective, developing an entire course is not be feasible because it is currently not cost-effective or necessary at the time being.

3.2.3. Study abroad for experience

Another model is a studying abroad program for experience in software internationalization. Studying abroad for practical experience incorporates the students' experiences entirely into the computing curriculum [1]. Students would work alongside international computer scientists and observe the different working environments [1]. One paper “recommends traveling to the target country and conducting usability tests as the best choice in international usability testing” [42] [43]. In comparison to the other models, studying abroad allows students to experience a different culture first hand while integrating globalization and technology material into their work and laboratory assignments [1]. In addition to witnessing how technology impacts a society through business and culture, students have a chance to understand the cultural issues, practices, and laws of various areas [1] [34]. This is important, since students will have the user’s cultural background in mind to more accurately predict the user interface preferences during software internationalization [44].

The tradeoff of this model is that this will only benefit a limited number of students [1]. Students may not be willing to commit the time and money to study abroad, even when they will gain a more thorough background in software internationalization. One recommendation is to create partnerships with other universities to exchange requirements on software internationalization [34]. This would allow collaboration with international partners [6]. Through the internet, universities could use web conferencing tools, such as WebEx or Adobe Acrobat Connect, to communicate with each other and bring the study abroad experience closer to home [2][34].

Like the previous model, this one also satisfies the completeness and correctness of the overall requirement of the customer’s objective, but it is not feasible because it is currently not cost-effective or necessary at this time.

3.2.4. Conclusion

We just discussed the three educational architectural models (Section 3.2) proposed by Paula Gabbert on how to educate students in software internationalization [1]:

- Integration into Existing Courses (Section 3.2.1)
- Develop an Entire Course (Section 3.2.2)
- Study Abroad for Experience (Section 3.2.3)

To summarize the proposed requirements to be compiled in Section 3.8, in order for computer science and software engineering graduates to be prepared for industry, students should:

- Learn software internationalization through lecture, reading, and writing (Section 3.2)
- Learn about software internationalization by integrating the subject into existing courses (Section 3.2.1)
- Learn about software internationalization by taking an entire course devoted to the subject (Section 3.2.2)
- Learn about software internationalization by studying abroad (Section 3.2.3)

Although each of the educational architectural models would carry out the customer's requirement, there were various tradeoffs that would not completely satisfy the overarching requirement. The only model that is feasible is possibly integrating software internationalization into existing courses.

Whatever educational architectural model will be used, the overarching requirement would still need to be broken down to a number of other requirements that embody the topic of software internationalization. The chapter will discuss many of the requirements of education for software internationalization broken down and derived from the customer's overarching requirement:

- Definitions and Standards (Section 3.3)
- Development Practices (Section 3.4)
- Translation and Documentation (Section 3.5)
- Social Responsibilities and Ethics (Section 3.6)
- Cultural Concerns (Section 3.7)

3.3. Definitions and standards

The foundational requirement for learning software internationalization is to understand its definitions and standards. A study was conducted to determine students' perceptions of software internationalization, detailed in Appendix D and Appendix E. The majority of students surveyed were unfamiliar with software internationalization. Even with a small sample, this shows that untrained students are unable to distinguish and define internationalization, globalization, and localization [11]. Having this foundation would allow students and future programmers to understand the context more fully and to be able to apply other requirements of software internationalization.

Students should be aware of one of the critical milestones in software internationalization, Unicode. Within software internationalization, “Unicode has provided a lingua franca for communicating textual data” [45]. The Unicode Worldwide Character Standard contains “letters, digits, diacritics, punctuation marks, and technical symbols for all the world's principal written languages, using a uniform encoding scheme” [46]. When the first version of Unicode was introduced in 1991, it provided a specific number for each and every character for 99 percent of the world’s known languages [36] [46] [47].

In addition, students also need to distinguish and define the software internationalization standards of the ISO, W3C, and LISA [11]. One important standard is the Common Locale Data Repository (CLDR), an industry standard for locale data [36]. The adoption of CLDR brings consistency across industry including display names, date and format, number format, and rules for parsing or collation [36]. There are always evolving concepts including debates on XML standards and localization, standards on locales, and standards on online documentation and format [11]. Being aware of these standards will give the students access to a variety of resources and documents when implementing requirements of software internationalization.

To summarize the proposed requirements to be compiled to Section 3.8, in order for computer science and software engineering graduates to be prepared for industry, students must:

- Define and distinguish between software internationalization, localization, and globalization (Section 3.3)
- Define and distinguish various industry standards on software internationalization (ISO, W3C, LISA) (Section 3.3)

3.4. Development practices

In order for students to be aware of how software internationalization works, another requirement is that students should at least apply and “critically compare and assess” the available technologies for software internationalization [11]. The “internationalization of software requires special diligence on all tasks related to software design and implementation” [37]. Given this, “students are exposed to the different software development techniques used by organizations in this arena and the perils and pitfalls of managing software internationalization projects” [11]. Instead of reinventing the wheel, technologies such as the .NET framework, ICU Project, and resource bundles in Java and Adobe Flex Builder allow software

internationalization and localization to function properly [11] [34] [36] [48]. For example, Java “has a huge set of supported locales where all necessary changes are already implemented” [37]. Without learning about these tools, “localization of complex software is very difficult and time consuming and quite often involves the same task to be uselessly repeated over and over again” [8].

Similarly, students should understand the benefits of Unicode. Unicode provides a common locale data repository, case mapping, bi-directional text reading, security, and mechanisms such as sorting, searching, and matching [36]. Unicode provides details of character properties, processing algorithms, and definitions that are useful for implementation [8]. Unicode is set up to avoid data corruption and makes software globalization possible by reducing development, maintenance, update, and support costs [36].

Following the “Learn by Doing” philosophy, students should get “down and dirty” with software internationalization in order to be aware of the technologies and the difficulties. For example, students should be aware that although internationalized applications may move smoothly between differing programming languages, it might not move as easily between different operating systems [11].

To summarize the proposed requirements to be compiled to Section 3.8, in order for computer science and software engineering graduates to be prepared for industry, students must:

- Apply software internationalization through laboratory assignments, projects, and activities (Section 3.2, 3.4)
- Be aware of possible technologies and tools for creating software that supports internationalization (Section 3.4)
- Apply the possible technologies and tools for creating software that supports internationalization (Section 3.4)

3.5. Translation and documentation

Programmers should understand the translation difficulties that occur between different languages and cultures [11]. First, words translated from English are usually 40 percent longer, requiring programmers to properly allocate enough space to use string buffers to avoid a system crash or severe failure “due to a string buffer overflow” [11] [37] [49]. Many languages require special fonts such as languages from “Japan, China, Russia and the Arabic countries” instead of the usual Times New Roman [49]. Not only does the translation apply to the text on the graphical user interface, but also the online help

and documentation [11]. There are some languages, especially Asian languages, which “have more characters in their language than keys on the keyboard” [37]. Second, it is important to be aware that there are words that exist in English that “do not exist in other languages” or “do not translate well in other cultures” [35]. For example, phrases like “disk drive,” “zooming,” or “panning” do not exist in the Thai language [35]. The word “uno” means “one” in Spanish, and “garbage” in Finnish, while the word “mist” means “manure” in German [35].

To summarize the proposed requirements to be compiled to Section 3.8, in order for computer science and software engineering graduates to be prepared for industry, students must:

- Be aware of possible technologies and tools used for language translation and text representation (Section 3.5)
- Apply the possible technologies and tools used for language translation and text representation (Section 3.4, 3.5)

Other requirements falling under translation and documentation include the following:

- Direction (Section 3.5.1)
- Sorting (Section 3.5.2)
- Date and Time (Section 3.5.3)
- Number and Currency (Section 3.5.4)

3.5.1. Direction

If a user was looking to open a bank account on a website, the text direction of the instructions should be properly placed in their respective language to avoid confusion or difficulties while reading. In English, sentences are written from left to right. This “arrangement would be counter-intuitive to an Arabic user who reads from right to left” [35] [37] [50]. In these cases, programmers need to be aware of the bidi algorithm and inline markups to ensure that sentences are shown properly [50]. There are some languages in Asian countries that “read from top to bottom and right to left” [17] [35] [36]. With languages such as Chinese, white spaces are not used for word breaks [42].

To summarize the proposed requirements to be compiled to Section 3.8, in order for computer science and software engineering graduates to be prepared for industry, students must:

- Be aware of the possible technologies and tools to represent the direction of text in various languages (Section 3.5.1)
- Apply the possible technologies and tools to represent the direction of text in various languages (Section 3.4, 3.5.1)

3.5.2. Sorting

If a user had numerous bank accounts and wanted to access one of them, the names should be properly sorted in their respective language to easily find the information. Another word for sorting is collation: “a comparison of information” [51]. In English, words are sorted from the letter “A” to the letter “Z.” In addition, capitalization differs among various languages [36]. Some languages are sorted depending on the dialects used and character set [36]. For example, in German, the character “ü” may have a different precedence than the character u [17]. In Spanish, the two letter sound “ch” is treated as one character between letters “c” and “d” [17] [36] [37]. In Chinese, words are sorted from the smallest number of strokes to the largest number of strokes [36]. In Japan only, the techniques to sort include “Ascending Lead Byte, Code Order, Pitch Casing, Pronunciation, and Stroke” [49].

To summarize the proposed requirements to be compiled to Section 3.8, in order for computer science and software engineering graduates to be prepared for industry, students must:

- Be aware of how text is sorted in various languages (Section 3.5.2)
- Apply the possible technologies and tools to sort text in various languages (Section 3.4, 3.5.2)

3.5.3. Date and time

When a user looks at a bank statement online with the date written as “02/03/09,” this date value can be ambiguous if the bank or the bank’s website originates from another country. In the United States, a date value of “02/03/09” means February 3, 2009 using a format of “MM/DD/YY” [35] [37]. In most of Europe, the date value would mean March 2, 2009 with the format of “DD/MM/YY” [17] [35] [36] [37] [50]. In comparison to Japan, the date value would mean March 9, 2002 with the format of YY/MM/DD [50]. The date value depends on the country and culture using it and how it’s been adopted over the years. To counteract this ambiguity, it is important that the date is written explicitly containing the name of the month and a four-digit year.

In addition to the representation of dates, differing calendaring systems exist. In the United States, the Gregorian calendar is used [17] [36] [49]. The Gregorian calendar “has a ‘leap’ year every four years to compensate for the fact that a solar year is approximately 365 1/4 days long [49]. In Israel, “years might be referred to as the 5700 decade, or the year 1990” [35]. In China, a lunar calendar is used by synchronizing “with the solar calendar by adding more months to some years as part of a 60-year cycle” [49]. There are also other calendars including the Buddhist and Japanese calendar [36]. In Japan, the Imperial calendar is “based on the date when the Emperor ascended to the throne” [49]. Therefore, in Japan, it is “very impolite to create a calendar that allows users to reset the year to the beginning, since this implies the mortality to the present emperor” [49]. In Islamic countries, the Islamic calendar “is based on Hegira, when Muhammad moved from Mecca to Medina” [49]. Lastly, in many Islamic countries, “the date doesn’t change at midnight, it changes at sunset” [49].

As for weekends, differences exist among calendar systems and they “are not necessarily contiguous” [49]. For example, weekends are Saturdays and Sundays in the United States while weekends are on Thursdays and Fridays in some Middle Eastern countries [17]. Information like this would be important to banks and accountants when calculating interests or work pay of their employees.

Differences even exist in time; some cultures that “use a 12 hour clock and include seconds” while “other cultures use 24-hour military time” [35]. In addition, instead of using a colon that is common in the United States, some use a period [35].

To summarize the proposed requirements to be compiled to Section 3.8, in order for computer science and software engineering graduates to be prepared for industry, students must:

- Be aware of how dates and times are used in various cultures and countries (Section 3.5.3)
- Apply the possible technologies and tools to decipher the date and time for various cultures and countries (Section 3.4, 3.5.3)

3.5.4. Number and currency

To the banking system, number and currency are a very important aspect of software internationalization because every conversion or translation of numbers and currencies could easily save banks money or lose banks money. For example, many countries have tax legislation that “effectively determines accounting standards” in order to claim for tax purposes [8]. In addition, a string literal of

\$123.45 does not say whether it is the value in the United States, Australia, or even Zimbabwe [36]. To add to the complexity, some countries like Portugal place the currency symbol “\$” like this: 1.234\$56 [49]. However, using “USD” next to a string literal of “\$123.45” would not be ambiguous and indicates to the user that the value is in United States Dollars [36]. It is also important to consider the symbols “£ for British pound or ¥ for Japanese Yen” to distinguish the currency [37] [49]. Any financial transaction should be accurate, meaning that the conversion and translation of numbers should be the same in every numbering system.

As for place-holders, commas are placed at every third mark of the currency value in the United States, but other countries might place them at other marks of the currency value [35] [36] [37] [52]. For example, countries in Europe use a period as a place-holder to separate large numbers instead of a comma [35] [36] [37] [52]. In China and Japan, ideographic characters are used as a place holder while the Swiss use apostrophes [52] [36].

To summarize, the proposed requirements to be compiled to Section 3.8, in order for computer science and software engineering graduates to be prepared for industry, students must:

- Be aware of how the number/currency systems are different for various cultures and countries (Section 3.5.4)
- Apply the possible technologies for the number/currency systems for various cultures and countries (Section 3.4, 3.5.4)

3.6. Social responsibilities and ethics

To be a computing professional, another requirement for students graduating from computer science or software engineering should have an awareness of their social and ethical responsibilities. Some universities require students to take a professional responsibilities and computer ethics course to understand how their actions can affect society. In addition to learning about the ACM Software Engineering Code of Ethics and Professional Practice, students would need to “demonstrate awareness of social responsibilities by being able to assess and remedy the impact of poorly or well-designed software on different groups of people” [11].

Lastly, students should be aware of some of the best practices within the software internationalization community. The GILT Industry Ethics document is the first place to start for social and

ethical responsibilities with software internationalization [53]. Please see Appendix A for more information on GILT. The GILT Industry Ethics document was developed by LISA to address the ethical business practices in the GILT industry [53]. The purpose of the GILT Industry Ethics document is “to provide a common reference for both suppliers and consumers of localization and related services as to what constitutes ethical behavior with regard to GILT-specific issues” [53].

To summarize, the proposed requirements to be compiled to Section 3.8, in order for computer science and software engineering graduates to be prepared for industry, students must:

- Be aware of the social responsibilities and ethics of producing software that supports internationalization (Section 3.6)

3.7. Cultural concerns

Lastly, cultural concerns are one aspect of software internationalization that should be regarded as an important requirement. One paper titled, “Cultural Issues and Opportunities in Computing Education” presents “an argument about why cultural issues need to be included in computing curricula” [42].

Although “more people have access to computers, little is still known about how different cultures interact with computers” [5]. One paper explained [12]:

Cultural aspects play a central role in the localization process, as claimed, proven, and validated in localization literature and real-world cases of market failures where companies did not thoroughly consider regional culture issues.

Students should be aware of how cultural concerns can affect and be affected by other cultures and countries. For example, those who work on the user interface “must increase their awareness of cross-cultural differences, and make changes to the traditional software development process” [35]. No matter how successful a project may be according to a company, the fundamental component of software internationalization is the ability to distinguish between common needs and culture-specific needs [2]. In addition, the underlying assumption behind software internationalization is that the culturally and linguistically sensitive software components need to be separated from the core of the application [8].

When identifying software requirements, cultural factors should supplement the information [12]. One paper even offered a “cross-cultural checklist of issues including text, local formats, images, symbols, colors, flow, and product functionality” [35]. It is important to identify the requirements for a user interface because it is one of the few components that are culture-dependent to the stakeholders [13]. Cultural factors

play a central role in localization and have been utilized in localization literature [12]. Culture and usability “intertwine into a single entity” where “cultural preferences and biases affect the degree of user friendliness of an interface” [54]. There even exist real-world cases of market failures when companies did not thoroughly consider cultural issues [12]. Since “social norms vary greatly between cultures, what is acceptable in one culture can be objectionable in another” [35]. Such cultural differences may damage business relationships and credibility if not addressed appropriately.

Students aren’t expected to take all the concerns into account but it would be necessary to examine the “worst faux pas” [17]. In an article published by Forbes entitled “Cultural Web Faux Pas,” he notes that when you disregard cultural differences, whatever you might be selling will be doomed [54]. The following additional requirements also fall under the requirement of being aware of cultural concerns:

- Colors (Section 3.7.1)
- Visual Elements (Section 3.7.2)
- Noise Elements (Section 3.7.3)
- Laws and Customs (Section 3.7.4)

3.7.1. Colors

Colors have different meanings in different cultures. Table 1 summarizes the cultural associations of color [35]. In China, the color red usually represents happiness or luck while it represents a warning or danger in the United States [2] [54]. Therefore, if a Chinese user saw a red button with text written in another language, the user might accidentally click the button and not realize the consequences. However, the Chinese “may be receptive to a web page with a red background” [54]. In Japan, the color white represents death in comparison to purity in the United States [54]. In Egypt, the color yellow means prosperity and in comparison to cowardice in the United States [8].

<u>Culture</u>	<u>Red</u>	<u>Blue</u>	<u>Green</u>	<u>Yellow</u>	<u>White</u>
China	Happiness	Heavens, Clouds	Ming Dynasty, Heavens, Clouds	Birth, Wealth, Power	Death, Purity
Egypt	Death	Virtue, Faith, Truth	Fertility, Strength	Happiness, Prosperity	Joy
France	Aristocracy	Freedom, Peace	Criminality	Temporary	Neutrality
India	Life, Creativity		Fertility, Prosperity	Success	Death, Purity
Japan	Anger, Danger	Villainy	Future, Youth, Energy	Grace, Nobility	Death
United States	Danger	Masculinity	Safety	Cowardice	Purity

Table 1 - Cultural associations of color [35].

To summarize the proposed requirements to be compiled to Section 3.8, in order for computer science and software engineering graduates to be prepared for industry, students must:

- Be aware of how particular colors are viewed differently among various countries (Section 3.7.1)

3.7.2. Visual elements

Visual elements can be used to enhance a program, but can also be ambiguous, offensive, or disturbing in other countries. In conservative regions, such as Islamic countries, software that includes women in bikinis would be considered extremely offensive [17]. A number of studies concluded that many images do not convey the same meaning, and like words, they “don’t always translate” [35] [37]. A standard folder icon, as shown in Figure 1, would be inappropriate for countries where documents are traditionally stored in cardboard boxes [55]. Similarly, an image of a mailbox with a red flag, as shown in Figure 2, to announce new mail is usually not seen outside the United States [17]. Although a trash bin, as shown in Figure 3, is seen commonly in the United States and “popular in Apple interfaces,” British users indicated that “it looked much more like a postal box [35].

Figure 1 - An image of a standard folder icon.

Figure 2 - An image of a mailbox with a flag.

Figure 3 - An image of a trash bin.

Programmers also “need to be careful when designing images that contain religious symbols (e.g., crosses and stars), the human body, women and hand gestures” [35]. Hand gestures is another visual element that is sometimes used within an image but have eighteen other cultural dimensions of differing hand postures [2]. For example, the hand gesture indicating “OK” in the United States, as shown in Figure 4, means “worthless” or “zero” in France, an indication of homosexuality in some Mediterranean countries, and is consider vulgar in Brazil and Germany [17] [35].

Figure 4 - Hand gesture meaning an affirmation in the United States.

To summarize the proposed requirements to be compiled to Section 3.7.2, in order for computer science and software engineering graduates to be prepared for industry, students must:

- Be aware of how particular visual elements are viewed differently among various countries
(Section 3.7.2)

3.7.3. Noise elements

Like colors, noises can have different meanings in different cultures. In Japan, it is very impolite to have the computer beep at the user because it calls attention to a possible serious user error [2] [49].

Since Japanese offices are typically more open than the ones in the United States, a beep is more likely to

cause an embarrassment [2] [49]. It is important to consider avoiding noise elements that are targeted to a certain culture or country. Noise elements that are not seen as neutral can conjure up defamatory requirements that can be heard or used as a racist or sexist noise.

To summarize, the proposed requirements to be compiled to Section 3.7.3, in order for computer science and software engineering graduates to be prepared for industry, students must:

- Be aware of how particular noise elements are viewed differently among various countries
(Section 3.7.3)

3.7.4. Laws and customs

Laws and customs also differ between cultures. There are three legal systems of common law, codified law, and commercial law that affect various countries and cultures [8]. It is important that students and computing professors “understand that laws relating to technology and digital media vary from country to country” [1]. However, “laws change worldwide as technology is developed so understanding the legal aspects of computing is an ongoing challenge for computer scientists” [1]. For example, Google abides by the Chinese government on the censorship of pornography [56]. Laws in France and Germany require that “all user interfaces, online help, and documentation must be translated” [33]. In Japan, all software and hardware, including the packaging, must be translated into Japanese [33].

Four dimensions of individualism, power distance, uncertainly avoidance, and masculinity affect cultures [8]. These dimensions relate to “issues of authority and independent thought” [2]. For example, it is socially acceptable in France for professors to only comment or share viewpoints to students’ papers because of the high hierarchy distance and the feminine orientation in France [2] [13] [35]. In comparison to the United States and Great Britain, students and teachers are able to comment on the papers together due to a low hierarchy distance and the masculine orientation [13]. While in “Scandinavian countries, independent discovery is greatly valued, so the idea that students cannot add their own viewpoints” is unacceptable [35].

Another example is how business cards hold a different meaning between the American culture and the Japanese culture [35]. In the United States, business cards “are used to convey information about a business or its representation” while in Japan, “business cards are exchanged at the start of a business encounter, and are used to establish seniority so an interaction can proceed appropriately” [35].

To summarize the proposed requirements to be compiled to Section 3.7.4, in order for computer science and software engineering graduates to be prepared for industry, students must:

- Be aware of how laws of various countries can affect the software that supports internationalization (Section 3.7.4)

3.8. Summary of proposed requirements

To summarize, in order for computer science and software engineering graduates to be prepared for industry, students must:

- Learn software internationalization through lecture, reading, and writing (Section 3.2)
- Learn about software internationalization by integrating the subject into existing courses (Section 3.2.1)
- Learn about software internationalization by taking an entire course devoted to the subject (Section 3.2.2)
- Learn about software internationalization by studying abroad (Section 3.2.3)
- Apply software internationalization through laboratory assignments, projects, and activities (Section 3.2, 3.4)
- Define and distinguish between software internationalization, localization, and globalization (Section 3.3)
- Define and distinguish various industry standards on software internationalization (ISO, W3C, LISA) (Section 3.3)
- Be aware of possible technologies and tools for creating software that supports internationalization (Section 3.4)
- Apply the possible technologies and tools for creating software that supports internationalization (Section 3.4)
- Be aware of possible technologies and tools used for language translation and text representation (Section 3.5)
- Apply the possible technologies and tools used for language translation and text representation (Section 3.4, 3.5)

- Be aware of the possible technologies and tools to represent the direction of text in various languages (Section 3.5.1)
- Apply the possible technologies and tools to represent the direction of text in various languages (Section 3.4, 3.5.1)
- Be aware of how text is sorted in various languages (Section 3.5.2)
- Apply the possible technologies and tools to sort text in various languages (Section 3.4, 3.5.2)
- Be aware of how dates and times are used in various cultures and countries (Section 3.5.3)
- Apply the possible technologies and tools to decipher the date and time for various cultures and countries (Section 3.4, 3.5.3)
- Be aware of how the number/currency systems are different for various cultures and countries (Section 3.5.4)
- Apply the possible technologies for the number/currency systems for various cultures and countries (Section 3.4, 3.5.4)
- Be aware of the social responsibilities and ethics of producing software that supports internationalization (Section 3.6)
- Be aware of how particular colors are viewed differently among various countries (Section 3.7.1)
- Be aware of how particular visual elements are viewed differently among various countries (Section 3.7.2)
- Be aware of how particular noise elements are viewed differently among various countries (Section 3.7.3)
- Be aware of how laws of various countries can affect the software that supports internationalization (Section 3.7.4)

3.9. Synopsis

This chapter discussed an underlying set of requirements that embodies the state of education for software internationalization. First, the chapter discussed three educational models (Section 3.2) for education in software internationalization. Second, the chapter discussed many of the requirements of software internationalization that are supported by numerous papers. This set of requirements will

hopefully be used as part of the resulting framework. The next chapter's analysis will entail another important question of this thesis:

Is there an adequate framework for software internationalization?

CHAPTER 4. IS THERE AN ADEQUATE FRAMEWORK FOR SOFTWARE INTERNATIONALIZATION?

The previous chapters have explored the importance and present state of education for software internationalization. Chapter 3 includes a discussion on the underlying set of requirements that embodies the state of education for software internationalization, supported by extensive research of numerous papers, such as the three educational architectural models (Section 3.2) and many of the requirements embodying software internationalization (Sections 3.3, 3.4, 3.5, 3.6, 3.7). Although Chapter 3 only presented an underlying set of requirements that can be used as a framework, “there is no real guarantee that any particular course teaches the particular skills and knowledge that industry needs” [3]. Chapter 4 will discuss the research question and why it is important.

4.1. LISA Education Initiative Taskforce (LEIT)

Recall from Chapter 3 the context of the definition of requirement; the customers are those in the industry, the product is the computer science and software engineering students, and the customer need or objective is for students to have the knowledge of software internationalization. In an attempt to meet the customer’s requirement of providing students with knowledge of software internationalization, the LISA Education Initiative Taskforce (LEIT) was formed in March 1998 to evaluate the state of education in software internationalization, localization, and globalization [3]. To carry this out, the taskforce was to [3]:

- Identify existing courseware in software internationalization, localization, and globalization
- Perform a market-needs survey of companies to determine what training the industry would like to see new graduates have
- Map existing programs to industry needs
- Present a list of recommendations to the LISA Executive Committee on what actions should be taken for courseware development

The tasks of the LEIT may be analogous to the tasks of software requirements engineering [38]:

- Establish the vision and scope of the product
- Find, listen to, and understand the requirements of the customer
- Document, prioritize, and validate the requirements
- Produce a software requirements specification

In 1998, LEIT identified only eleven universities offering some relevant program in localization, software internationalization, or globalization, of which only three universities had an academic software internationalization course [3]. LEIT concluded that since most computer science curricula are already rigorous, universities find “little incentive to add anything” [3]. Although the ACM and IEEE Computing Curriculum stated that it was necessary to address “the need to develop implementation models that are international in scope and could be practiced in universities around the world,” ACM has no provision for internationalization training [1] [3]. However, some universities state that they “might focus on internationalization issues if they were convinced that they these would help their students find jobs” [3].

Among the universities identified by LEIT, the courseware was dependent on the instructor who has developed the materials [3]. Although a typical course involved presentations, handouts, and photocopied articles, the materials and syllabus were “neither standardized, nor easily portable” for instructors [3]. Given this, there is a need for a “standard set of skills and knowledge that could be taught by any knowledgeable instructor” [3]. A syllabus, defined as an outline course of study, could only scratch the surface of software internationalization with no guarantee that it satisfies industry requirements [29]. The requirement is to develop a framework, defined as an underlying set of ideas, which can be taught by other instructors and also addresses relevant industry requirements of today [57].

LEIT concluded that although there were education resources readily available, “the set of given skills that are taught at any one location varies tremendously” [3]. Therefore, the LEIT’s intermediate plan of action is the development of a model curricula or a framework that “would provide a standard set of skills and knowledge that could be taught by any knowledgeable instructor” [3]. Without loss of generality, the term “framework” in this thesis will be known as “a standard set of skills and knowledge that could be taught by any knowledgeable instructor” [3]. The requirement of the framework would satisfy various attributes of an excellent requirement that are complete, correct, feasible, necessary, prioritized, unambiguous, and verifiable [38]. The framework involves portability, allowing “the materials could be easily transferred to and applied by instructors who have not developed the materials themselves” [3]. However, LEIT stated that such a framework “would also need to be based on educational consensus and the approval of business contacts who could affirm that the skills and knowledge taught met real demands both from an educational perspective and from a business perspective” [3].

4.2. LEIT's Market Needs Survey

To affirm the requirement of providing students with knowledge of software internationalization from an educational and business perspective, LEIT conducted a market needs survey [3]. As did the survey about requirements, the market needs survey involved LISA members and other industry professionals “to determine how important training in various areas were felt” for upper managers, project managers, software engineers, translators, and technical writers [3]. The survey asked “respondents to indicate how important previous education in certain job skills is on a scale of 1 (highest priority)—5 (lowest priority)” [3]. The survey had a sample of 53 individuals, with an average score of 2.22, indicating that “most of these skills are considered to be part of the mandatory experience required for these jobs” [3]. The participants felt “that education prior to beginning a job is viewed as extremely important” [3]. The survey concluded that there is a clear gap between what the industry would like to see in education and what is actually available [3]. This clear gap indicates that the requirement is incomplete, infeasible, and necessary to the industry customers.

4.3. Case studies evaluating students' perceptions

To determine whether students are more aware of issues in software internationalization today than a decade ago, a simple case study was conducted to measure the students' perceptions at Cal Poly. The results and details are found in Appendix D and Appendix E.

Appendix D includes the results of a survey questionnaire on students' perceptions of software internationalization in evaluating whether or not students believe that it is a familiar topic, if it will be important to their future careers; is easy to learn, implement and deploy, and is easy to fix, maintain and debug. After analyzing the data, the conclusion reached was that at least a majority of students felt software internationalization was unfamiliar; was not important to their future career; was easy to learn, implement, and deploy; and was easy to fix, maintain, and debug. Although the survey was subjective, one important question can be drawn from this case study:

Can a majority of students who are unfamiliar with internationalization accurately determine whether or not internationalization is important to their future career, is easy to learn, implement, and deploy, and is easy to fix, maintain and debug?

Appendix E includes the results of a survey questionnaire on whether the exposure to internationalization activities changed the students' perceptions of software internationalization. Measuring

the students' perceptions would help answer whether software internationalization was more familiar to them, more important to their career, should be taught more at Cal Poly, is harder to learn, is harder to implement and deploy, and is harder to fix and maintain. The survey questionnaire was given before and after the internationalization laboratory assignment. Using a nonparametric one-sample signed test on the changes of students' perceptions, the conclusion reached was that there was little effect on their perceptions except that students were more familiar with software internationalization after the internationalization laboratory assignment. Like Appendix D, the conclusion reached from the survey analyzed in Appendix E that at least a majority of students felt software internationalization was unfamiliar before the internationalization laboratory assignment.

4.4. Research question

Even a decade after the LEIT report was produced, students still have little or no knowledge about software internationalization. Although LEIT reported only three of the eleven identified universities had an academic software internationalization course, a brief search on Google revealed that only seven colleges or universities are covering software internationalization today [3]. The colleges and universities are Austin Community College [23], Chico State University [24], Elon University [25], Queensland University of Technology [11], Rensselaer Polytechnic Institute [26], San Diego State University [27], and University of Mississippi [28]. This implies that the set of requirements and suggestions made by LEIT had little or no impact in encouraging universities to expand and prepare students with a software internationalization background.

Therefore, it is worth asking the question “*Is there an adequate framework for software internationalization?*” We can conclude that there is not one. Like a software requirements specification document, an adequate framework must be complete, consistent, modifiable, and traceable [38]. To continue the work of LEIT, I am determined to create an adequate framework that meets the requirements of our industry customers for software internationalization. The requirement of the framework is to “promote communication between academia and industry ... that could serve as a common reference point in discussions” [3]. LEIT acknowledges that although instructors “would still need to develop lesson and teaching plans and gather teaching materials, ... this task would be much easier since there would be a clear outline of instruction” [3]. The general hope in utilizing this framework is to provide a portable and

standardized set of requirements for computer science and software engineering programs to teach future graduates.

To develop the framework for instruction, this thesis will consider the needs of the customers and their preference to hire students who have knowledge of software internationalization. Chapter 5 and Chapter 6 will discuss an industry survey questionnaire that will be used to validate the underlying set of requirements discussed in Chapter 3. After gathering the feedback and analyzing the data, Chapter 7 will discuss the final framework that is not only supported by various reviews of literature but also validated against industry requirements. The framework will be the resulting software requirements specification document, detailing the requirements important to the customers in the industry today.

CHAPTER 5. INDUSTRY SURVEY QUESTIONNAIRE DESIGN

5.1. Foreword

The previous chapters highlight the importance of software internationalization and its present state in education. Using the support of numerous papers, Chapter 3 discussed an underlying set of requirements that embody the state of educating software internationalization. Although only an underlying set of requirements that can be used as a framework was presented in Chapter 3, “there is no real guarantee that any particular course teaches the particular skills and knowledge that industry needs” [3]. In Chapter 4, the conclusion reached was that no adequate framework meets the requirements of our industry customers for software internationalization. As analyzed in Chapter 4, the LEIT conducted a market needs survey where the participants felt “that education prior to beginning a job is viewed as extremely important” [3]. The survey concluded that there is a clear gap between what the industry would like to see in education and what is actually available [3]. This clear gap indicates that the requirement to satisfy our industry customer needs is incomplete, infeasible, and necessary.

Continuing the research of LEIT, Chapter 5 and Chapter 6 will discuss an industry survey questionnaire used to validate the underlying set of requirements discussed in Chapter 3. After gathering the feedback and analyzing the data, Chapter 7 will discuss the final framework that is not only supported by an extensive literature review, but also validated against industry requirements. The framework helps us to understand what requirements are important to the industry today.

Unless otherwise noted, the industry survey documents and data can be found in Appendix H.

5.2. The plan

In this thesis, I will attempt to answer the research question by presenting a feasible and cost-effective method by developing and providing a framework that has been validated by the industry. In Chapter 3, we derived the proposed requirements for educating students in software internationalization. The general hope in utilizing this framework is to provide a portable and standardized set of requirements for computer science and software engineering programs to teach future graduates. The next step is to know which of the proposed requirements meet with the requirements of the industry. To do this, we brought the proposed requirements directly to the industry representatives themselves.

To prove the completeness, consistency, modifiability, and traceability of the framework, the thesis presents responses gathered from the industry, as well as how the responses will be used to validate the proposed requirement in Chapter 3. Chapter 5 includes a discussion on the design of the industry survey questionnaire. Chapter 6 discusses the statistical analysis of the data, including the outcomes of the responses and the hypothesis tests. The expected result from the industry survey questionnaire is that:

- The industry will strongly agree with a certain requirement more than another requirement
- The industry will not feel neutral about any of the proposed requirements
- At least a majority of people will feel strongly agree or strongly disagree with one or more proposed requirements

The results from Chapter 6, used to validate the proposed requirements, will document and prioritize the requirements for the final framework. The final framework, which will adhere to the industry needs in Chapter 7, will therefore be an adequate framework that is complete, consistent, modifiable, and traceable.

5.3. Goals

We need to ensure that the requirement of our industry customers, which is to hire computer science and software engineering students who are knowledgeable in software internationalization, is complete and feasible. Using a similar approach as the LEIT used in their survey, we will design our own “Market Needs Survey.” The industry survey questionnaire in this thesis will help validate and prioritize the importance of the underlying set of requirements to the industry today. The underlying set of requirements can be used as a framework for software internationalization to “promote communication between academia and industry ... that could serve as a common reference point in discussions” [3]. Indeed, LEIT acknowledges that although instructors “would still need to develop lesson and teaching plans and gather teaching materials, ... this task would be much easier since there would be a clear outline of instruction” [3]. The goal of the industry survey questionnaire is to provide both a framework that has been validated by the industry to be both a portable and a standardized set of requirements for computer science and software engineering programs to teach future graduates.

5.4. Summary of hypotheses

In the industry survey, there were 24 statements that could be answered on a five-point Likert scale, with options strongly agree; somewhat agree; neither; somewhat disagree and strongly disagree. Likert scale responses are used to provide a sense of how the respondent feels about a particular statement or question, and the strength of that feeling [58]. To simplify the statements and maintain readability, Table 2 provides abbreviations pertaining to each of the 24 statements used in the industry survey questionnaire. In addition, Table 3 provides traceability for each of the statements used in the industry survey questionnaire to the particular sections discussed in Chapter 3.

Abbreviation	Statements in the industry survey questionnaire starting with <i>"In order for computer science and software engineering graduates to be prepared for industry, students must ..."</i>
Learn	... learn software internationalization through lecture, reading, and writing
Apply	... apply software internationalization through laboratory assignments, projects, and activities
Integrate	... learn about software internationalization by integrating the subject into existing courses
Course	... learn about software internationalization by taking an entire course devoted to the subject
Abroad	... learn about software internationalization by studying abroad
Technology-Aware	... be aware of possible technologies and tools for creating software that supports internationalization
Technology-Apply	... apply the possible technologies and tools for creating software that supports internationalization
Definitions	... define and distinguish between software internationalization, localization, and globalization
Standards	... define and distinguish various industry standards on software internationalization (ISO, W3C, LISA)
Colors	... be aware of how particular colors are viewed differently among various countries
Visual	... be aware of how particular visual elements are viewed differently among various countries
Noise	... be aware of how particular noise elements are viewed differently among various countries
Laws	... be aware of how laws of various countries can affect the software that supports internationalization
Translation-Aware	... be aware of possible technologies and tools used for language translation and text representation
Translation-Apply	... apply the possible technologies and tools used for language translation and text representation
Direction-Aware	... be aware the possible technologies and tools to represent the direction of text in various languages
Direction-Apply	... apply the possible technologies and tools to represent the direction of text in various languages
Collation-Aware	... be aware of how text is sorted in various languages
Collation-Apply	... apply the possible technologies and tools to sort text in various languages
Dates-Aware	... be aware of how dates and times are used in various cultures and countries
Dates-Apply	... apply the possible technologies and tools to decipher the date and time for various cultures and countries
Currency-Aware	... be aware of how the number/currency systems are different for various cultures and countries
Currency-Apply	... apply the possible technologies for the number/currency systems for various cultures and countries
Ethics	... be aware of the social responsibilities and ethics of producing software that supports internationalization

Table 2 - Abbreviations for the statements used in the industry survey questionnaire.

Abbreviation	Tracing to the Sections from Chapter 3
Learn	Section 3.2
Apply	Section 3.2, 3.4
Integrate	Section 3.2.1
Course	Section 3.2.2
Abroad	Section 3.2.3
Technology-Aware	Section 3.4
Technology-Apply	Section 3.4
Definitions	Section 3.3
Standards	Section 3.3
Colors	Section 3.7.1
Visual	Section 3.7.2
Noise	Section 3.7.3
Laws	Section 3.7.4
Translation-Aware	Section 3.5
Translation-Apply	Section 3.4, 3.5
Direction-Aware	Section 3.5.1
Direction-Apply	Section 3.4, 3.5.1
Collation-Aware	Section 3.5.2
Collation-Apply	Section 3.4, 3.5.2
Dates-Aware	Section 3.5.3
Dates-Apply	Section 3.4, 3.5.3
Currency-Aware	Section 3.5.4
Currency-Apply	Section 3.4, 3.5.4
Ethics	Section 3.6

Table 3 – Tracing each of the abbreviations to the sections in Chapter 3

5.5. Method of data observations and hypothesis tests

After summarizing the hypotheses, this section will provide a brief background in statistics on the importance of a p-value and significance level, discuss how the data will be observed and discussed, explain how to check and validate the sample size of the data, and describe the number of hypothesis tests that will be conducted to validate the data. Statistics is “the science of collecting, analyzing, presenting, and interpreting data” and hypothesis testing “is a form of statistical inference that uses data from a sample to draw conclusions about a population parameter” [59]. Using statistics, a branch of mathematics, to convince the reader that the data is valid is much more preferred over observational data [59].

5.5.1. P-value and significance level

In order to provide proper inferential analysis of statistical data, one must be able to describe the p-value, or an observed significance level, on a particular hypothesis test [60]. A p-value is the smallest significance level at which the null hypothesis would be rejected [61]. A significance level, usually denoted as α , is the largest value that can be tolerated, defines the rejection region, and is chosen depending on the “seriousness of a type I error” [61]. A type I error is when the null hypothesis is rejected when in fact it is

true [61]. The significance level, α , equals the probability of a type I error [60]. A type II error is when the null hypothesis fails to be rejected when in fact it is false [61]. Since a type I error is usually more serious than a type II error, a convention used is that the more serious the type I error, the smaller the significance level should be [61]. Once the p-value and significance level are determined, the following conclusion can be made about the null hypothesis [61]:

- If the p-value $\leq \alpha$, reject the null hypothesis.
- If the p-value $> \alpha$, fail to reject the null hypothesis.

In some studies, it is possible to have several simultaneous tests or hypotheses. If this is the case, the overall type I error rate would increase [60] [61]. To appropriately compare with the p-value, a suggestion is to use the Bonferroni Adjustment as shown in Figure 5 [60].

$\frac{\alpha}{k}$ <p>α is the significance level k is the number of simultaneous tests</p>

Figure 5 - Bonferroni Adjustment

In regards to the industry survey, k would be 24 for the number of hypotheses, according to Figure 5. This adjustment to the individual significance level for each question conservatively ensures that the overall significance level, the probability of falsely rejecting at least one of the hypotheses, is at most α .

5.5.2. Check for sample size

Before conducting statistical analysis, one of the most important steps is to ensure that the sample size is sufficient enough for hypothesis testing for each of the statements in the industry survey questionnaire [62]. Checking for the sample size before conducting any statistical analysis help to validate the data and determine whether more industry representatives are needed [60]. Discussion and details of the calculation provided by Dr. Karen McGaughey, Assistant Professor of the Statistics Department at Cal Poly, can be found in Appendix H.

The goal of this section is to determine a target sample size value, n , during the planning stages of the industry survey questionnaire. In order to understand this process, the power and effect size will be discussed in the following paragraphs [62].

First, the calculation of the effect size pertaining to the industry survey questionnaire is shown in Figure 6. The null hypothesis states that the respondents answered each statement equally among the values

of strongly agree, somewhat agree, neither, somewhat disagree, and strongly disagree. The effect size value, w , tells us “how different the alternative proportions are from the null proportions” [62]. Note that as the effect size value, w , gets larger, the sample size required to be sufficient get smaller [62]. A small effect size value is suggested to be when $w=0.1$, while a medium effect size value is suggested to be when $w=0.3$, and a large effect size is suggested at $w=0.5$ [62].

Null Hypothesis:	$H_0: p_{\text{Strongly Agree}} = p_{\text{Somewhat Agree}} = p_{\text{Neither}} = p_{\text{Somewhat Disagree}} = p_{\text{Strongly Disagree}} = 0.2$
Alternative Hypothesis:	$H_a: \text{At least one } p_i \text{ does not equal } 0.2$
Effect Size Value:	$w = \sqrt{\sum_{i=\text{each category}} \frac{(\text{expected}-\text{observed})^2}{\text{expected}}} = \sqrt{\sum_{i=\text{each category}} \frac{(.2-\text{observed})^2}{.2}}$ $w = \sqrt{\frac{\left(.2 - \widehat{p_{\text{Strongly Agree}}}\right)^2 + \left(.2 - \widehat{p_{\text{Somewhat Agree}}}\right)^2 + \left(.2 - \widehat{p_{\text{Neither}}}\right)^2 + \left(.2 - \widehat{p_{\text{Somewhat Disagree}}}\right)^2 + \left(.2 - \widehat{p_{\text{Strongly Disagree}}}\right)^2}{.2}}$

Figure 6 - Test for sample size with effect size value.

Second, the power for a given sample size needs to be calculated. The power for a given sample size is important because it tells us the probability that we reject the null hypothesis when in fact the null hypothesis is false [62]. Therefore, as the power for a given sample size gets closer to the ideal of 1.0, the higher the probability we can reject the null hypothesis [62]. Similarly, as the sample size gets larger, the power gets closer to the ideal value of 1.0 [62]. The relationship between sample size and the effect size is shown with the non-central parameter value for a Chi-Square distribution, λ , as shown in Figure 7.

$\lambda = nw^2$ <p>where n is the sample size and w is the effect size value.</p>
--

Figure 7- Non-centrality parameter for a Chi-Square distribution.

Finally, to obtain a desirable power of 0.90 with four degrees of freedom, and a significance level of 0.05, the non-central parameter value for a Chi-Square distribution, λ , is 15 as shown in Figure 7 [62]. The value of 15 is discussed in Appendix H. With a little bit of algebra, our target sample size is calculated as shown in Figure 8.

$\frac{15}{w^2} = n$ <p>where n is the sample size and w is the effect size value.</p>
--

Figure 8 - Minimal sample size with the non-centrality parameter at 15.

Therefore, a small effect size requires a sample size of $n=1,500$, a medium effect size requires a sample size of $n=167$, and a large effect size requires a sample size of $n=60$.

My hypothesis is that the resulting effect size will be medium, since I doubt the respondents will answer each statement equally among the choices of strongly agree, somewhat agree, neither, somewhat disagree, and strongly disagree. Therefore, my target sample size to detect a medium effect size for the study will be of $n=167$ respondents.

5.5.3. Calculating the averages

Once the data have been gathered, calculating the averages of the responses is a simple way to draw preliminary conclusions. Using the Likert scale by assigning values of strongly agree with 5, somewhat agree with 4, neither with 3, somewhat disagree with 2, and strongly disagree with 1, we are able to calculate the averages for each of the statements. Using these calculated averages, we can sort the values and determine with which statements the industry representatives more strongly agree or more strongly disagree.

5.5.4. Test for equal proportions

The first hypothesis test in the industry survey questionnaire will measure whether each of the statements received responses with equal proportions as shown in Figure 9.

Null Hypothesis:	$H_0: p_{\text{Strongly Agree}} = p_{\text{Somewhat Agree}} = p_{\text{Neither}} = p_{\text{Somewhat Disagree}} = p_{\text{Strongly Disagree}} = 0.2$
Alternative Hypothesis:	$H_a: \text{At least one } p_i \text{ does not equal } 0.2$
Test Statistic Value:	$\chi^2 = \sum_{i=\text{each category}} \frac{(\text{observed} - \text{expected})^2}{\text{expected}} = \sum_{i=\text{each category}} \frac{(n_i - (n \cdot 0.2))^2}{(n \cdot 0.2)}$

Figure 9 - Test for equal proportions with Chi-Squared Goodness of Fit test.

The test will tell us whether the respondents answered a certain statement equally among strongly agree, somewhat agree, neither, somewhat disagree, and strongly disagree. In other words, do we have enough evidence to suggest that the choices were not equally likely? Do we have enough evidence to suggest that the values are far enough from 0.2 to show that at least one of the population proportions differs from 0.2? The test to “measure of the discrepancy between the observed numbers in the categories and the expected numbers,” when the null hypothesis is true, is called the Chi-Squared Goodness-of-Fit test [61].

5.5.5. Test for significant majority

The second hypothesis test in the industry survey questionnaire will measure whether each of the statements received responses with significant majority, at least 50 percent. Both sides of the Likert scale are statistically calculated for its validity. Using a simple one proportion z-test, this will tell us whether a significant majority of respondents answered each of the following:

- A significant majority overall agree as shown in Figure 10
- A significant majority strongly agree as shown in Figure 11
- A significant majority overall disagree as shown in Figure 12
- A significant majority strongly disagree as shown in Figure 13

Null Hypothesis:	$H_0: \left(p_{\text{Strongly Agree}} + p_{\text{Somewhat Agree}} \right) = 0.50$
Alternative Hypothesis:	$H_a: \left(p_{\text{Strongly Agree}} + p_{\text{Somewhat Agree}} \right) > 0.50$

Figure 10 - Test for significant majority with one proportion z-test on "Overall Agree."

Null Hypothesis:	$H_0: p_{\text{Strongly Agree}} = 0.50$
Alternative Hypothesis:	$H_a: p_{\text{Strongly Agree}} > 0.50$

Figure 11 - Test for significant majority with one proportion z-test on "Strongly Agree."

Null Hypothesis:	$H_0: \left(p_{\text{Strongly Disagree}} + p_{\text{Somewhat Disagree}} \right) = 0.50$
Alternative Hypothesis:	$H_a: \left(p_{\text{Strongly Disagree}} + p_{\text{Somewhat Disagree}} \right) > 0.50$

Figure 12 - Test for significant majority with one proportion z-test on "Overall Disagree."

Null Hypothesis:	$H_0: p_{\text{Strongly Disagree}} = 0.50$
Alternative Hypothesis:	$H_a: p_{\text{Strongly Disagree}} > 0.50$

Figure 13 - Test for significant majority with one proportion z-test on "Strongly Disagree."

5.6. Subjects

The subjects, or the customers, of the industry survey questionnaire were industry representatives stemming mostly from the software sector. Since software internationalization is mostly applicable to the software sector, it would make sense to gather the industry representatives from that arena. About a thousand companies were individually contacted with details about the response rate and data filtering found in Chapter 6. Major organizations such as ACM, LISA, and the Unicode Consortium were also contacted through their member mailing lists. The LISA Director, Michael Anobile, personally contacted his colleagues and advocated support for the industry survey questionnaire. The Computer Science Department Chair at Cal Poly, Dr. Ignatios Vakalis, personally contacted his colleagues and advocated his support to the members of the Industrial Advisory Board.

Demographics were used to help understand the diversity of subjects among multiple companies, their locations, the number of years they have worked in the industry, their titles, their highest level of education, and the group within the company. The demographic analysis will help evaluate whether we have a representative sample, draw other conclusions, and aid any future study. My goals from the demographics were to include:

- Respondents from as many participating companies as possible.
- Respondents from as many locations around the world as possible.
- Respondents from as many years working in the industry as possible.
- Respondents from various titles from executives and below as possible.
- Respondents outside the globalization, internationalization, or localization team as possible.
- Respondents from all levels of education as possible.
- Respondents from all groups of the company as possible.

5.7. Survey design and procedure

5.7.1. Tools for gathering data

A number of options were considered to gather responses from the subjects, including mailing or calling the industry representatives. However, the most cost-effective and quickest method to gather responses from the industry representatives was to contact each company by e-mail and provide a link to a web survey. A web survey is when a respondent is asked to visit a web site and respond to a survey questionnaire [58]. Since the subjects in question have a high rate of internet use, web surveys make the process fast and effective [58]. The advantage of a web survey includes the flexibility of designing a questionnaire with logic and graphics where responses are recorded directly into a database and are reported quickly [58]. The tool that was used for the web survey was SurveyMonkey. SurveyMonkey is an online survey tool that enables users to create surveys quickly and easily [63]. SurveyMonkey has been serving thousands of people since 1999 “including more than 80% of the Fortune 100” [63].

5.7.2. Human Subjects Committee and consent web page

When the industry representatives access the web survey, the first web page is a page of consent. Respondents are informed on the first web page that they are going to take part in an industry survey questionnaire about software internationalization. The consent form and survey questionnaire can be found

in Appendix H and were reviewed and approved by Dr. Steve Davis, Chair of the Cal Poly Human Subjects Committee. The page of consent outlined the description of the study, the potential benefits and risks of the study, how long the survey questionnaire would take, how the responses would be kept confidential, how to contact the researchers, and how to contact the Human Subjects Committee. When the subject pressed the button marked “Next” he/she indicated willingness to participate in the survey questionnaire. However, subjects were able to choose to opt out by not answering any of the survey questions or by closing the web page.

5.7.3. Survey questionnaire

The third and fourth web pages are the survey questionnaire. A screenshot of the third and fourth web pages of the survey questionnaire can be found in Appendix H. Each page first defines software internationalization and localization, then details the statements used for the industry survey questionnaire. Software internationalization is defined as:

...the process of producing an application that can be localized for a particular country without making any changes to the program code.

Localization is defined as:

...the process of adapting software for a particular geographical region.

Following the definitions, the statements for the survey questionnaire are summarized in Table 2 in Section 5.4. Each statement can be answered on a five-point Likert scale, with options from strongly agree, to somewhat agree, to neither, to somewhat disagree, to strongly disagree. The view of the survey questionnaire is shown as a matrix, where the survey questionnaire statements are the rows and the five-point Likert scale answer options are the columns.

5.7.4. Procedure

When contacting the representatives to answer the industry survey, there were only two rounds of e-mails sent to each company. The first e-mail was sent to inform the company of the research and the industry survey. The first e-mail stated that it was important for universities to appropriately meet with the industry’s need for software internationalization and that this will benefit the company if they chose to participate. Contact information was also provided for industry representatives if they had questions and/or if they wanted to see the results of the survey. The second e-mail was the last and final reminder asking the

industry representatives to answer the industry survey. Unlike the first e-mail, the second e-mail covered the progress of the industry survey questionnaire, such as having at least 200 responses.

The web survey was a total of five web pages; screenshots can be found in Appendix H. The first web page is the page of consent as discussed in Section 5.7.2. The second through fourth page displays the following statement to reduce the number of possible user errors:

The following questions will be used primarily for comparative analysis with others in the industry. Although you may omit any items that you prefer not to answer, you are asked to answer all of the questions as honestly and truthfully as you can. If you do not understand a question, please omit the item.

The second web page asks the industry representatives about their demographics as discussed in Section 5.6. The third and fourth web page is the survey questionnaire as discussed in Section 5.7.3. The fifth and final web page displays a thank you page for completing the survey questionnaire.

CHAPTER 6. INDUSTRY SURVEY QUESTIONNAIRE ANALYSIS

This chapter will discuss the analysis of the data gathered from the industry survey. Unless otherwise noted, all of the industry survey documents and data are found in Appendix H.

6.1. Response rate

Approximately one thousand companies were successfully contacted for the industry survey questionnaire by e-mail. A successful contact is when an e-mail was delivered properly and not rejected by the server. About thirteen companies replied that they chose not to participate in the survey due to reasons such as the size of the company or that the survey was irrelevant. The procedures of how the companies were contacted are discussed in Section 5.6 and 5.7. In the end, there were a total of 361 recorded responses. A small percentage of the recorded responses came from the same company or organization in the industry. Some of the recorded responses were filtered for two relevant reasons:

- The responses to the third and fourth web pages containing the industry survey questionnaire were left blank
- The responses indicated that the subject was clearly not part of the industry (for example, responses from subjects in academia such as a professor or a student).

After reviewing and filtering the recorded responses, there were a total of 278 valid responses. Unless otherwise noted, the valid responses were used in this analysis.

6.2. Demographics

Demographic-based questions were asked to help understand the diversity of subjects among multiple companies, their locations, the number of years they have worked in the industry, their titles, their highest level of education, and the group within the company. The demographics are described in greater detail in Section 5.6.

6.2.1. Participating companies

Of the valid responses, a total of 148 distinct companies included the name of the company. A full list of participating companies can be found in Appendix H. Some well-known organizations that participated in the industry survey questionnaire include Adobe, AMD, Amgen, ACM, Cisco, eBay, Expedia, Google, HP, IBM, Microsoft, NVIDIA, PayPal, Qualcomm, Raytheon, Salesforce, Siemens, Sun, Symantec, and Yahoo!.

6.2.2. Locations

Of the valid responses, a total of 126 responses indicated the work location. A full list of locations can be found in Appendix H. My goal to include as many people from around the world as possible, since software was originally developed by Americans for Americans, was fulfilled [5]. The following pie chart, Figure 14, shows the distribution of respondents located within the continents of North America, Europe, Asia, and South America. A highlight is that about 40 percent of the respondents were outside North America.

Figure 14 - Industry responses by location separated by continent.

6.2.3. Years working in the industry

Of the valid responses, a total of 236 responses indicated the number of years they have worked in the industry. A full break down of each year can be found in Appendix H. My goal to include a wide distribution of respondents with various years of industry experience, since an entry-level employee may have different insights from a principal-level employee, was met. This holds especially true if one has been working with software internationalization for a number of years. A histogram summarizing the distribution of respondents among the number of years they have worked in the industry is summarized in Figure 15. A highlight of this finding is that the majority of respondents have worked at least 10 years in the industry.

Figure 15 - Industry responses to the number of years working in the industry.

6.2.4. Titles

Of the valid responses, a total of 227 responses indicated the individual's title. A full list of the titles extracted from the database can be found in Appendix H. This met my goal to include a diverse number of respondents serving in various positions from executives down. My goal to receive numerous responses outside the globalization, internationalization, or localization teams, was also met. Highlights included about four Presidents, five Chief Executive Officers, ten Chief Technology Officers, fifteen Vice Presidents, and over fifty Managers, Directors, or Principals.

6.2.5. Highest level of education

Of the valid responses, a total of 263 responses indicated the individual's highest level of education. A bar chart summarizing the distribution of respondents among the highest level of education is summarized in Figure 16. My goal to include respondents from all levels of education was fulfilled. There were 34 respondents with a Doctor of Philosophy degree, which was highly beneficial to the analysis because it is the highest level of university degree awarded to someone "who has successfully completed a lengthy piece of original research" [64]. A highlight is that the majority of respondents either had a bachelor's degree or master's degree.

Figure 16 - Industry responses to "Highest Level of Education."

6.2.6. Group

Of the valid responses, a total of 268 responses indicated a group in which they work. A bar chart summarizing the distribution of respondents among the highest group is summarized in Figure 17. A highlight is that the majority of respondents were either in the Internationalization and Localization group or the Corporate and Executive group, followed by Development, Engineering, and Research group as well as the Manager, and Supervisor group. My goal to include respondents from all groups of the company was satisfied. This demographic also helped to illustrate the great number of information; for instance, a respondent could list themselves as a "Software Engineer" who works in the "Internationalization, Localization" group.

Figure 17 - Industry responses to "Group."

6.3. Calculating the averages

This section refers to Section 5.5.3. The results for calculating the averages of the Likert scale values of each of the survey industry statements are found in following table, Table 4.

Abbreviation	Average
Technology-Aware	4.4945
Dates-Aware	4.4479
Currency-Aware	4.3822
Direction-Aware	4.3668
Definitions	4.3248
Apply	4.2924
Collation-Aware	4.2654
Translation-Aware	4.2269
Technology-Apply	4.1465
Learn	4.1087
Integrate	4.1051
Dates-Apply	4.004
Currency-Apply	3.996
Laws	3.9783
Direction-Apply	3.9767
Ethics	3.9538
Collation-Apply	3.9186
Visual	3.8442
Standards	3.8309
Translation-Apply	3.8062
Colors	3.6836
Noise	3.6460
Course	3.2935
Abroad	2.5474

Table 4 - Averages of the Likert scale values of each of the statements

Using the Likert scale by assigning values of strongly agree with 5, somewhat agree with 4, neither with 3, somewhat disagree with 2, and strongly disagree with 1, we are able to calculate the averages for each of the statements. Based on the averages, only two statements did not tend toward strongly agree or somewhat agree by the industry responses (an average below 3.5). These two statements include:

- “In order for computer science and software engineering graduates to be prepared for industry, students must learn about software internationalization by taking an entire course devoted to the subject,” with an average score of 3.294.
- “In order for computer science and software engineering graduates to be prepared for industry, students must learn about software internationalization by studying abroad,” with an average score of 2.547.

The top five statements that lean towards strongly or somewhat agree by the industry responses were:

- “In order for computer science and software engineering graduates to be prepared for industry, students must be aware of possible technologies and tools used for language translation and text representation” by an average score of 4.495.
- “In order for computer science and software engineering graduates to be prepared for industry, students must be aware of how dates and times are used in various cultures and countries” by an average score of 4.448.
- “In order for computer science and software engineering graduates to be prepared for industry, students must be aware of how the number/currency systems are different for various cultures and countries” by an average score of 4.382.
- “In order for computer science and software engineering graduates to be prepared for industry, students must be aware the possible technologies and tools to represent the direction of text in various languages” by an average score of 4.367.
- “In order for computer science and software engineering graduates to be prepared for industry, students must define and distinguish between software internationalization, localization, and globalization” by an average score of 4.325.

By brief observation, the industry overall agrees that students should be aware of a variety of issues in regard to software internationalization.

6.4. Check for sample size

This section refers to Section 5.5.2. The results for checking the sample size can be found in Table 17 in Appendix H. Among the statements for the industry survey questionnaire, all met more than the target sample size of $n=167$ for a medium effect size value, of $w=0.3$. That implies that there is sufficient power to detect this effect size; therefore, the sample size is sufficiently large to carry out a statistical hypothesis test for each statement.

6.5. Test for equal proportions

This section refers to Section 5.5.3. The results for the test for equal proportions can be found in Table 22 in Appendix H. Among all of the statements for the industry survey questionnaire, all were

answered significantly different from equal proportions. This suggests that for each statement, more or less than 20 percent of the respondents answered leaning towards one or more Likert scale value.

6.6. Test for significant majority

This section refers to Section 5.5.5. The results for the test for significant majority can be found in Table 23 and Table 24 in Appendix H. Table 5 displays a summary of the results.

Abbreviation	Result with $\alpha = .05$	Result with $\alpha = (.05/24) \approx .002$
Learn	Overall Agree	Overall Agree
Apply	Overall Agree	Overall Agree
Integrate	Overall Agree	Overall Agree
Course	Neither	Neither
Abroad	Neither	Neither
Technology-Aware	Strongly Agree	Overall Agree
Technology-Apply	Overall Agree	Overall Agree
Definitions	Overall Agree	Overall Agree
Standards	Overall Agree	Overall Agree
Colors	Overall Agree	Overall Agree
Visual	Overall Agree	Overall Agree
Noise	Overall Agree	Neither
Laws	Overall Agree	Overall Agree
Translation-Aware	Overall Agree	Overall Agree
Translation-Apply	Overall Agree	Overall Agree
Direction-Aware	Overall Agree	Overall Agree
Direction-Apply	Overall Agree	Overall Agree
Collation-Aware	Overall Agree	Overall Agree
Collation-Apply	Overall Agree	Overall Agree
Dates-Aware	Strongly Agree	Overall Agree
Dates-Apply	Overall Agree	Overall Agree
Currency-Aware	Overall Agree	Overall Agree
Currency-Apply	Overall Agree	Overall Agree
Ethics	Overall Agree	Overall Agree

Table 5 – Summary of results from test for significant majority.

In Table 5, an entry of “Overall Agree” means a significant majority overall agree with the statement. An entry of “Strongly Agree” means a significant majority strongly agree with the statement. An entry of “Neither” means no significantly majority overall agree and no significantly majority overall disagree with a statement. None of the entries had “Overall Disagree” or “Strongly Disagree,” which would have indicated a significant majority overall disagree or strongly disagree with the statement, respectively.

Table 5 shows the results compared between two significance levels. The first significance level of $\alpha = .05$ is traditionally used in statistics [61] [60]. The second significance level of $\alpha = .05/24$ uses a Bonferroni Adjustment in reference to the 24 simultaneous statements used in the industry survey questionnaire as discussed in Section 5.5.1. Since 24 statements increases the probability of a type I error, a

more conservative level to use for any conclusion is $\alpha \approx .002$. Recall from Section 5.5.5 that a significance level, usually denoted as α , is the largest value that can be tolerated while finding a rejection region and is chosen depending on the “seriousness of a type I error” [61]. A type I error is when the null hypothesis is rejected when in fact it is true [61].

There are three highlights from this hypothesis test. First, a significant majority strongly agree or overall agree with 22 of 24 statements when the significance level is $\alpha = .05$. When the significance level decreased to $\alpha \approx .002$, a significant majority overall agree with 21 of 24 statements. Second, there are three different conclusions in regard to the statements from a significance level of $\alpha = .05$ to a significance level of $\alpha \approx .002$:

- The statement “In order for computer science and software engineering graduates to be prepared for industry, students must be aware of possible technologies and tools for creating software that supports internationalization,” went with a majority of respondents from strongly agree to overall agree.
- The statement “In order for computer science and software engineering graduates to be prepared for industry, students must be aware of how particular noise elements are viewed differently among various countries,” went with a majority of respondents from overall agree to neither.
- The statement “In order for computer science and software engineering graduates to be prepared for industry, students must be aware of how dates and times are used in various cultures and countries,” went with a majority of respondents from strongly agree to overall agree.

Third, there was no significant majority that agrees or disagrees with the following three statements:

- The statement “In order for computer science and software engineering graduates to be prepared for industry, students must learn about software internationalization by taking an entire course devoted to the subject,” when the significance level was $\alpha = .05$ or $\alpha \approx .002$.
- The statement “In order for computer science and software engineering graduates to be prepared for industry, students must learn about software internationalization by studying abroad,” when the significance level was $\alpha = .05$ or $\alpha \approx .002$.

- The statement “In order for computer science and software engineering graduates to be prepared for industry, students must be aware of how particular noise elements are viewed differently among various countries” when the significance level was $\alpha \approx .002$.

After discussing the analysis of the data gathered from the industry survey, Chapter 7 uncovers the resulting framework for software internationalization validated against industry requirements.

CHAPTER 7. FRAMEWORK FOR SOFTWARE INTERNATIONALIZATION VALIDATED AGAINST INDUSTRY REQUIREMENTS

After gathering and analyzing the responses from the industry survey questionnaire, Chapter 7 presents the resulting framework for software internationalization validated against industry requirements. The framework only contains requirements that a majority of respondents strongly agree or overall agree with. The requirements in the framework are then prioritized by averages calculated from Section 6.3.

The first part of the framework is the architectural educational model with which the majority of respondents from the industry overall agree. The architectural educational models were discussed in Section 3.2. The majority of respondents agree the following requirement as the architectural educational model:

In order for computer science and software engineering graduates to be prepared for industry, students must learn about software internationalization by integrating the subject into existing courses.

As mentioned in Chapter 3, most computing programs employ the model of integrating software internationalization into existing computing courses [1]. The ACM and IEEE Computing Curriculum 2001 identified certain courses using this educational architectural model, including ethics or social issues courses, capstone courses, and team-based courses [1]. As for the other two other architectural educational models, there was no consensus; the majority of respondents from the industry either overall agrees or disagrees with the following requirements:

- In order for computer science and software engineering graduates to be prepared for industry, students must learn about software internationalization by taking an entire course devoted to the subject.
- In order for computer science and software engineering graduates to be prepared for industry, students must learn about software internationalization by studying abroad.

After validating the architectural educational model, the overarching requirement is still broken down into a number of other requirements that embody the topic of software internationalization.

The second part of the framework is the requirements that validate the overarching requirement of the industry. The requirements were discussed in Sections 3.3, 3.4, 3.5, 3.6, and 3.7. Table 6 lists, in order of importance, the requirements with which the majority of respondents from the industry overall agree:

Order of importance	Statements in the industry survey questionnaire starting with <i>"In order for computer science and software engineering graduates to be prepared for industry, students must ..."</i>
1	... be aware of possible technologies and tools used for language translation and text representation (Section 3.5)
2	... be aware of how dates and times are used in various cultures and countries (Section 3.5.3)
3	... be aware of how the number/currency systems are different for various cultures and countries (Section 3.5.4)
4	... be aware the possible technologies and tools to represent the direction of text in various languages (Section 3.4)
5	... define and distinguish between software internationalization, localization, and globalization (Section 3.3)
6	... apply software internationalization through laboratory assignments, projects, and activities (Section 3.2, 3.4)
7	... be aware of how text is sorted in various languages (Section 3.5.2)
8	... be aware of possible technologies and tools used for language translation and text representation (Section 3.5)
9	... apply the possible technologies and tools for creating software that supports internationalization (Section 3.4)
10	... learn software internationalization through lecture, reading, and writing (Section 3.2)
11	... apply the possible technologies and tools to decipher the date and time for various cultures and countries (Section 3.4, 3.5.3)
12	... apply the possible technologies for the number/currency systems for various cultures and countries (Section 3.4, 3.5.4)
13	... be aware of how laws of various countries can affect the software that supports internationalization (Section 3.7.4)
14	... apply the possible technologies and tools to represent the direction of text in various languages (Section 3.4, 3.5.1)
15	... be aware of the social responsibilities and ethics of producing software that supports internationalization (Section 3.6)
16	... apply the possible technologies and tools to sort text in various languages (Section 3.4, 3.5.2)
17	... be aware of how particular visual elements are viewed differently among various countries (Section 3.7.2)
18	... define and distinguish various industry standards on software internationalization (ISO, W3C, LISA) (Section 3.3)
19	... apply the possible technologies and tools used for language translation and text representation (Section 3.4, 3.5)
20	... be aware of how particular colors are viewed differently among various countries (Section 3.7.1)

Table 6 - Framework of requirements the industry overall agree listed in order of importance.

Table 6 shows how the requirements developed from the literature review with which the industry most strongly agree or overall agree. Unexpectedly, a majority of respondents from the industry did not strongly disagree or overall disagree with any of the requirements. The following statement was the only requirement with which a majority of respondents from the industry neither agree or disagree:

- In order for computer science and software engineering graduates to be prepared for industry, students must be aware of how particular noise elements are viewed differently among various countries

Referring to the requirements in Table 6, one highlight is that the level of importance was much higher for students being aware of aspects of software internationalization than for students to apply aspects of software internationalization. The only exception to the awareness is that the sixth most agreed requirement by a majority of respondents is:

In order for computer science and software engineering graduates to be prepared for industry, students must apply software internationalization through laboratory assignments, projects, and activities.

The overall conclusion of this thesis is:

The majority of respondents from the industry overall agree that in order for computer science and software engineering graduates to be prepared for industry, universities should teach students software internationalization by integrating the various requirements from Table 6, in order of importance, into current existing courses.

CHAPTER 8. CONCLUSION

8.1. Summary

In Chapter 2, we saw how software internationalization is important. Chapter 3 presented the state of educating software internationalization by discussing an underlying set of requirements embodying it. Chapter 3 concluded that although the set of requirements can be used as a framework, “there is no real guarantee that any particular course teaches the particular skills and knowledge that industry needs” [3]. Chapter 3 introduces the definition of a requirement, where the customer are those in the industry, the product are the computer science and software engineering students, and the customer need or objective is the students having the knowledge of software internationalization.

In Chapter 4, the research question of the thesis was presented: “*Is there a framework for software internationalization that has been validated against industry requirements?*” It turns out that there did not exist such a framework, nor was there an adequate framework that met the requirements of our industry customers for software internationalization. The LEIT concluded that there is a clear gap between what the industry would like to see in education and what is actually available [3]. This clear gap indicates that the requirement to satisfy our industry customer needs is incomplete, infeasible, and necessary. Since no such framework for software internationalization currently exists, one was developed in this thesis. The contribution of this thesis includes a provisional framework to prepare graduates to internationalize software and a validation of the framework against industry requirements. The requirement of this framework is to provide a portable and standardized set of requirements for computer science and software engineering programs to teach future graduates.

In Chapter 5, we continued the research of LEIT by discussing the design of the industry survey questionnaire to validate the underlying set of requirements discussed in Chapter 3. In Chapter 6, we discussed the results of the industry survey questionnaire and analyzed the results of a number of hypothesis tests.

The overall conclusion of this thesis is:

The majority of respondents from the industry overall agree that in order for computer science and software engineering graduates to be prepared for industry, universities should teach students software internationalization by integrating the various requirements from Table 6, in order of importance, into current existing courses.

8.2. Lessons learned

Throughout the process of the thesis, the following lessons were learned:

- The industry respondents overall agree that prioritized and validated requirements of software internationalization should be integrated into existing courses
- Universities should pay further attention to the growing field of software internationalization
- Key members of major organizations, such as ACM, LISA, and the Unicode Consortium, can be contacted through their members through their member mailing lists
- Representatives from the industry have helped guide the research question and thesis
- Planning and writing a thesis takes a lot of time and motivation
- Do not expect to complete a thesis within a timeframe of your choice

8.3. Threats to validity

There were a number of threats to validity that may have affected the results of the data:

- There was no guarantee that could have prevented duplicate responses since respondents could have intentionally answered the industry survey questionnaire more than once
- Although there were a total of 278 valid responses, the true size of the population represented by the industry survey questionnaire is unknown
- Since no incentive was offered, some respondents who may feel strongly for or strongly against software internationalization will respond and outweigh those who don't care enough to do the survey [6]

8.4. Future work

There is plenty of future work for the growing field of software internationalization. However, in regards to the education for software internationalization, the following are opportunities for future work:

- Carry out various case studies at universities or colleges implementing the framework described in this thesis
- Reevaluate the validity of the framework of the thesis by either modifying, adding, or removing requirements
- Reevaluate the validity of the framework of the thesis by increasing the number of subjects from different companies, programming experiences, or groups of the company

- Consider other possible venues to gather responses aside from the web survey
- Evaluate other requirements for the framework, such as software internationalization security

8.5. Contributions

The following are the contributions made to the industry and the academic community of the computer science and software engineering fields:

- Found no existing and sufficient framework for software internationalization
- Extended the work of LISA Education Initiative Taskforce (LEIT)
- Gathered and analyzed requirements from literature review on the present state of education for software internationalization
- Contacted representatives from the industry for input on the various requirements for software internationalization
- Organized and prioritized requirements in order for students to gain sufficient knowledge of software internationalization
- Validated and prioritized requirements for software internationalization against industry needs using statistical analysis and hypothesis testing
- Presented a validated framework with a more sufficient set of requirements for the education of software internationalization coursework

CHAPTER 9. BIBLIOGRAPHY

- [1] Paula Gabbert, "Globalization and the computing curriculum," *ACM SIGCSE Bulletin*, vol. 35, no. 2, pp. 61-65, June 2003.
- [2] Michael J. Mahemoff and Lorraine J. Johnston, "Software Internationalisation: Implications for Requirements Engineering," in *Proceedings of the Third Australian Workshop on Requirements Engineering (ACRE)*, Geelong, Australia, October 1998, pp. 83-90.
- [3] Alan K. Melby, Lommel Arle, Sue Ellen Wright, and Alison Rowles, "LEIT phase I final report," LISA Education Initiative Taskforce (LEIT), Report November 22, 1998.
- [4] Eric Brechner, "Things they would not teach me of in college: what Microsoft developers learn later," in *Conference on Object Oriented Programming Systems Languages and Applications, Companion of the 18th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications*, Anaheim, CA, USA, 2003, pp. 134-136.
- [5] Alvin Yeo, "Software internationalisation and localisation," in *Computer-Human Interaction, 1996. Proceedings., Sixth Australian Conference*, Hamilton, New Zealand, November 24-27, 1996, pp. 348-349.
- [6] Franz J. Kurfess, Thesis Revision Comments, March 2010, Franz J. Kurfess, Professor, Computer Science Department, California Polytechnic State University, San Luis Obispo, CA.
- [7] Bert Esselink, *A practical guide to software localization: for translators, engineers and project managers.*: John Benjamins Publishing Company, October 1998.
- [8] Elvis Hau and Manuela Aparicio, "Software internationalization and localization in web based ERP," *ACM Special Interest Group for Design of Communication, Proceedings of the 26th annual ACM international conference on Design of communication*, pp. 175-180, September 22-24 2008.
- [9] International Business Machines (IBM). (1999, July 1) Glossary of Unicode Terms. [Online]. www.ibm.com/developerworks/library/glossaries/unicode.html
- [10] Localization Industry Standards Association (LISA). (2008-2009) LISA: Glossary. [Online]. www.lisa.org/Glossary.108.0.html
- [11] Tony Sahama, Chris Ho-Stuart, and James M. Hogan, "Developing and delivering a software internationalisation subject," in *ACM International Conference Proceeding Series; Vol. 54, Proceedings of the second workshop on Australasian information security, Data Mining and Web Intelligence, and Software Internationalisation - Volume 32*, Dunedin, New Zealand, 2004, pp. 199-204.
- [12] Kateryna Makarenko, Konstantyn Nagorny, and Nikolay Tkachuk, "Software localization in Ukraine: social-cultural issues and technological aspects," in *International Conference on Software Engineering, Proceedings of the 2008 international workshop on Software Engineering in east and south europe*, Leipzig, Germany, May 13, 2008, pp. 103-106.
- [13] Gregory E. Kersten, Mik A. Kersten, and Wojciech M. Rakowski, "Software and Culture: Beyond the Internationalization of the Interface," *Journal of Global Information Management*, vol. 10, no. 4, pp. 86-101, October-December 2002.
- [14] Kathleen Marvin, "Globalizing Your Business," *The Globalization Insider*, no. 1, January 1999.
- [15] Localization Industry Standards Association (LISA). (2008-2009) LISA: Why Globalize? [Online]. www.lisa.org/Why-Globalize.50.0.html
- [16] Encyclopædia Britannica Online. (2009) Microsoft Corporation. [Online]. search.eb.com

- [17] Erica L. Young, "A framework for the integration of the internationalization into the software development process," University of South Dakota, Thesis 1998.
- [18] Encyclopædia Britannica Online. (2009) Sun Microsystems, Inc. [Online]. search.eb.com
- [19] California Polytechnic State University. (2009, September 17) Cal Poly. [Online]. calpoly.edu
- [20] California Polytechnic State University. (2009, August 25) Simply the Best - Cal Poly. [Online]. www.calpoly.edu/simplythebest/simplythebest.html
- [21] Cal Poly College of Engineering. (2009, September 29) Dean's Advisory Council - Cal Poly. [Online]. ceng.calpoly.edu/dac
- [22] Cal Poly Computer Science Department. (2009, September) Industry Advisory Board - Cal Poly Computer Science Department. [Online]. www.csc.calpoly.edu/iab
- [23] Austin Community College - High Technology Institute - Technical Certification Programs. (2009, September 18) Localization / Globalization. [Online]. <http://www.austincc.edu/techcert/localization.php>
- [24] California State University, Chico. (2007, December 18) Enrollment Opens for Localization Certification Programs. [Online]. news.csuchico.edu/2007/12/18/enrollment-opens-for-localization-certification-programs/
- [25] Elon University Department of Computing Sciences. (2008, Fall) CSC 330 Computer Science III. [Online]. jonah.cs.elon.edu/dpowell2/Planning/Curriculum/Assessment/Fall08/CSC/PostAssessment-CSC330-F08.pdf
- [26] Rensselaer at Hartford Graduate Programs. (2003-2004) Catalog & Student Handbook. [Online]. www.ewp.rpi.edu/publications/catalog/archive/rhcatalog0304.pdf
- [27] San Diego State University. (2009-2010) SDSU General Catalog. [Online]. arweb.sdsu.edu/es/catalog/0910/webfolder/Computer%20Science.pdf
- [28] The University of Mississippi Department of Computer and Information Science School of Engineering. (2008, January 31) Course Catalog. [Online]. www.cs.olemiss.edu/academics/course_catalog.html
- [29] Encarta World English Dictionary [North American Edition]. (2009) syllabus definition - Dictionary - MSN Encarta. [Online]. encarta.msn.com/dictionary_1861717646/syllabus.html
- [30] Encyclopædia Britannica Online. (2009) International Business Machines Corporation. [Online]. search.eb.com
- [31] David Ira Rosenbaum, *Market Dominance: How Firms Gain, Hold, or Lose It and the Impact on Economic Performance.*: Praeger, May 30, 1998.
- [32] Encyclopædia Britannica Online. (2009) Computer. [Online]. search.eb.com
- [33] Jose Coronado and Carrie Livermore, "Business: going global with the product design process: does it make business sense?," *interactions*, vol. 8, no. 6, pp. 21-26, November, December 2001.
- [34] Tex Texin, Initial Phone and E-mail Interview, March 12, 2009, Tex Texin is the chief architect and globalization consultant for XenCraft.
- [35] Patricia Russo and Stephen Boor, "How Fluent is Your Interface?," in *Conference on Human Factors in Computing Systems, Proceedings of the INTERACT '93 and CHI '93 conference on Human factors in computing systems*, Amsterdam, The Netherlands, April 24-29 1993, pp. 342-347.

- [36] Markus Scherer and Mark Davis. (2005-2006) ICU Project. [Online]. www.icu-project.org/docs/papers/globalizing_software.ppt
- [37] Steffen Gross, "Internationalization and Localization of Software," Eastern Michigan University, Department of Computer Science, Ypsilanti, Michigan, Thesis June 19, 2006.
- [38] Karl E. Wiegers, *Software Requirements*, 2nd ed. Redmond, Washington, United States of America: Microsoft Press, 2003.
- [39] Tony Fernandes, *Global Interface Design.*: Morgan Kaufmann Pub, June 1995.
- [40] John Huan Vu and Clark Savage Turner. (2009, February) Survey Questionnaire of the Study of Software Internationalization and Globalization from an Academic and Industry Standpoint.
- [41] Localization Industry Standards Association (LISA). (2008-2009) LISA: Consulting. [Online]. www.lisa.org/LISA-Consulting.629.0.html
- [42] Ben Keller, Manuel A. Pérez-Quiñones, and Ravikiran Vatrappu, "Cultural Issues and Opportunities in Computing Education," in *9th International Conference on Engineering Education*, San Juan, Puerto Rico, July 23-28, 2006, pp. R1E-14 to R1E-19.
- [43] Jakob Nielsen, *International User Interfaces*, 1st ed., Elisa M. del Galdo, Ed.: Wiley, June 15, 1996.
- [44] Katharina Reinecke and Abraham Bernstein, "Predicting user interface preferences of culturally ambiguous users," in *Conference on Human Factors in Computing Systems, CHI '08 Extended Abstracts on Human Factors in Computing Systems*, Florence, Italy, April 05-10, 2008, pp. 3261-3266.
- [45] Unicode Consortium. (2009, May 5) Unicode Locale Data Markup Language (LDML). [Online]. www.unicode.org/reports/tr35/tr35-12.html
- [46] Encyclopædia Britannica Online. (2009) Unicode. [Online]. search.eb.com
- [47] John R. Vacca, "Unicode Breaks the ASCII Barrier," *Datamation*, pp. 55-56, August 1 1991.
- [48] International Components for Unicode. (2009) ICU Home Page. [Online]. site.icu-project.org
- [49] Matt Belge, "The next step in software internationalization," *interactions*, vol. 2, no. 1, pp. 21-25, January 1995.
- [50] World Wide Web Consortium. (2004-2008, August 27) W3C I18N (X)HTML & CSS Techniques. [Online]. www.w3.org/International/techniques/authoring-html
- [51] Encarta World English Dictionary [North American Edition]. (2009) collation definition - Dictionary - MSN Encarta. [Online]. encarta.msn.com/dictionary_1861598583/collation.html
- [52] XenCraft. (2002, 2003, 2004, 2005, January 3) Online Resources For Globalization, Internationalization, and Localization Provided By XenCraft. [Online]. www.xencraft.com/resources/
- [53] Localization Industry Standards Association (LISA). (2009) LISA: Best Practice Guides. [Online]. www.lisa.org/Best-Practice-Guides.467.0.html
- [54] Patrick Y. K. Chau, Melissa Cole, Anne P. Massey, Mitzi Montoya-Weiss, and Robert M. O'Keefe, "Cultural differences in the online behavior of consumers," *Communications of the ACM*, vol. 45, no. 10, pp. 138-143, October 2002.
- [55] Aaron Marcus, "Icon and symbol design issues for graphical user interfaces," in *International Users Interface*, 1996, pp. 257-270.

- [56] Simon Elegant, "Chinese Government Attacks Google Over Internet Porn," *TIME*, June 22 2009.
- [57] Encarta World English Dictionary [North American Edition]. (2009) framework definition - Dictionary - MSN Encarta. [Online]. http://encarta.msn.com/dictionary_1861613305/framework.html
- [58] Rafael J. Engel and Russell K. Schutt, *Practice of Research in Social Work, The*, 2nd ed.: Sage Publications, Inc, December 11, 2008.
- [59] Encyclopædia Britannica Online. (2009, December) statistics. [Online]. search.eb.com
- [60] Beth L. Chance, Professor, 2009-2010, Department of Statistics, California Polytechnic State University, San Luis Obispo.
- [61] Jay L. Devore, *Probability and Statistics for Engineers and the Sciences*, 7th ed. United States of America: Duxbury Press, January 29, 2008.
- [62] Karen J. McGaughey, Assistant Professor, 2009, Department of Statistics, California Polytechnic State University, San Luis Obispo.
- [63] SurveyMonkey.com. (1999-2009) SurveyMonkey.com. [Online]. www.surveymonkey.com
- [64] Encarta World English Dictionary [North American Edition]. (2009, December) doctor of philosophy definition - Dictionary - MSN Encarta. [Online]. <http://encarta.msn.com>
- [65] Association of Computing Machinery. (2009) Welcome. [Online]. www.acm.org
- [66] Encyclopædia Britannica Online. (2009) Association for Computing Machinery. [Online]. search.eb.com
- [67] Encyclopædia Britannica Online. (2009) Turing Award. [Online]. search.eb.com
- [68] Encyclopædia Britannica Online. (2009, September 16) ASCII. [Online]. search.eb.com
- [69] American Standards Association Incorporated, "American Standard Code for Information Interchange," American Standards Association Incorporated, New York, NY, Report ASA X3.4-1963, June 17, 1963.
- [70] Localization Industry Standards Association (LISA). (2008-2009) LISA: Frequently Asked Questions. [Online]. www.lisa.org/Frequently-Asked-Que.46.0.html
- [71] IEEE. (2009) IEEE. [Online]. ieee.org
- [72] Encyclopædia Britannica Online. (2009) Electrical and Electronics Engineers, Institute of. [Online]. search.eb.com
- [73] Encyclopædia Britannica Online. (2009) International Organization for Standardization. [Online]. search.eb.com
- [74] Localization Industry Standards Association (LISA). (2008-2009) LISA: Homepage. [Online]. www.lisa.org
- [75] Localization Industry Standards Association (LISA). (2008-2009) LISA: About LISA. [Online]. www.lisa.org/About-LISA.31.0.html
- [76] Unicode Consortium. (2006, August 31) History of Unicode: Summary Narrative. [Online]. www.unicode.org/history/summary.html
- [77] Miniwatts Marketing Group. (2009) Internet Usage World Stats - Internet and Population Statistics. [Online]. www.internetworldstats.com

- [78] Miniwatts Marketing Group. (2001-2009) Top Ten Internet Languages - World Internet Statistics. [Online]. www.internetworldstats.com/stats7.htm
- [79] Miniwatts Marketing Group. (2001-2009) World Internet Usage Statistics News and World Population Stats. [Online]. www.internetworldstats.com/stats.htm
- [80] Bastin Tony Roy Savarimuthu and Maryam Purvis, "Towards a multi-lingual workflow system: a practical outlook," in *ACM International Conference Proceeding Series; Vol. 54, Proceedings of the second workshop on Australasian information security, Data Mining and Web Intelligence, and Software Internationalisation - Volume 32*, Dunedin, New Zealand, 2004, pp. 205-210.
- [81] United States Census Bureau. (2009, September 10) International Data Base (IDB). [Online]. www.census.gov/ipc/www/idb/
- [82] Encyclopædia Britannica Online. (2009, September 16) Africa. [Online]. search.eb.com
- [83] California Polytechnic State University. (2007, Summer) 2007-09 Catalog Course Descriptions. [Online]. www.catalog.calpoly.edu/coursedescrip2007-09.html
- [84] Minitab Inc. (2007) 1-Sample Sign. Minitab Help.
- [85] Localization Industry Standards Association (LISA). (2008-2009) LISA: What is Globalization? [Online]. www.lisa.org/What-Is-Globalization.48.0.html
- [86] Association for Computing Machinery. (1999) Software Engineering Code of Ethics and Professional Practice. [Online]. www.acm.org/about/se-code
- [87] World Wide Web Consortium (W3C). (1994-2004) About World Wide Web Consortium (W3C). [Online]. www.w3.org/Consortium
- [88] Daniel S. Yates, David S. Moore, and Daren S. Starnes, *The Practice of Statistics: TI-83/89 Graphing Calculator Enhanced*, Second Edition ed. United States of America: W. H. Freeman, July 3, 2002.
- [89] John W. Chinneck. (1999, September 29) How to Organize Your Thesis. [Online]. sce.carleton.ca/faculty/chinneck/thesis.html

APPENDIX A. GLOSSARY AND ACRONYMS

A.1. ACM (Association for Computing Machinery)

ACM stands for “Association for Computing Machinery” [65] [66]. ACM is an international organization for computer science, technology professionals, and institutions associated with the field [66]. ACM is also known for giving the Alan M. Turing Award, the most prestigious award in computer science [66] [67].

A.2. ASCII (American Standard Code for Information Interchange)

ASCII stands for “American Standard Code for Information Interchange” [68]. ASCII is a “standard data-transmission code that is used by smaller and less-powerful computers to represent both textual data” and device commands [68].

ASCII, an 8-bit character, was created in 1963 and supports the basic characters of the Latin alphabet [17] [69]. Although ASCII became the standard for personal computers, it was inadequate for software internationalization because it did not support many non-English languages [68] [47]. For example, ASCII does not support the German letter ü.

A.3. Cal Poly

Cal Poly is the short name for California Polytechnic State University. Cal Poly is “a nationally ranked, four-year, comprehensive public university located in San Luis Obispo, California,” with over 18,000 students who follow a “learn by doing” philosophy [19]. Cal Poly prides itself on about 90% of its graduates “working full-time or attending graduate school within one year of graduation” for the past decade [20]. Depending on the economy, 300 to 600 different employers visit the university to recruit and employ graduates each year [20].

To ensure that Cal Poly graduates are the “cream of the crop,” the colleges and respective departments “foster relationships with industry, scientific and professional communities, and government” [21]. For example, the Dean’s Advisory Council of the College of Engineering addresses issues such as strategic planning, industry-university partnership, and program review and assessment [21]. Within the College of Engineering, the Computer Science Department seeks advice from members of the Industrial Advisory Board¹ [22].

¹ Participating members of the Industry Advisory Board are listed under Appendix B.

A.4. g11n (Globalization)

Globalization is abbreviated as g11n because there are 11 characters between the first letter “G” and the last letter “N” [70].

A.5. GILT (Globalization, Internationalization, Localization and Translation)

GILT stands for “Globalization, Internationalization, Localization and Translation” [70]. GILT is a term “used to refer to all of the language- and culture-related processes involved in global business” [70].

A.6. i18n (Internationalization)

Internationalization is abbreviated as i18n because there are 18 characters between the first letter “I” and the last letter “N” [8] [70].

A.7. ICU

ICU stands for “International Components for Unicode” [48]. ICU is an “open source software development project delivering Unicode support on a variety of platforms” [48].

A.8. IEEE

IEEE stands for “Institute of Electrical and Electronic Engineers” [71] [72]. IEEE is an “international organization of engineers and scientists in electrical engineering, electronic, and allied fields” [72].

A.9. ISO

ISO stands for “International Organization for Standardization” [73]. ISO is an international organization for the “standardization in all technical and non-technical fields, except electrical and electronic engineering” [73].

A.10. l10n (Localization)

Localization is abbreviated as l10n because there are 10 characters between the first letter “L” and the last letter “N” [8] [70].

A.11. LISA

LISA stands for “The Localization Industry Standards Association” [74]. LISA is “member-governed organization led by a board of leading figures in globalization and related industries” [75].

A.12. Unicode

Unicode is an international “character-encoding system designed to support the electronic interchange, processing, and display of the written texts of the diverse languages of the modern and classical world” [46]. Unicode is either denoted as UTF-8 or UTF-16 depending on the domain in which the character set is used [36].

In the 1980s, the Unicode Consortium was formed to “standardize, extend and promote the Unicode character encoding, a fixed-width, 16-bit character encoding for over 60,000 graphic characters” [76]. Companies such as Apple, IBM, Sun Microsystems, Novell, Lotus, and Xerox contributed to this milestone [47].

APPENDIX B. CAL POLY INFORMATION

B.1. Members of the Computer Science Department Industrial Advisory Board (Spring 2009)

Adobe	Lawrence Livermore National Labs
Amazon.com	Locus Technologies
Amgen	Management Technologies
Borland Software	Microsoft
Centrify Corporation	NVIDIA
Chevron	Qualcomm
Cisco Systems	Raytheon
DreamWorks	SAP Labs, Inc.
Embarcadero Technologies	St. Jude Medical
IBM	Sun Microsystems
Intel	VMWare
Intuit	Yahoo, Inc.
Jet Propulsion Laboratories, NASA	

APPENDIX C. STATISTICS OF INTERNET USERS

The statistics on internet users is retrieved from the Internet World Stats, an “international website that features up to date world internet usage ... for over 233 individual countries and world regions” [77].

Four figures from the Internet World Stats that will be discussed in this section are [77]:

- Figure 18 shows the number of internet users ranked by the top ten languages [78].
- Figure 19 shows the growth of internet users ranked by the top ten languages from 2000 [78].
- Figure 20 shows the number of internet users ranked by region [79].
- Figure 21 shows the growth of internet users ranked by region from 2000 [79].

C.1. Internet users ranked by top ten languages

Figure 18 displays the top ten languages as of June 30, 2009, ranked by number of internet users [78]. Among these ten languages, English and Chinese were the most frequently used [78]. Since Chinese is so broadly used and the fact that China houses the largest portion of the world’s population (over 1.338 billion people), it makes business sense to invest money and time into localizing Chinese [80] [81].

Figure 18 - Number of internet users ranked by the top 10 languages on June 30, 2009.

Figure 19 shows the growth of internet users classified by the top ten languages used from the year 2000 to June 30, 2009 [78]. In comparison to Figure 18, the largest language growth is in Arabic [78]. Arabic has grown over 1,860%, followed by Russian at over 1,125%. According to the data, it makes business sense to invest money and time into localizing the Arabic and Russian language because it is growing rapidly on the internet.

Figure 19 - Growth of internet users ranked by the top 10 languages from 2000 to June 30, 2009.

C.2. Internet users ranked by region

Figure 20 shows the number of internet users ranked by region on June 30, 2009 [79]. The figure shows that the largest number of internet users by region is Asia [79]. This makes sense since Asia is the biggest continent in the world and includes its two most populous countries – China and India [81].

Figure 20 - Number of internet users ranked by region on June 30, 2009.

Figure 21 shows the growth of internet users ranked by region from 2000 to June 30, 2009 [79]. In comparison to Figure 20, Figure 21 shows that the Middle East and Africa have both grown over 1,300% since 2000 [79]. This parallels Figure 19, where the largest growth of internet users ranked by language is

in Arabic, the language spoken in many parts of the Middle East and Northern Africa. The growth in Africa could possibly be due to its considerable economic development and various other factors [82].

Figure 21 - Growth of internet users ranked by region from 2000 to June 30, 2009.

C.3. Synopsis

The overall conclusion is that the English language does not constitute the majority of the internet market. The figures show that the Chinese and Arabic languages, along with the regions of the Middle East and Asia are markets to target for globalization. If a developer decides to localize for the Chinese language, it is important to understand the orientation and arrangement of the character set of the Chinese language [80]. If a developer decides to localize for the Arabic language, they should understand the character set and arrangement of the sentences from right to left [36]. If a developer decides to localize software for India, they should note that although Hindi is the national language, it is only spoken by 25% of the population followed by various other unique languages [80].

APPENDIX D. STUDENTS' CURRENT PERCEPTIONS

D.1. Foreword

The section discusses a survey questionnaire on students' perception of software internationalization. This section was created and reviewed as part of a graduate software engineering course taught by Dr. David S. Janzen, Associate Professor at Cal Poly.

D.2. Goals

The goal of the survey questionnaire was to determine students' perceptions of software internationalization. The internationalization survey questionnaire would evaluate whether or not students believe that:

- Internationalization is a familiar topic
- Internationalization will be important to their future careers
- Internationalization is easy to learn, implement, and deploy
- Internationalization is easy to fix, maintain, and debug
- They are prepared to internationalize any product

D.3. Subjects

The subjects of the survey questionnaire were students from the Professional Responsibilities course taught at Cal Poly during the winter quarter of 2009. According to the Cal Poly 2007-2009 Course Catalog, this course taught students the ethics of computer science and analyzed the IEEE and ACM Software Engineering Code of Ethics [83]. Students were educated about the ACM Software Engineering Code of Ethics through lectures and laboratory assignments. Topics covered in the course included quality tradeoffs, software system safety, intellectual property, the history of computing, and the social implications of computers in the modern world [83]. Although the students varied in experience, all were required to complete one introductory software engineering course before taking the Professional Responsibilities course [83]. The course is required for the completion of an undergraduate degree in Computer Science or Software Engineering and is regularly taught by computer science professor, Dr. Clark Savage Turner J.D. Ph.D.

D.4. Survey design and procedure

D.4.1. Human Subjects Committee and consent form

The students were notified that they were going to take part in the software internationalization survey questionnaire. Each student was given a two-page handout reviewed and approved by the Cal Poly Human Subjects Committee.

The first page contained the consent form to participate in the survey questionnaire. The consent form outlined the purpose of the survey questionnaire, how to answer the questions in the survey questionnaire, how long the survey questionnaire should take, how the students' responses would be kept confidential, and how to contact the researchers. The consent form included contact information for the Cal Poly Human Subjects Committee and the Cal Poly Dean of Research and Graduate Programs. In order to participate in the survey questionnaire, students had to sign the consent form. However, students could choose to opt out by not answering any of the survey questions.

The consent form can be found in Appendix A. It was approved by Dr. Steve Davis, Chair of the Cal Poly Human Subjects Committee.

D.4.2. Survey

The second page contained the survey questionnaire, which included the definition of internationalization, a list of demographic questions, and a list of survey questions. The survey questionnaire can be found in Appendix A. It was reviewed by Dr. Jeffrey C. Sklar, a statistics professor at Cal Poly, and was approved by Dr. Steve Davis, Chair of the Cal Poly Human Subjects Committee.

The definition of software internationalization given to the students is as follows:

Internationalization is the process of producing an application that can be localized for a particular country without any changes to the program code. Internationalized applications store their text in external sources, and use locale-sensitive utilities for formatting and collation.

The demographic questions included the student's name, year, and major, and then asked the student to list languages in which he or she is fluent, and to list the courses he or she has taken at Cal Poly that covered internationalization. To protect the student's identity and prevent duplicates, only the researchers would read the answers. Students could provide either their actual names or a pseudonym.

A five-point Likert scale ranging from agree, to neutral, to disagree was used to answer the survey questionnaire. The survey questions asked the students whether or not they agree with each of the following statements:

- Internationalization is a familiar topic to me
- Internationalization will be important to my future career
- Internationalization is easy to learn, implement, and deploy
- Internationalization is easy to fix, maintain, and debug
- I am prepared to internationalize any product

D.4.3. Procedure

The two-page survey questionnaire was distributed to the students at the beginning of class. The professor collected the surveys once students were finished. Students were asked to stay quiet until all complete and incomplete surveys were collected. The survey did not last more than ten minutes.

D.5. Data results and analysis

D.5.1. Data filtering

There were three specific data results that were omitted or changed for this analysis.

First, when students were asked what year they were in college, some filled in a numerical value while others filled in words such as “senior.” Although the intention of the question was to produce a numerical value, the conclusion drawn from this question was that all students were of senior standing.

Second, when students were asked about languages in which they are fluent, some believed the question meant programming languages such as C, Java, and PHP instead of spoken languages. The intention of this question was to determine if there were other non-programming languages a student was able to speak, read, or write fluently. Therefore, no conclusion was drawn from this question.

Third, when students were asked whether or not they were prepared to internationalize any software product, both students and researchers realized that the question was ambiguous. For example, when would one say that a student or even a computing professional is *prepared*? Therefore, no conclusion was drawn from this question.

D.5.2. Demographics

There were 19 senior-level computer science and software engineering students who participated in the survey questionnaire. Of the 19 students, four were software engineering majors and 15 were computer science majors. Figure 22 shows a pie chart of participants by major.

Figure 22 - Pie chart of participants of the survey questionnaire by major.

When asked if the students could list courses that covered software internationalization, only about a quarter answered. Of these students, the courses indicated included the fields of software engineering, databases, professional responsibilities, and general education courses.

D.5.3. Summary of data results

The summary of data results from the survey questionnaire on students' perceptions is displayed in Table 7 and Figure 23.

Likert Scale	i18n is a familiar topic to me	i18n will be important to my future career	i18n is easy to learn, implement, and deploy	i18n is easy to fix, maintain, and debug
Disagree	3	4	0	0
Somewhat Disagree	6	8	1	1
Neutral	4	6	5	4
Somewhat Agree	3	4	6	5
Agree	3	0	7	8

Table 7 - Summary of results from the survey questionnaire on students' perceptions.

Figure 23 - Summary of results from the survey questionnaire on students' perceptions.

The first statement was “Internationalization is a familiar topic to me.” There was no consensus on whether or not internationalization is a familiar topic to the students, but at least half of them disagreed or somewhat disagreed, implying that they felt unfamiliar with internationalization.

The second statement was “Internationalization will be important to my future career.” About two-thirds of the students overwhelmingly disagreed or somewhat disagreed that internationalization will be important to their career.

The third statement was “Internationalization is easy to learn, implement, and deploy.” In contrast to the second statement, at least two-thirds of the students agreed or somewhat agreed that internationalization is easy to learn, implement, and deploy.

The fourth statement was “Internationalization is easy to fix, maintain, and debug.” Similarly to the third statement, at least three-fourths of the students agreed or somewhat agreed that internationalization is easy to fix, maintain, and debug.

The last statement “I am prepared to internationalize any product” was omitted from the analysis as mentioned in Section **Error! Reference source not found.** under **Error! Reference source not found.**.

D.6. Threats to validity

The most obvious threat to validity was the small sample size of 19 students. A minor threat to validity was the ambiguous statements mentioned in Section **Error! Reference source not found.**.

D.7. Future work

Future work is extremely necessary for measuring the students' perceptions of software internationalization in the future. The survey questionnaire would first need to be revised. In addition, there is a need for a larger sample of subjects, subjects with differing programming experiences (ranging from students to professionals), and subjects coming from different universities and areas of the world. All of these changes will decrease the threats to validity and help the measurements of students' perceptions.

D.8. Synopsis

Based on the survey questionnaire on students' perception of software internationalization, it is concluded that at least a majority of students in the survey questionnaire felt:

- Unfamiliar with internationalization
- Internationalization is not important to their future career
- Internationalization is easy to learn, implement, and deploy
- Internationalization is easy to fix, maintain, and debug

Although all of the questions were quite subjective, the results lead to a number of questions:

- Can a majority of students who are unfamiliar with internationalization accurately determine whether or not internationalization is important to their future career?
- Can a majority of students who are unfamiliar with internationalization accurately determine whether or not internationalization is easy to learn, implement, and deploy?
- Can a majority of students who are unfamiliar with internationalization accurately determine whether or not internationalization is easy to fix, maintain, and debug?

The plausible answer to all the questions above is simply no. A majority of students who are unfamiliar with internationalization cannot accurately and properly answer the other statements. Students could have been unaware of the "difficulty in determining what can be generalized across cultures and what needs to be customized to a given culture" [13].

Appendix D leads to the conclusion that there is a need for students to learn about software internationalization. However, the next question is, would students' perceptions change if an internationalization laboratory assignment is given? Appendix E includes a revised version of the survey

questionnaire from Appendix D and an attempt to determine whether an experimental internationalization laboratory assignment can change students' perception of software internationalization.

The result from this section does not imply a generalization to other contexts since there were multiple factors that could have biased the results. The results show that Cal Poly students were not meeting the demand nor were they well-educated about software internationalization.

APPENDIX E. STUDENTS' PERCEPTIONS ON A LABORATORY ASSIGNMENT

E.1. Foreword

This section will discuss a revised survey questionnaire on students' perceptions of software internationalization. The survey questionnaire has been revised due to the ambiguity from the survey questionnaire in Appendix D. The biggest difference is determining whether the students' perception of software internationalization changes after an internationalization laboratory assignment. This section was created and reviewed as part of a graduate software engineering course taught by Dr. David S. Janzen, Associate Professor at Cal Poly.

This section discusses a case study wherein students localized an existing program into multiple languages using an existing internationalization infrastructure. Students not only learned about software internationalization, but they engaged in a hands-on laboratory assignment. The goal was to determine whether the exposure to internationalization activities changed the students' perception of software internationalization. Although Appendix D found that the students' were more familiar with software internationalization after the laboratory assignment, their perceptions on other aspects of software internationalization did not change.

E.2. Goals and research questions

The goal was to determine whether the exposure to internationalization activities changed the students' perceptions of software internationalization. The students' perceptions were measured by their answers to a survey questionnaire before the internationalization laboratory assignment (pre-lab) and after the internationalization laboratory assignment (post-lab).

Measuring the students' perception would help answer a number of research questions, such as:

- Do students feel more or less familiar with internationalization?
- Do students feel internationalization is more or less important to their career?
- Do students feel internationalization should be taught more or less at Cal Poly?
- Do students feel that learning internationalization is harder or easier afterwards?
- Do students feel that implementing and deploying software that supports internationalization is harder or easier afterwards?

- Do students feel that fixing and maintaining software that supports internationalization is harder or easier afterwards?

To determine any changes in students' perceptions after the laboratory assignment, the data from the answered survey questionnaire would be analyzed statistically.

E.3. Subjects

The subjects of the survey questionnaire were students from the Professional Responsibilities course taught at Cal Poly during the spring quarter of 2009. Section D.3 provides information about the Professional Responsibilities course in addition to the course description, the students' experience levels, course prerequisites, and the name of the professor who regularly teaches it.

E.4. Survey design and procedure

E.4.1. Human Subjects Committee and consent form

All students were notified that they were part of a case study on software internationalization. The students were each given a two-page handout approved by the Cal Poly Human Subjects Committee. The consent form is given before and after the internationalization laboratory assignment with a pre-lab and post-lab survey questionnaire, respectively.

The first page contained the consent form to participate in the case study survey questionnaire. The consent form outlined the purpose of the case study survey questionnaire, how to answer the questions in the case study survey questionnaire, how long the case study survey questionnaire shall take, how the students' responses would be kept confidential, and how to contact the researchers. It included contact information for the Cal Poly Human Subjects Committee and the Cal Poly Dean of Research and Graduate Programs. In order to participate in the case study survey questionnaire, students must sign the consent form. However, students may choose to opt out by not answering any of the survey questions.

The consent form can be found in Appendix G. The consent form was approved by Dr. Steve Davis, Chair of the Cal Poly Human Subjects Committee.

E.4.2. Pre-lab survey questionnaire and procedure

A pre-lab survey questionnaire is given to students at the beginning of the class and before starting the internationalization laboratory assignment. The pre-lab survey questionnaire is the second page following the consent form described in Section E.4.1. It included the definition of internationalization, a

list of demographic questions, and a list of survey questions. The pre-lab survey questionnaire can be found in Appendix G. The pre-lab survey questionnaire was reviewed by Dr. Jeffrey C. Sklar, a statistics professor at Cal Poly, and approved by Dr. Steve Davis, Chair of the Cal Poly Human Subjects Committee.

The definition given to the student is as follows:

Internationalization is the process of producing an application that can be localized for a particular country without any changes to the program code. Internationalized applications store their text in external sources, and use locale-sensitive utilities for formatting and collation.

The demographic questions included the student's name, year, and major. Students were also asked whether they have used Adobe Flex Builder; what other languages, besides English, they could read and write fluently; and what other courses they have taken that covered internationalization. To protect students' identities and prevent duplicates, only the researchers would be reading the answers.

The survey was given in two parts. The first part asked the students whether or not they agreed with each of the statements below. A five-point Likert scale from agree, to neutral, to disagree was used to answer the survey questionnaire.

- Internationalization is a familiar topic to me
- Internationalization will be important to my future career
- Internationalization should be taught more at Cal Poly

The second part of the survey questions asked the students whether or not they believe each scenario described in the statements below is easy or hard. A five-point Likert scale from easy, to neither, to difficult was used to answer the survey questionnaire.

- Learning internationalization
- Implementing, deploying software that supports internationalization
- Fixing, maintaining software that supports internationalization

Students were asked to stay quiet until all complete and incomplete surveys were collected. The researcher collected the pre-lab survey questionnaire from the students once they were done. The survey questionnaire did not last more than ten minutes.

E.4.3. Internationalization laboratory assignment and procedure

After completing the pre-lab questionnaire, the students were required to complete an internationalization laboratory assignment as part of their class coursework. Students completed this assignment over the course of a week. Students were given at least two laboratory hours to work with the help of the instructor and the researcher. However, students were also expected to work outside of class with their group members.

Following the same format as the other laboratory assignments in the course, the internationalization laboratory assignment consisted of a purpose, procedure, and instructions for deliverables. The procedure was further broken down to a pre-lab, a list of preliminary questions, an analysis of the ACM Software Engineering Code of Ethics, an implementation of the project, a presentation, and a list of concluding questions. The assignment can be found in Appendix G. It was reviewed and approved by the professor of the course.

The pre-lab portion of the procedure required the students to install Adobe Flex Builder onto their computer. The choice of Adobe Flex Builder instead of a development environment supporting Java or C++ was to level the playing field for students, since nearly every student had no experience with Adobe Flex Builder.

Before starting on the assignment, preliminary questions were given to the students. The preliminary questions required each student to answer a variety of questions about software internationalization, which helped them to learn about the procedure. The questions allowed students to research and understand some concepts about software internationalization. These questions can be found in Appendix G.

As part of the Professional Responsibilities course, students were required to analyze the ACM Software Engineering Code of Ethics. Students needed to define, interpret, and justify which sections of the ACM Software Engineering Code of Ethics apply to the use of software internationalization.

During the internationalization laboratory assignment, students worked in groups of five or six to implement the project. The group converted an existing program into multiple languages using an existing internationalization infrastructure. The existing program searches and retrieves images from Flickr after the user enters keywords in a search prompt. This process of making a program available in multiple languages

is called localization, an aspect of software internationalization. Each group made the program available in twenty languages, including right-to-left translation, complex character sets such as Chinese and Hebrew, and differing date formats. Groups were given a website link showcasing the final product.

Once the group completed the assignment, they were required to upload their resulting program to a website and present it to the class. Each group was required discuss their contribution to the internationalization laboratory assignment. For a maximum of five minutes, the groups described their techniques or tools, the difficulties they encountered, and their experiences with Adobe Flex Builder.

After the presentation, each group reflected on what they had learned from the internationalization laboratory assignment. Their answers will be used to help the professor of the course and the researcher to improve the internationalization laboratory assignment for future classes. The concluding questions can be found in Appendix G.

At the end of the project, each group submitted a packet of their work and their final product to the professor.

E.4.4. Post-lab survey questionnaire and procedure

A post-lab survey questionnaire was given to students after they completed the internationalization laboratory assignment. The post-lab survey questionnaire is the second page, following the consent form described in Section E.4.1. The post-lab survey questionnaire included the definition of internationalization, a list of demographic questions, and a list of survey questions can be found in Appendix G. It was reviewed by Dr. Jeffrey C. Sklar, a statistics professor at Cal Poly, and approved by Dr. Steve Davis, Chair of the Cal Poly Human Subjects Committee.

Unlike the pre-lab survey questionnaire in Section E.4.2, the demographic questions only included the student's name and year. This demographic question is to use the post-lab survey questionnaire to pair with the pre-lab survey questionnaire. To protect the student's identity and prevent duplicates, only the researchers would be reading the answers.

Like the pre-lab survey questionnaire in Section E.4.2, the survey was given in two parts. The first part asked students whether or not they agreed with each of the statements below after completing the internationalization laboratory assignment. A five-point Likert scale from agree, to neutral, to disagree was used to answer the survey questionnaire.

- Internationalization is a familiar topic to me
- Internationalization will be important to my future career
- Internationalization should be taught more at Cal Poly

The second part of the survey asked the students whether or not they believe that each the scenario described below is easy or hard after completing the internationalization laboratory assignment. A five-point Likert scale from easy, to neither, to difficult was used to answer the survey questionnaire:

- Learning internationalization
- Implementing, deploying software that supports internationalization
- Fixing, maintaining software that supports internationalization

The researcher collected the post-lab survey questionnaire from the students once they were done. Students were asked to stay quiet until all complete and incomplete surveys were collected. The survey questionnaire did not last more than ten minutes.

E.5. Summary of hypotheses

The summary of hypotheses is summarized in Table 8 for measuring the change in students' perceptions for each research question. The value of zero is explained in Section E.7.1 on calculating the change of students' perceptions.

Research Question	Null Hypothesis	Alternative Hypothesis
Do students feel more or less familiar with internationalization?	$\text{familiar}_{\text{post-pre}} = 0$	$\text{familiar}_{\text{post-pre}} \neq 0$
Do students feel internationalization is more or less important to their career?	$\text{important}_{\text{post-pre}} = 0$	$\text{important}_{\text{post-pre}} \neq 0$
Do students feel internationalization should be taught more or less at Cal Poly?	$\text{taught}_{\text{post-pre}} = 0$	$\text{taught}_{\text{post-pre}} \neq 0$
Do students feel that learning internationalization is harder or easier afterwards?	$\text{learning}_{\text{post-pre}} = 0$	$\text{learning}_{\text{post-pre}} \neq 0$
Do students feel that implementing and deploying software that supports internationalization is harder or easier afterwards?	$\text{implementing}_{\text{post-pre}} = 0$	$\text{implementing}_{\text{post-pre}} \neq 0$
Do students feel that fixing and maintaining software that supports internationalization is harder or easier afterwards?	$\text{fixing}_{\text{post-pre}} = 0$	$\text{fixing}_{\text{post-pre}} \neq 0$

Table 8 - Summary of hypotheses for the research questions.

Data were gathered when students answer each of the research questions in the pre-lab survey questionnaire and post-lab survey questionnaire. The null hypothesis states that the students' perceptions before and after the internationalization laboratory assignment stayed the same. The alternative hypothesis states the students' perceptions before and after the internationalization laboratory assignment differed.

E.6. Data results

E.6.1. Demographics

With the exception of one student, there were twenty-five upper-level undergraduate and graduate students in the computer science and software engineering programs. Of those twenty-five students, twenty were in the computer science program, four were in the software engineering program, and one was in neither of the programs. Figure 24 shows a pie chart of participants of the case study by major indicated by their percentages.

Figure 24 - Pie chart of participants of the case study by major.

There were 11 third year students, seven fourth years and the remaining students were in their fifth year or higher. Figure 25 shows a pie chart of participants by year of study.

Figure 25 - Pie chart of participants of the case study by year of study.

When asked if students have used Adobe Flex Builder before, only three students indicated that they had any experience. When asked if students could list courses that covered software internationalization, only about a fifth of them answered. Of that portion of students, the courses indicated included the fields of software engineering, graphical user interface design, systems programming, professional responsibilities, and general education courses.

When asked whether students knew other languages besides English in which they could both read and write fluently, fourteen students responded positively. Figure 26 depicts a pie chart of participants by the number of fluent languages. The following languages were indicated by one or more students in the survey questionnaire:

- Chinese
- French
- German
- Gujarati
- Hindi
- Japanese
- Spanish
- Swedish
- Telugu
- Vietnamese

Figure 26 - Pie chart of participants of the case study by number of languages.

E.6.2. Survey questionnaire

This section summarizes the overall pre-lab and post-lab perceptions for each of the research questions and any trends among the data. The first part is displayed in Table 9 and as a bar chart in Figure 27. The second part of the survey questionnaire is displayed in Table 10 and as a bar chart in Figure 28. The data from the survey questionnaire can be found in Appendix G.

	i18n is a familiar topic to me			i18n will be important to my future career			i18n should be taught more at Cal Poly	
	Pre	Post		Pre	Post		Pre	Post
Disagree	5	1		1	1		0	3
Somewhat Disagree	5	4		2	4		1	3
Neutral	7	7		8	9		12	10
Somewhat Agree	5	9		8	8		8	7
Agree	3	4		6	3		4	2

Table 9 - Summary of results from the first three questions of the pre- and post-survey questionnaire.

	Learning i18n			Implementing, deploying software that supports i18n			Fixing, maintaining software that supports i18n	
	Pre	Post		Pre	Post		Pre	Post
Difficult	4	2		5	3		4	3
Somewhat Difficult	3	4		10	8		7	6
Neither	12	10		6	6		10	9
Somewhat Easy	6	6		4	6		4	6
Easy	0	3		0	2		0	1

Table 10 - Summary of results from the last three questions of the pre- and post-survey questionnaire.

Figure 27 - Summary of results from the first part of the pre- and post-survey questionnaire.

Figure 28 - Summary of results from the second part of the pre- and post-survey questionnaire.

There were a few significant trends on the summarized results. On the first part of the survey questionnaire, only the first statement, “Internationalization is a familiar topic to me,” had an increase in agree and somewhat agree and a decrease in disagree and somewhat disagree. The last two statements of the first part of the survey questionnaire had the opposite result, in that there was a decrease in agree and somewhat agree and an increase in disagree and somewhat disagree. On the second part of the survey questionnaire, all of the statements had an increase in easy and somewhat easy and a decrease in difficult and somewhat difficult.

It can be concluded that after the internationalization laboratory assignment, the change of perceptions in students felt:

- Internationalization is a more familiar topic to them
- Internationalization is not important to their future career
- Internationalization should not be taught more at Cal Poly
- Learning internationalization is easy
- Implementing and deploying software that supports internationalization is easy
- Fixing and maintaining software that supports internationalization is easy

However, this section does not offer a conclusion of student perceptions. A statistical analysis of the changes for each particular student will be discussed in Section E.7.

E.7. Data analysis

E.7.1. Calculating changes in perceptions

Since the survey questionnaire used a Likert scale, each of the answers was given a unit value.

Without loss of generality, assume the following values were used for each answer:

Unit Value Assigned	-2	-1	0	1	2
First Part of the Survey Questionnaire	Disagree	Somewhat Disagree	Neutral	Somewhat Agree	Agree
Second Part of the Survey Questionnaire	Difficult	Somewhat Difficult	Neither	Somewhat Easy	Easy

Table 11 - Assigning values to the Likert scale used in the survey questionnaire.

To calculate the change in students' perceptions, the following equation is used:

$$(\text{Post-Lab Answer Unit Value}) - (\text{Pre-Lab Answer Unit Value}) = (\text{Change in Students' Perceptions})$$

Figure 29 - Equation to calculate change in student's perceptions.

Given this equation, if an answer from the pre-lab to the post-lab was the same unit value, then there is no change in students' perceptions. If the resulting value from the change in students' perceptions is positive, then the change in students' perceptions is towards agree or towards easy. Similarly, if the resulting value from the change in students' perceptions is negative, then change in students' perceptions is towards disagree or towards difficult. The change in students' perceptions is summarized in the histograms displayed in Section E.7.2.

E.7.2. Changes in perceptions

This section summarizes the overall change in students' perceptions after the laboratory assignment with using histogram with numerical values along the horizontal and noting any trends among the data.

Figure 30 displays a histogram of change of students' perceptions of "Internationalization is a familiar topic to me." The histogram skews to the left, showing that the majority of students changed their perception towards agree.

Figure 30 - Histogram of change of "i18n is a familiar topic to me."

Figure 31 displays a histogram of change of students' perceptions of "Internationalization will be important to my future career." The histograms weakly skew to the right, showing that a number of students changed their perception towards disagree.

Figure 31 - Histogram of change of "i18n will be important to my future career."

Figure 32 displays a histogram of change of students' perceptions of "Internationalization should be taught more at Cal Poly." The histograms skew to the right, showing that nearly half of the students changed their perception towards disagree.

Figure 32 - Histogram of change of "i18n should be taught more at Cal Poly."

Figure 33 displays a histogram of change of students' perceptions of "Learning internationalization." The histogram is symmetric, showing that a number of students did not change their perceptions while a number changed their perception towards easy.

Figure 33 - Histogram of change of "Learning i18n."

Figure 34 displays a histogram of change of students' perceptions of "Implementing, deploying software that supports internationalization." The histogram skews to the left, showing that the majority of students changed their perception towards easy.

Figure 34 - Histogram of change of "Implementing, deploying software that supports i18n."

Figure 35 displays a histogram of change of students' perceptions of "Fixing, maintaining software that supports internationalization." The histogram is symmetric with a slight skew to the right showing that the majority of students did not change their perceptions while a few changed their perceptions towards easy.

Figure 35 - Histogram of change of "Fixing, maintaining software that supports i18n."

E.7.3. Nonparametric one-sample signed test

To determine if the change in students' perceptions for each research question is statistically significant, the researchers used a nonparametric one-sample signed test. The nonparametric one-sample signed test is a "hypothesis test for the median of a single population" [84]. In this case, the null hypothesis is the value zero, which stands for the median, meaning no change in students' perceptions. In addition, the test is an "analog of the one-sample t-test because it does not require the data to come from a normally distributed population" [84]. Using Table 8 as a reference, Section E.8 will discuss the results after running the nonparametric one-sample signed test on the hypotheses using Minitab. The test was advised by Dr. Jeffrey C. Sklar, a statistics professor at Cal Poly.

E.8. Results

Using the nonparametric one-sample signed test on the hypotheses discussed in Section E.5 on page 86 with Minitab, Table 12 summarizes the recorded p-values and the overall conclusion using a significance level of 0.05.

Research Question	P-Value	Conclusion
Do students feel more or less familiar with internationalization?	0.0127	Reject the null hypothesis
Do students feel i18n is more or less important to their career?	0.3018	Fail to reject the null hypothesis
Do students feel i18n should be taught more or less at Cal Poly?	0.1435	Fail to reject the null hypothesis
Do students feel that learning i18n is harder or easier afterwards?	0.1460	Fail to reject the null hypothesis
Do students feel that implementing and deploying software that supports internationalization is harder or easier afterwards?	0.1671	Fail to reject the null hypothesis
Do students feel that fixing and maintaining software that supports internationalization is harder or easier afterwards?	0.3438	Fail to reject the null hypothesis

Table 12 - Summary of p-values after conducting a nonparametric one-sample signed test.

E.9. Threats to validity

During the presentation portion of the laboratory assignment, the students provided a great deal of feedback that later contributed to the threats to validity. First and foremost, internationalization and localization were relatively new concepts, introducing some overhead for the laboratory assignment. In addition, the majority of students felt that installing and using Adobe Flex Builder with its respective documentation was extremely difficult and tedious. The students commented that although the laboratory assignment was useful in learning internationalization, they would have preferred to implement the program in Java. Another threat to validity was the small sample size of twenty-five students.

E.10. Future work

Future work is extremely necessary for measuring the students' perceptions of software internationalization in the future. The survey questionnaire and the internationalization laboratory assignment would first need to be revised. In addition, there is a need for a larger sample of subjects, subjects with differing programming experiences (ranging from students to professionals), and subjects coming from different universities and areas of the world. The biggest change to the internationalization laboratory assignment is the use of a different programming language such as Java or C++. All of these changes will decrease the threats to validity and help the measurements of students' perceptions.

E.11. Synopsis

The goal is to determine whether the exposure to internationalization activities changed the students' perceptions of software internationalization. The change in students' perceptions of software internationalization based on the research questions concludes that:

- There is a statistically significant change between the students' perceptions of "Internationalization is a familiar topic to me" after the laboratory assignment.
- There is not a statistically significant change between the students' perceptions of "Internationalization will be important to my future career" after the laboratory assignment.
- There is not a statistically significant change between the students' perceptions of "Internationalization should be taught more at Cal Poly" after the laboratory assignment.
- There is not a statistically significant change between the students' perceptions of "Learning internationalization" after the laboratory assignment.
- There is not a statistically significant change between the students' perceptions of "Implementing, deploying software that supports internationalization" after the laboratory assignment.
- There is not a statistically significant change between the students' perceptions of "Fixing, maintaining software that supports internationalization" after the laboratory assignment.

The section concludes that there is no statistically significant change in students' perceptions except that students were more familiar with software internationalization. This lack of significance could be an result of the small sample size and other threats of validity. Like Appendix E, the section also

concludes there is a need for students to learn about software internationalization. Since there was no consensus on the other research questions, students might have still been unaware of what software internationalization really entails.

Although the internationalization laboratory assignment can be seen as a failure, there are a number of questions that can be drawn from the results:

- Why were there no statistically significant changes to students' perceptions for the other research questions?
- What parts of the internationalization laboratory assignment should be kept, removed, or modified to change the students' perceptions?
- What aspects of software internationalization should be included in the internationalization laboratory assignment to change the students' perceptions?
- Would a framework for software internationalization assist in the development of the internationalization laboratory assignment?

If a framework was in place for use on the internationalization laboratory assignment, would this have changed the students' perceptions of software internationalization?

There is a need for students to learn about software internationalization. The result does not imply a generalization to other contexts, given the multiple factors that could have biased the results. These results show that Cal Poly students were not meeting the demand in, nor were they well-educated about software internationalization.

APPENDIX F. STUDENTS' CURRENT PERCEPTIONS DOCUMENTS AND DATA

F.1. Figures

<p style="text-align: center;">INFORMED CONSENT TO PARTICIPATE IN A STUDY OF SOFTWARE INTERNATIONALIZATION AND GLOBALIZATION FROM AN ACADEMIC AND INDUSTRY STANDPOINT</p> <p>A research project on software internationalization and globalization is being conducted by John Vu and Dr. Clark Turner in the Department of Computer Science at Cal Poly, San Luis Obispo. The purpose of the study is to determine if computer science and software engineering students will gain a better appreciation of internationalization and globalization after a laboratory assignment.</p> <p>You are being asked to take part in this study by completing the attached/enclosed questionnaire. You are asked to answer all of the questions of the questionnaire as honestly and truthfully as you can. Your participation will take approximately 10 minutes. Please be aware that you are not required to participate in this research and you may discontinue your participation at any time without penalty. You may also omit any items on the questionnaire(s) you prefer not to answer.</p> <p>There should not be any risks associated with participation in this study.</p> <p>Your responses will be provided confidentially to protect your privacy. Only the researchers, John Vu and Dr. Clark Turner, will be reading your questionnaires. Potential benefits associated with the study may include a better understanding of the effects of CSC 300 on student perceptions about software internationalization and globalization.</p> <p>If you have questions regarding this study or would like to be informed of the results when the study is completed, please feel free to contact John Vu at (408) 910- or Dr. Clark Turner at (805) 756-. If you have questions or concerns regarding the manner in which the study is conducted, you may contact Dr. Steve Davis, Chair of the Cal Poly Human Subjects Committee, at 756-, @calpoly.edu, or Dr. Susan Opava, Dean of Research and Graduate Programs, at 756-, @calpoly.edu.</p> <p>If you agree to voluntarily participate in this research project as described, please indicate your agreement by signing below and completing and returning the attached questionnaire. Please retain a copy of this consent cover form for your reference, and thank you for your participation in this research.</p> <table border="0" style="width: 100%;"><tr><td style="width: 50%; text-align: center;">_____ Signature of Volunteer</td><td style="width: 50%; text-align: center;">_____ Date</td></tr><tr><td style="width: 50%; text-align: center;">_____ Signature of Researcher</td><td style="width: 50%; text-align: center;">_____ Date</td></tr></table>		_____ Signature of Volunteer	_____ Date	_____ Signature of Researcher	_____ Date
_____ Signature of Volunteer	_____ Date				
_____ Signature of Researcher	_____ Date				

Figure 36 - Informed consent form for a survey on students' perception.

Internationalization Questionnaire

Internationalization

Internationalization is the process of producing an application that can be localized for a particular country without any changes to the program code. Internationalized applications store their text in external resources, and use locale-sensitive utilities for formatting and collation.

The following questionnaire is for a thesis field analysis about internationalization. You can choose to **OPT OUT** of the **OPTIONAL** questionnaire by not placing any answers. The answers you give will also be kept **CONFIDENTIAL** and will only be used for research. Please take your time answering **all** the questions as truthfully as you can.

Name	
Year	
Major	
What other languages can you read and write in?	
What courses have you taken at Cal Poly that covers internationalization?	

How do you agree with each of the following statements?

	Agree		Neutral		Disagree	
Internationalization is a familiar topic to me	☐	☐	☐	☐	☐	☐
Internationalization will be important to my future career	☐	☐	☐	☐	☐	☐
Internationalization is easy to learn, implement, and deploy	☐	☐	☐	☐	☐	☐
Internationalization is easy to fix, maintain, and debug	☐	☐	☐	☐	☐	☐
I am prepared to internationalize any software product	☐	☐	☐	☐	☐	☐

Figure 37 - Internationalization survey questionnaire on students' perception.

F.2. Tables

Student	i18n is a familiar topic to me	i18n will be important to my future career	i18n is easy to learn, implement, and deploy	i18n is easy to fix, maintain, and debug	I am prepared to internationalize any product.
1	Netural	Netural	Somewhat Agree	Somewhat Agree	Netural
2	Somewhat Disagree	Netural	Agree	Agree	Netural
3	Disagree	Disagree	Agree	No answer	Somewhat Disagree
4	Netural	Somewhat Disagree	Netural	Somewhat Agree	Netural
5	Agree	Netural	Agree	Agree	Agree
6	Somewhat Disagree	Somewhat Agree	Somewhat Agree	Netural	Somewhat Disagree
7	Somewhat Disagree	Somewhat Disagree	Somewhat Agree	Somewhat Agree	Somewhat Agree
8	Disagree	Disagree	Somewhat Disagree	Somewhat Disagree	Disagree
9	Somewhat Agree	Netural	Agree	Agree	Netural
10	Netural	Somewhat Disagree	Netural	Somewhat Agree	Somewhat Agree
11	Somewhat Agree	Netural	Somewhat Agree	Agree	Agree
12	Somewhat Disagree	Somewhat Disagree	Netural	Somewhat Agree	Netural
13	Disagree	Disagree	Netural	Agree	Disagree
14	Somewhat Disagree	Somewhat Disagree	Agree	Agree	Netural
15	Agree	Disagree	Agree	Agree	Agree
16	Netural	Somewhat Disagree	Agree	Agree	Somewhat Agree
17	Somewhat Agree	Somewhat Disagree	Somewhat Agree	Netural	Somewhat Agree
18	Somewhat Disagree	Somewhat Disagree	Netural	Netural	Somewhat Disagree
19	Agree	Netural	Somewhat Agree	Netural	Somewhat Agree

Table 13 - Survey questionnaire responses.

APPENDIX G. STUDENTS' PERCEPTIONS THROUGH A LABORATORY ASSIGNMENT

DOCUMENTS AND DATA

G.1. Figures

<p style="text-align: center;">INFORMED CONSENT TO PARTICIPATE IN A STUDY OF SOFTWARE INTERNATIONALIZATION AND GLOBALIZATION FROM AN ACADEMIC AND INDUSTRY STANDPOINT</p> <p>A research project on software internationalization and globalization is being conducted by John Vu and Dr. Clark Turner in the Department of Computer Science at Cal Poly, San Luis Obispo. The purpose of the study is to determine if computer science and software engineering students will gain a better appreciation of internationalization and globalization after a laboratory assignment.</p> <p>You are being asked to take part in this study by completing the attached/enclosed questionnaire. You are asked to answer all of the questions of the questionnaire as honestly and truthfully as you can. Your participation will take approximately 10 minutes. Please be aware that you are not required to participate in this research and you may discontinue your participation at any time without penalty. You may also omit any items on the questionnaire(s) you prefer not to answer.</p> <p>There should not be any risks associated with participation in this study.</p> <p>Your responses will be provided confidentially to protect your privacy. Only the researchers, John Vu and Dr. Clark Turner, will be reading your questionnaires. Potential benefits associated with the study may include a better understanding of the effects of CSC 300 on student perceptions about software internationalization and globalization.</p> <p>If you have questions regarding this study or would like to be informed of the results when the study is completed, please feel free to contact John Vu at (408) 910- or Dr. Clark Turner at (805) 756-. If you have questions or concerns regarding the manner in which the study is conducted, you may contact Dr. Steve Davis, Chair of the Cal Poly Human Subjects Committee, at 756-, @calpoly.edu, or Dr. Susan Opava, Dean of Research and Graduate Programs, at 756-, @calpoly.edu.</p> <p>If you agree to voluntarily participate in this research project as described, please indicate your agreement by signing below and completing and returning the attached questionnaire. Please retain a copy of this consent cover form for your reference, and thank you for your participation in this research.</p> <table><tr><td>_____ Signature of Volunteer</td><td>_____ Date</td></tr><tr><td>_____ Signature of Researcher</td><td>_____ Date</td></tr></table>		_____ Signature of Volunteer	_____ Date	_____ Signature of Researcher	_____ Date
_____ Signature of Volunteer	_____ Date				
_____ Signature of Researcher	_____ Date				

Figure 38 - Informed consent form for a case study on the change of students' perception.

Internationalization Questionnaire (Pre-Lab)

Internationalization

Internationalization is the process of producing an application that can be localized for a particular country without any changes to the program code. Internationalized applications store their text in external resources, and use locale-sensitive utilities for formatting and collation.

The following questionnaire is for a thesis field analysis about internationalization. You can choose to **OPT OUT** of the **OPTIONAL** questionnaire by not placing any answers. The answers you give will also be kept **CONFIDENTIAL** and will only be used for research. Please take your time answering **ALL** the questions as truthfully as you can.

Name (used for comparison)	
Year (circle)	1 st 2 nd 3 rd 4 th 5 th 6 th +
Major	
Have you used Adobe Flex Builder before? If so, at what extent?	
What other languages, aside from English, can you read and write in?	
What courses have you taken at Cal Poly that covers internationalization?	

How do you agree with each of the following statements?

	Agree		Neutral		Disagree
Internationalization is a familiar topic to me	☐	☐	☐	☐	☐
Internationalization will be important to my future career	☐	☐	☐	☐	☐
Internationalization should be taught more at Cal Poly	☐	☐	☐	☐	☐

How do you feel about each of the following statements?

	Easy		Neither		Difficult
Learning internationalization	☐	☐	☐	☐	☐
Implementing, deploying software that supports internationalization	☐	☐	☐	☐	☐
Fixing, maintaining software that supports internationalization	☐	☐	☐	☐	☐

Figure 39 - Pre-lab internationalization survey questionnaire on students' perception.

Internationalization Lab

Purpose:

The world is getting smaller and smaller each day through various innovations that connect us. Programs like Skype allows us to see, hear, and talk with people around the world for free. Programs also allow you to change the language to cater to your environment without charge. Although it's a benefit to tailor our software towards people of different languages, the question is whether it's worth the investment of money and time. The Internationalization Laboratory Assignment (ILA) explores the topic of internationalization where students will work on a project to understand the techniques and methods used to cater to people of different languages.

Procedure:

Pre-Lab

In preparation for the Internationalization Project, you will need to install Adobe® Flex™ Builder (built on Eclipse) by downloading the program from: www.adobe.com/products/flex

You can either use the program on a 30-day trial or:

- Use the serial number: [REDACTED]
- Obtain a free serial number with an uploaded image of your Cal Poly identification card by going to: <http://www.flexregistration.com>

Preliminary Questions

1. What are Localization, Globalization, and Internationalization?
2. What are the abbreviations for Localization, Globalization, and Internationalization?
3. What languages are you most fluent in (speaking, writing, talking, hearing)?
4. What language is spoken the most around the world?
5. What courses have you taken at Cal Poly that has covered internationalization? How so?
6. In your opinion, should topics and labs about internationalization be utilized in the Computer Science Department? Why or why not?

Software Engineering Code of Ethics

What are the top three (3) provisions in the Software Engineering Code of Ethics that apply to this lab? Give a brief justification for each that you choose.

Internationalization Project

Make sure that your group has completed the Pre-Lab portion of the assignment.

Your group will be taking an existing program to make available in multiple languages. Creating this infrastructure is one of the aspects of software internationalization.

Download the existing program from: <http://www.calpoly.edu> [REDACTED]
Import the existing program as a Flex project by going to: *File > Import > Flex Project ...*

Figure 40 - First page of the Internationalization Laboratory Assignment.

The existing program takes in keywords from the user to retrieve images from Flickr. Your group is in charge of changing all of the text fields and labels based on the user's language selection. The final product must be available in the following languages:

• English (U.S.)	• Hebrew	• Polish
• Arabic	• Hungarian	• Portuguese
• Chinese (Simp.)	• Korean	• Russian
• Chinese (Trad.)	• Italian	• Spanish
• Dutch	• German	• Swedish
• French	• Japanese	• Turkish
• Greek	• Norwegian	

The final product of the program must look very similar to the program displayed at:

<http://www.calpoly.edu>

You can start by reading from Adobe Flex Builder on localizing Flex Applications from:

http://livedocs.adobe.com/flex/3/html/help.html?content=110n_6.html

Within the *ILA System* project, there is a folder named *bin-debug*. The folder contains multiple *swf* files with corresponding *html* files. When a user opens the *html* file, the *swf* file is loaded. Place the contents within the *bin-debug* folder onto a hornet, falcon, or polylog1 to present to the class. Make sure the program can be accessed by Dr. Turner.

Presentation

For a maximum of five minutes, each group member must discuss **their contribution** to the Internationalization Laboratory Assignment. Each member must speak at least once covering what techniques or tools used, difficulties encountered, and experience with Adobe Flex Builder.

Concluding Questions

Write a concise paragraph on each of the following concluding questions:

1. What did you learn or experience in the lab that was new to you?
2. What does the issue of internationalization mean to you as a developer?
3. What does the issue of internationalization mean to members of society?
4. What other issues could you have covered in the lab? Why?
5. What language was most difficult to translate?

Deliverables:

On or before the date due, please provide a group lab report in professional format.

Include the following:

1. Individual responses to the preliminary questions
2. Individual responses to the Software Engineering Code of Ethics
3. Individual responses to the concluding questions
4. Appendices
 - a. Web address and screenshot of the Internationalization Laboratory Assignment
 - b. Printout of the Internationalization Laboratory Assignment MXML file
 - c. Printout of the Internationalization Laboratory Assignment language file(s)

Figure 41 - Second page of the Internationalization Laboratory Assignment.

Internationalization Questionnaire (Post-Lab)

Internationalization

Internationalization is the process of producing an application that can be localized for a particular country without any changes to the program code. Internationalized applications store their text in external resources, and use locale-sensitive utilities for formatting and collation.

The following questionnaire is for a thesis field analysis about internationalization. You can choose to **OPT OUT** of the **OPTIONAL** questionnaire by not placing any answers. The answers you give will also be kept **CONFIDENTIAL** and will only be used for research. Please take your time answering **ALL** the questions as truthfully as you can.

Name (used for comparison)						
Year (circle)	1 st	2 nd	3 rd	4 th	5 th	6 th +

After working on the lab, how do you agree with each of the following statements?

	Agree		Neutral		Disagree
Internationalization is a familiar topic to me	☐	☐	☐	☐	☐
Internationalization will be important to my future career	☐	☐	☐	☐	☐
Internationalization should be taught more at Cal Poly	☐	☐	☐	☐	☐

After working on the lab, how do you feel about each of the following statements?

	Easy		Neither		Difficult
Learning internationalization	☐	☐	☐	☐	☐
Implementing, deploying software that supports internationalization	☐	☐	☐	☐	☐
Fixing, maintaining software that supports internationalization	☐	☐	☐	☐	☐

Figure 42 - Post-lab internationalization survey questionnaire on students' perception.

G.2. Tables

Student	i18n is a familiar topic to me	i18n will be important to my future career	i18n should be taught more at Cal Poly	Learning i18n	Implementing, deploying software that supports i18n	Fixing, maintaining software that supports i18n
1	Somewhat Disagree	Agree	Agree	Somewhat Easy	Somewhat Easy	Somewhat Easy
2	Neutral	Somewhat Agree	Agree	Somewhat Easy	Somewhat Difficult	Somewhat Difficult
3	Somewhat Agree	Disagree	Neutral	Neither	Difficult	Somewhat Difficult
4	Agree	Agree	Agree	Neither	Neither	Neither
5	Neutral	Neutral	Somewhat Agree	Neither	Somewhat Difficult	Somewhat Difficult
6	Disagree	Agree	Neutral	Somewhat Difficult	Difficult	Difficult
7	Somewhat Agree	Somewhat Agree	Somewhat Agree	Neither	Somewhat Easy	Somewhat Easy
8	Neutral	Neutral	Neutral	Neither	Neither	Neither
9	Neutral	Somewhat Agree	Somewhat Agree	Neither	Somewhat Difficult	Neither
10	Somewhat Agree	Agree	Agree	Difficult	Neither	Difficult
11	Somewhat Disagree	Somewhat Agree	Somewhat Agree	Neither	Somewhat Difficult	Neither
12	Disagree	Neutral	Somewhat Agree	Somewhat Easy	Somewhat Difficult	Somewhat Easy
13	Disagree	Somewhat Disagree	Neutral	Difficult	Somewhat Difficult	Neither
14	Neutral	Neutral	Somewhat Disagree	Somewhat Easy	Somewhat Easy	Somewhat Easy
15	Disagree	Somewhat Agree	Neutral	Difficult	Difficult	Somewhat Difficult
16	Agree	Neutral	Neutral	Somewhat Easy	Somewhat Easy	Somewhat Difficult
17	Somewhat Disagree	Neutral	Neutral	Somewhat Easy	Somewhat Difficult	Neither
18	Neutral	Agree	Neutral	Neither	Neither	Somewhat Difficult
19	Neutral	Somewhat Agree	Neutral	Somewhat Difficult	Somewhat Difficult	Neither
20	Somewhat Agree	Somewhat Agree	Somewhat Agree	Neither	Difficult	Difficult
21	Agree	Somewhat Agree	Somewhat Agree	Somewhat Difficult	Somewhat Difficult	Somewhat Difficult
22	Somewhat Disagree	Agree	Somewhat Agree	Difficult	Difficult	Difficult
23	Somewhat Disagree	Neutral	Neutral	Neither	Somewhat Difficult	Neither
24	Disagree	Neutral	Neutral	Neither	Neither	Neither
25	Somewhat Agree	Somewhat Disagree	Neutral	Neither	Neither	Neither

Table 14 - Pre-lab survey results from the survey questionnaire.

Student	i18n is a familiar topic to me	i18n will be important to my future career	i18n should be taught more at Cal Poly	Learning i18n	Implementing, deploying software that supports i18n	Fixing, maintaining software that supports i18n
1	Somewhat Agree	Agree	Agree	Somewhat Easy	Somewhat Easy	Somewhat Easy
2	Neutral	Disagree	Somewhat Disagree	Somewhat Difficult	Neither	Somewhat Difficult
3	Agree	Neutral	Somewhat Disagree	Easy	Neither	Somewhat Difficult
4	Neutral	Neutral	Somewhat Agree	Neither	Somewhat Difficult	Somewhat Difficult
5	Somewhat Disagree	Neutral	Neutral	Somewhat Easy	Somewhat Easy	Somewhat Easy
6	Neutral	Somewhat Agree	Somewhat Agree	Neither	Somewhat Difficult	Somewhat Difficult
7	Somewhat Agree	Somewhat Agree	Neutral	Easy	Somewhat Easy	Somewhat Easy
8	Neutral	Somewhat Agree	Somewhat Agree	Somewhat Difficult	Somewhat Difficult	Neither
9	Somewhat Agree	Somewhat Agree	Neutral	Somewhat Easy	Somewhat Difficult	Neither
10	Disagree	Somewhat Agree	Somewhat Disagree	Difficult	Difficult	Difficult
11	Somewhat Disagree	Neutral	Somewhat Agree	Neither	Neither	Neither
12	Somewhat Disagree	Somewhat Agree	Agree	Somewhat Easy	Somewhat Easy	Somewhat Easy
13	Neutral	Somewhat Disagree	Neutral	Difficult	Difficult	Difficult
14	Agree	Neutral	Disagree	Easy	Easy	Somewhat Easy
15	Somewhat Agree	Neutral	Neutral	Somewhat Easy	Somewhat Easy	Somewhat Easy
16	Agree	Somewhat Disagree	Disagree	Neither	Somewhat Difficult	Difficult
17	Somewhat Agree	Neutral	Neutral	Somewhat Easy	Somewhat Easy	Neither
18	Somewhat Agree	Neutral	Disagree	Neither	Easy	Easy
19	Somewhat Agree	Agree	Somewhat Agree	Neither	Somewhat Difficult	Neither
20	Somewhat Agree	Somewhat Agree	Neutral	Neither	Somewhat Difficult	Somewhat Difficult
21	Agree	Agree	Somewhat Agree	Somewhat Difficult	Difficult	Neither
22	Neutral	Somewhat Agree	Neutral	Somewhat Difficult	Somewhat Difficult	Somewhat Difficult
23	Neutral	Somewhat Disagree	Somewhat Agree	Neither	Neither	Neither
24	Somewhat Disagree	Neutral	Neutral	Neither	Neither	Neither
25	Somewhat Agree	Somewhat Disagree	Neutral	Neither	Neither	Neither

Table 15 - Post-lab survey results from the survey questionnaire.

Student	i18n is a familiar topic to me	i18n will be important to my future career	i18n should be taught more at Cal Poly	Learning i18n	Implementing, deploying software that supports i18n	Fixing, maintaining software that supports i18n
1	Towards Agree (2)	No Change	No Change	No Change	No Change	No Change
2	No Change	Towards Disagree (3)	Towards Disagree (3)	Towards Difficult (2)	Towards Easy (1)	No Change
3	Towards Agree (1)	Towards Agree (2)	Towards Disagree (1)	Towards Easy (2)	Towards Easy (2)	No Change
4	Towards Disagree (2)	Towards Disagree (2)	Towards Disagree (1)	No Change	Towards Difficult (1)	Towards Difficult (1)
5	Towards Disagree (1)	No Change	Towards Disagree (1)	Towards Easy (1)	Towards Easy (2)	Towards Easy (2)
6	Towards Agree (2)	Towards Disagree (1)	Towards Agree (1)	Towards Easy (1)	Towards Easy (1)	Towards Easy (1)
7	No Change	No Change	Towards Disagree (1)	Towards Easy (2)	No Change	No Change
8	No Change	Towards Agree (1)	Towards Agree (1)	Towards Difficult (1)	Towards Difficult (1)	No Change
9	Towards Agree (1)	No Change	Towards Disagree (1)	Towards Easy (1)	No Change	No Change
10	Towards Disagree (3)	Towards Disagree (1)	Towards Disagree (3)	No Change	Towards Difficult (2)	No Change
11	No Change	Towards Disagree (1)	No Change	No Change	Towards Easy (1)	No Change
12	Towards Agree (1)	Towards Agree (1)	Towards Agree (1)	No Change	Towards Easy (2)	No Change
13	Towards Agree (2)	No Change	No Change	No Change	Towards Difficult (1)	Towards Difficult (2)
14	Towards Agree (2)	No Change	Towards Disagree (1)	Towards Easy (1)	Towards Easy (1)	No Change
15	Towards Agree (3)	Towards Disagree (1)	No Change	Towards Easy (3)	Towards Easy (3)	Towards Easy (2)
16	No Change	Towards Disagree (1)	Towards Disagree (2)	Towards Difficult (1)	Towards Difficult (2)	Towards Difficult (1)
17	Towards Agree (2)	No Change	No Change	No Change	Towards Easy (2)	No Change
18	Towards Agree (1)	Towards Disagree (2)	Towards Disagree (2)	No Change	Towards Easy (2)	Towards Easy (3)
19	Towards Agree (1)	Towards Agree (1)	Towards Agree (1)	Towards Easy (1)	No Change	No Change
20	No Change	No Change	Towards Disagree (1)	No Change	Towards Easy (1)	Towards Easy (1)
21	No Change	Towards Agree (1)	No Change	No Change	Towards Difficult (1)	Towards Easy (1)
22	Towards Agree (1)	Towards Disagree (1)	Towards Disagree (1)	Towards Easy (1)	Towards Easy (1)	Towards Easy (1)
23	Towards Agree (1)	Towards Disagree (1)	Towards Agree (1)	No Change	Towards Easy (1)	No Change
24	Towards Agree (1)	No Change	No Change	No Change	No Change	No Change
25	No Change	No Change	No Change	No Change	No Change	No Change

Table 16 - Change in survey results from the survey questionnaire.

APPENDIX H. INDUSTRY SURVEY QUESTIONNAIRE DOCUMENTS AND DATA

H.1. Figures

Google

Evaluation Study of a Proposed Education Framework for Software Internationalization

[Exit this survey](#)

1. Informed Consent

Informed Consent to Participate in an Evaluation Study of a Proposed Educational Framework for Software Internationalization by the Industry

DESCRIPTION OF THE STUDY

An evaluation study of a proposed educational framework for software internationalization is being conducted by John Vu and Dr. Clark Turner of the Department of Computer Science at California Polytechnic State University in San Luis Obispo. The purpose of this study is for representatives of the industry to evaluate a proposed educational framework for software internationalization. The results from this study will be used for a master's thesis.

Potential benefits of this study include:

1. Meet the software industry's need for graduates with a software internationalization background.
2. Provide a relevant framework to support software internationalization coursework.

CONDITIONS OF PARTICIPATION AND RISKS

You are being asked to take part in this study by completing a survey about software internationalization. Your participation will take approximately 10 minutes and is greatly appreciated. Although you may omit any items that you prefer not to answer, you are asked to answer all of the questions as honestly and truthfully as you can. Please be aware that you are not required to participate in this evaluation study and you may discontinue your participation at any time. There should not be any risks or conflicts of interest associated with the participation in this study.

CONFIDENTIALITY AND CONTACT

Your responses are anonymous to protect your privacy. If you have any questions regarding this study or would like to be informed of the results when the study is completed, please contact John Vu at (408) 910- or jvu@calpoly.edu. You may also contact Dr. Clark Turner at (805) 756- or csturner@calpoly.edu.

HUMAN SUBJECTS APPROVAL

This evaluation study was approved by the Human Subjects Committee on June 11, 2009 at California Polytechnic State University in San Luis Obispo, California. If you have any questions, please contact Dr. Steve Davis, Chair of the Human Subjects Committee, at (805) 756- or sdavis@calpoly.edu.

[Next](#)

Figure 43 - Informed consent page of the industry survey.

Evaluation Study of a Proposed Education Framework for Software Internationalization

[Exit this survey](#)

2. Demographics

The following questions will be used primarily for comparative analysis with others in the industry. Although you may omit any items that you prefer not to answer, you are asked to answer all of the questions as honestly and truthfully as you can. If you do not understand a question, please omit the item.

1. Company and Location (Optional)

2. Years Working in the Industry (Optional)

3. Title (Optional)

4. Highest Level of Education (Optional)

☐ High School Diploma, No Degree

☐ Associate's Degree, Vocational

☐ Bachelor's Degree

☐ Master's Degree

☐ Doctor of Philosophy

Other (please specify)

5. Group (Optional)

☐ Corporate, Executive

☐ Manager, Supervisor

☐ Internationalization, Localization

☐ Experience Design, Usability

☐ Development, Engineering, Research

☐ Quality Assurance, Testing

☐ Human Resources

Other (please specify)

Figure 44 - Demographics page of the industry survey.

Evaluation Study of a Proposed Education Framework for Software Internationalization

[Exit this survey](#)

4. Survey (Page 2)

The following questions will be used primarily for comparative analysis with others in the industry. Although you may omit any items that you prefer not to answer, you are asked to answer all of the questions as honestly and truthfully as you can. If you do not understand a question, please omit the item.

DEFINITIONS:
 "Software Internationalization" is the process of producing an application that can be localized for a particular country without making any changes to the program code.
 "Localization" is the process of adapting software for a particular geographical region.

1. In order for computer science and software engineering graduates to be prepared for industry, students ...

	Strongly Agree	Somewhat Agree	Neither	Somewhat Disagree	Strongly Disagree
Must BE AWARE of possible technologies and tools used for language translation and text representation	☐	☐	☐	☐	☐
Must APPLY the possible technologies and tools used for language translation and text representation	☐	☐	☐	☐	☐
Must BE AWARE of how the direction of text is represented in various languages	☐	☐	☐	☐	☐
Must APPLY the possible technologies and tools to represent the direction of text in various languages	☐	☐	☐	☐	☐
Must BE AWARE of how text is sorted in various languages	☐	☐	☐	☐	☐
Must APPLY the possible technologies and tools to sort text in various languages	☐	☐	☐	☐	☐
Must BE AWARE of how dates and times are used in various cultures and countries	☐	☐	☐	☐	☐
Must APPLY the possible technologies and tools to decipher the date and time for various cultures and countries	☐	☐	☐	☐	☐
Must BE AWARE of how the number/currency systems are different for various cultures and countries	☐	☐	☐	☐	☐
Must APPLY the possible technologies for the number/currency systems for various cultures and countries	☐	☐	☐	☐	☐
Must BE AWARE of the social responsibilities and ethics of producing software that supports internationalization	☐	☐	☐	☐	☐

Figure 46 - Second page of the industry survey.

Figure 47 - Confirmation page of the industry survey.

$$H_o : \pi_1 = \pi_2 = \pi_3 = \pi_4 = \pi_5 = 0.20$$

Let π_{0i} = null proportion value in cell i = 0.20 in this example

Let π_{1i} = alternative proportion value in cell i. These values could vary from cell to cell.

Example: Possible alternative values are: 0.15, 0.25, 0.30, 0.10, 0.20

$$\text{Let } w = \sqrt{\sum_{i=1}^m \frac{(\pi_{0i} - \pi_{1i})^2}{\pi_{0i}}} \text{ be the effect size where } m = 5 \text{ cells in this example.}$$

For every possible alternative set of proportions, the effect size can be calculated. Since we do not know what the alternative will be, Cohen (1987) recommends letting $w = 0.1$ be a “small” effect size, 0.3 be a medium effect size, and 0.5 be a large effect size.

So, you can find a “worst case” sample size using the small effect size.

To find the power for a given sample size:

1. Define α for the test. Example: $\alpha = 0.05$
2. Find the $1 - \alpha$ quantile of the Chi-square distribution. Example: The 0.95 quantile of the Chi-square distribution with 4 df (since 5 cells) is 9.48773 (from Minitab).
3. Find $\lambda = Nw^2$, the non-centrality parameter for a Chi-Square distribution. Example: Let $N=100$, then for a small effect size ($w = 0.1$), $\lambda = 1$.
4. Find β = cumulative probability of the non-central Chi-Square with 4 df using the critical value found above in #2. Example: The cumulative probability of a non-central Chi-Square with $\lambda = 1$, $df = 4$, at a critical value of 9.48773 is 0.8945.
5. Power = $1 - \beta$. Example: Power = 0.1055.

The power for a sample size of 100, with an effect size of 0.3 is 0.6635, and for an effect size of 0.5 is 0.98936.

Usually, we want the power to be between 0.80 and 0.90. For power of 0.80 (when $df=4$, and $\alpha = 0.05$) the non-centrality parameter needs to be 12, which means you need a sample size of 1200 if the effect size is small ($w=0.1$). For power of 0.90, the non-centrality parameter needs to be 15, which means you need a sample size of 1500 if the effect size is small. If the effect size is medium ($w=0.30$), then you need about $n=134$ for 0.80 power, and 167 for 0.90 power. If the effect size is large ($w=0.50$), then you need about 48 for 0.80 power, and 60 for 0.90 power.

Figure 48 - Discussion and details with Dr. Karen McGaughey (1 of 3)

Example:

$$H_o : \pi_1 = \pi_2 = \pi_3 = \pi_4 = \pi_5 = 0.20$$

Suppose: $H_A : \pi_1 = 0.15, \pi_2 = 0.25, \pi_3 = 0.30, \pi_4 = 0.20, \pi_5 = 0.10$

Then, the effect for this case is:

$$w = \sqrt{\sum_{i=1}^m \frac{(\pi_{0i} - \pi_{1i})^2}{\pi_{0i}}} = \sqrt{\frac{(0.20 - 0.15)^2}{0.20} + \frac{(0.20 - 0.25)^2}{0.20} + \frac{(0.20 - 0.30)^2}{0.20} + \frac{(0.20 - 0.20)^2}{0.20} + \frac{(0.20 - 0.10)^2}{0.20}} \\ = 0.3536$$

So, in this example the effect is a “medium” size effect. Setting $w = 0.30$, the values in the alternative would be approximately equal to those in my example above. The size of w is based on how different the alternative proportions are from the null proportions. Since there are an infinite number of possible alternatives, Cohen just defines “ w ” to get around having to decide what the value of each alternative proportion should be.

The “worst case” sample size is:

The sample size must be largest to detect the smallest effect, or the smallest differences from the null values. To detect a difference of only $w = 0.1$, it will require many more observations than to detect an effect of $w = 0.50$. So, if you wish to know what the largest sample size required is, you compute that using the smallest effect which you would expect to see in the study. Collecting 1500 observations will enable you to detect an effect of $w=0.1$ with power of 0.90.

The power of the study is the probability that you reject H_o , when in fact H_o is false. In other words, if H_o is not true, then we should reject it. For example, if $w=0.10$, then clearly H_o is false. (Note: if $w=0$, then H_o would be true.) A power value of 0.90, says that 90 times out of 100, the statistical test will reject when the observed difference, or observed effect is $w=0.10$. The higher the power, the better. A power of

Figure 49 - Discussion and details with Dr. Karen McGaughey (2 of 3)

0.10, would mean that only 10 times out of 100 would we correctly reject H_0 . Power and sample size are related. The higher the sample size, the higher the power.

Beta = 1-power. Beta is the probability of making a Type II error. There is no p-value computation here. We set alpha (probability of making a Type I error) at 0.05 (usually). So, if we were actually conducting the GOF test, our p-value would need to be less than alpha to reject H_0 .

Figure 50 - Discussion and details with Dr. Karen McGaughey (3 of 3)

H.2. Tables

Abbreviation	Effect Size (w) Observed	Type of Effect	Non-Centrality Parameter for a Chi-Square Distribution	Sample size required for a power effect of .9
Learn	0.95575	Large	252.116	16.42101633
Apply	0.97055	Large	260.924	15.9241657
Integrate	0.85613	Large	202.297	20.46494967
Course	0.46022	Medium	58.4565	70.82186687
Abroad	0.49619	Medium	67.4599	60.92512443
Technology-Aware	1.16629	Large	371.341	11.02761009
Technology-Apply	0.8813	Large	212.037	19.31270083
Definitions	0.99116	Large	269.175	15.26886677
Standards	0.7121	Large	137.926	29.58097878
Colors	0.60794	Large	101.636	40.58586762
Visual	0.69234	Large	132.297	31.29320261
Noise	0.57883	Large	91.8029	44.76981792
Laws	0.76246	Large	160.449	25.8025472
Translation-Aware	0.94595	Large	232.654	16.76310134
Translation-Apply	0.70171	Large	127.039	30.46314376
Direction-Aware	1.02629	Large	272.795	14.24144422
Direction-Apply	0.78181	Large	157.698	24.54062823
Collation-Aware	0.93999	Large	229.731	16.97639377
Collation-Apply	0.73578	Large	139.674	27.70729271
Dates-Aware	1.10895	Large	318.51	12.19743254
Dates-Apply	0.78155	Large	158.201	24.55740226
Currency-Aware	1.03324	Large	276.502	14.05053481
Currency-Apply	0.78104	Large	157.388	24.588977
Ethics	0.74182	Large	143.077	27.25806452

Table 17 – Results calculated for the power and effect size of the survey results.

<u>City</u>	<u>State</u>	<u>Country</u>	<u>Continent</u>	<u>Count</u>
Buenos Aires		Argentina	South America	1
		Brazil	South America	2

Table 18 - Locations of participants of the industry survey questionnaire in South America.

<u>City</u>	<u>State</u>	<u>Country</u>	<u>Continent</u>	<u>Count</u>
Dalian		China	Asia	1
Hong Kong		China	Asia	1
		China	Asia	1
Bangalore		India	Asia	1
		Israel	Asia	1
Tokyo		Japan	Asia	1
		Japan	Asia	2
Lahore		Pakistan	Asia	1
		Pakistan	Asia	1
Colombo		Sri Lanka	Asia	1
		Thailand	Asia	1

Table 19 - Locations of participants of the industry survey questionnaire in Asia.

<u>City</u>	<u>State</u>	<u>Country</u>	<u>Continent</u>	<u>Count</u>
		Austria	Europe	1
Prague		Czech Republic	Europe	1
		Denmark	Europe	1
		Estonia	Europe	1
		France	Europe	1
		Germany	Europe	4
Szeged		Hungary	Europe	1
		Hungary	Europe	1
Dublin		Ireland	Europe	3
Limerick		Ireland	Europe	1
Eindhoven		Netherlands	Europe	1
		Netherlands	Europe	1
Oslo		Norway	Europe	1
		Norway	Europe	1
		Romania	Europe	1
Saint Petersburg		Russia	Europe	1
		Russia	Europe	1
Madrid		Spain	Europe	1
		Spain	Europe	1
		Sweden	Europe	2
Basel		Switzerland	Europe	1
		Switzerland	Europe	3
Bristol		UK	Europe	1
London		UK	Europe	1
		UK	Europe	2
		Ukraine	Europe	1

Table 20 - Locations of participants of the industry survey questionnaire in Europe.

<u>City</u>	<u>State</u>	<u>Country</u>	<u>Continent</u>	<u>Count</u>
Vancouver	BC	Canada	North America	1
Montreal		Canada	North America	1
Ottawa		Canada	North America	1
		Canada	North America	1
Huntsville	AL	USA	North America	1
Berkeley	CA	USA	North America	1
Brisbane	CA	USA	North America	1
Cupertino	CA	USA	North America	2
Goleta	CA	USA	North America	1
Irvine	CA	USA	North America	1
Livermore	CA	USA	North America	1
Los Angeles	CA	USA	North America	2
Menlo Park	CA	USA	North America	1
Mountain View	CA	USA	North America	2
Palo Alto	CA	USA	North America	1
Redwood City	CA	USA	North America	1
San Bruno	CA	USA	North America	1
San Diego	CA	USA	North America	2
San Francisco	CA	USA	North America	7
San Jose	CA	USA	North America	3
San Leandro	CA	USA	North America	1
San Mateo	CA	USA	North America	2
San Ramon	CA	USA	North America	1
Santa Barbara	CA	USA	North America	1
Santa Clara	CA	USA	North America	2
Silicon Valley	CA	USA	North America	2
Thousand Oaks	CA	USA	North America	1
	CA	USA	North America	4
Stamford	CT	USA	North America	1
Atlanta	GA	USA	North America	3
Boise	ID	USA	North America	1
Chicago	IL	USA	North America	1
Boston	MA	USA	North America	1
Burlington	MA	USA	North America	1
Cambridge	MA	USA	North America	1
Okemos	MI	USA	North America	1
Eden Prairie	MN	USA	North America	1
New York	NY	USA	North America	2
Malvern	PA	USA	North America	1
	SC	USA	North America	1
Austin	TX	USA	North America	2
Dallas	TX	USA	North America	1
Bellevue	WA	USA	North America	2
Redmond	WA	USA	North America	4
Seattle	WA	USA	North America	3
		USA	North America	3

Table 21 - Locations of participants of the industry survey questionnaire in North America.

Abbreviation	Chi-Square Value	P-Value	Result with $\alpha = .05$	Result with $\alpha = (.05/24)$
Learn	252.115942	0.00000	Proportions different	Proportions different
Apply	260.9241877	0.00000	Proportions different	Proportions different
Integrate	202.2971014	0.00000	Proportions different	Proportions different
Course	58.45652174	0.00000	Proportions different	Proportions different
Abroad	67.45985401	0.00000	Proportions different	Proportions different
Technology-Aware	371.3406593	0.00000	Proportions different	Proportions different
Technology-Apply	212.03663	0.00000	Proportions different	Proportions different
Definitions	269.1751825	0.00000	Proportions different	Proportions different
Standards	137.9264706	0.00000	Proportions different	Proportions different
Colors	101.6363636	0.00000	Proportions different	Proportions different
Visual	132.2971014	0.00000	Proportions different	Proportions different
Noise	91.80291971	0.00000	Proportions different	Proportions different
Laws	160.4492754	0.00000	Proportions different	Proportions different
Translation-Aware	232.6538462	0.00000	Proportions different	Proportions different
Translation-Apply	127.0387597	0.00000	Proportions different	Proportions different
Direction-Aware	272.7953668	0.00000	Proportions different	Proportions different
Direction-Apply	157.6976744	0.00000	Proportions different	Proportions different
Collation-Aware	229.7307692	0.00000	Proportions different	Proportions different
Collation-Apply	139.6744186	0.00000	Proportions different	Proportions different
Dates-Aware	318.5096525	0.00000	Proportions different	Proportions different
Dates-Apply	158.2007722	0.00000	Proportions different	Proportions different
Currency-Aware	276.5019305	0.00000	Proportions different	Proportions different
Currency-Apply	157.3875969	0.00000	Proportions different	Proportions different
Ethics	143.0769231	0.00000	Proportions different	Proportions different

Table 22 - Results for Test for Equal Proportions

Abbreviation	Strongly Agree Z Value	Overall Agree Z Value	Overall Disagree Z Value	Strongly Disagree Z Value
Learn	-5.177	11.557	-14.687	-15.771
Apply	0.421	11.716	-14.841	-16.042
Integrate	-3.973	10.233	-14.567	-15.891
Course	-11.798	-0.120	-7.584	-14.085
Abroad	-15.466	-9.787	-0.121	-10.149
Technology-Aware	1.755	15.191	-16.039	-16.402
Technology-Apply	-3.813	10.470	-15.070	-16.160
Definitions	0.604	11.962	-15.103	-16.190
Standards	-7.640	6.548	-12.854	-15.644
Colors	-8.744	4.281	-11.638	-15.498
Visual	-7.223	6.501	-12.761	-16.011
Noise	-8.941	2.779	-11.720	-15.828
Laws	-5.658	8.186	-13.844	-16.132
Translation-Aware	-2.481	11.535	-14.760	-15.752
Translation-Apply	-7.720	6.101	-12.078	-15.440
Direction-Aware	1.056	12.117	-14.975	-15.845
Direction-Apply	-5.852	8.093	-13.448	-15.689
Collation-Aware	-0.744	11.163	-14.636	-15.752
Collation-Apply	-6.475	6.848	-13.448	-15.689
Dates-Aware	2.423	13.111	-15.224	-15.845
Dates-Apply	-5.157	8.388	-13.484	-15.596
Currency-Aware	0.932	12.241	-15.472	-15.845
Currency-Apply	-5.479	8.093	-13.821	-15.564
Ethics	-5.706	6.946	-14.140	-15.380

Table 23 - Results of Z Values for Test for Significant Majority

Abbreviation	Strongly Agree P-Value	Overall Agree P-Value	Overall Disagree P-Value	Strongly Disagree P-Value
Learn	1.000	0.000	1.000	1.000
Apply	0.337	0.000	1.000	1.000
Integrate	1.000	0.000	1.000	1.000
Course	1.000	0.548	1.000	1.000
Abroad	1.000	1.000	0.548	1.000
Technology-Aware	0.040	0.000	1.000	1.000
Technology-Apply	1.000	0.000	1.000	1.000
Definitions	0.273	0.000	1.000	1.000
Standards	1.000	0.000	1.000	1.000
Colors	1.000	0.000	1.000	1.000
Visual	1.000	0.000	1.000	1.000
Noise	1.000	0.003	1.000	1.000
Laws	1.000	0.000	1.000	1.000
Translation-Aware	0.993	0.000	1.000	1.000
Translation-Apply	1.000	0.000	1.000	1.000
Direction-Aware	0.145	0.000	1.000	1.000
Direction-Apply	1.000	0.000	1.000	1.000
Collation-Aware	0.772	0.000	1.000	1.000
Collation-Apply	1.000	0.000	1.000	1.000
Dates-Aware	0.008	0.000	1.000	1.000
Dates-Apply	1.000	0.000	1.000	1.000
Currency-Aware	0.176	0.000	1.000	1.000
Currency-Apply	1.000	0.000	1.000	1.000
Ethics	1.000	0.000	1.000	1.000

Table 24 - Results of P-Values for Test for Significant Majority

Years	Count
1	3
2	4
3	8
4	6
5	15
6	5
7	2
8	5

Years	Count
9	6
10	25
11	8
12	12
13	6
14	6
15	22
16	8

Years	Count
17	2
18	10
19	5
20	21
21	3
22	2
23	4
24	5

Years	Count
25	16
26	3
27	2
28	1
29	3
30	9
31	1
32	1

Years	Count
35	1
39	2
40	3
42	1
58	1

Table 25 - Industry responses on number of years working in the industry.

H.3. Titles

Among the respondents that indicated their titles, the following are the data extracted exactly from the database. Titles that are marked with an asterisk (*) were edited for the following reasons:

- Title that could be linked to an organization or to a person
- Errors with spelling or capitalization problems
- Invalid titles such as the abbreviation for “Doctor”, “Mister”, or “Miss”.

• Account Executive	• Development Testing Manger	• Globalization Test Architect
• Administrator*	• Director	• GM
• Applications Engineer, Italian Localization	• Director	• Head of Localization Department
• Architect	• Director	• Head Software Development Unit
• Associate IS	• Director *	• Information Architect
• Bidi Architect	• Director	• Information Development Lead
• Business Developer (Sales and Marketing Department)	• Director - Business Development	• Internationalization Architect
• Business Development/ Project Manager	• Director of Human Resources	• Language Technology Specialist, Business Development Manager
• Campus Program Manager	• Director of Information Technology	• Linguist
• CEO *	• Director of Localization	• Linguist *
• CEO	• Director of Marketing	• Linguist
• CEO	• Director of Recruiting and HR	• Localization Manager *
• CEO	• Director of Sales	• Localization Coordinator
• CEO	• Director of Web Development	• Localization coordinator
• Chief Architect	• Director, External Research Office	• Localization Engineer
• Chief Globalization Architect	• Director, Global Academic & Certification Programs	• Localization Engineer
• Chief Software Architect	• Director, Product Development	• Localization Manager
• Chief Technology Officer	• Division President *	• Localization Manager
• CMS and Language Technologies Manager	• Documentation manager	• Localization Manager
• cofounder	• Economic Advisor *	• Localization Program Manager
• Co-founder & CTO	• Education Manager	• Localization Project Manager
• Communication Manager	• Engineering Manager	• Localization Project Manager
• Computer Applications Developer *	• Engineering VP	• Localization Project Manager and Advisory IT Specialist
• Computer Scientist	• European Localization Director *	• Localization PM *
• Consultant	• Executive Officer *	• Manager
• Consultant	• Founder	• Manager
• COO	• Founder & Chief Architect *	• Manager of Software Development
• Copy Manager	• Founder and Chie Linguist	• Manager Operations
• CTO	• Freelance translator	• Manager, Consumer Technical Publications
• CTO	• General Manager *	• Managing Director
• CTO	• General Manager	• Managing Editor
• CTO	• Global Director*	• Marketing manager
• CTO	• Globalization Educator & Consultant	• MD
• CTO	• Globalization Executive	
• CTO/GM		
• Curriculum Manager		

- MD
- Operations Manager
- Owner
- Owner
- Postgrad Student
- President
- President
- President
- President
- Principal
- Principal development manager *
- Principal Software Engineer
- Principal Technology Architect
- Principal *
- Product Manager
- Product Marketing Manager
- Program director, globalization requirements
- Program Manager
- Program Manager
- Program Manager
- Program Manager
- Programmer
- Project Coordinator
- Project Manager
- Project manager *
- QA Engineer
- QA Manager
- Quality manager
- Research
- Sales Engineer
- Security Analyst
- Senior Business Development Manager *
- Senior Director
- Senior Engineer
- Senior I18n Engineering Manager
- Senior Language Manager
- Senior Language Technology Specialist
- Senior Manager, Campus Recruiting
- Senior Product Manager
- Senior Project Manager
- Senior Research Director
- Senior Software Development Engineer
- Senior Software Engineer
- Senior Terminology Researcher
- Software Development Engineer
- Software Engineer
- Software Engineer
- Software Engineer
- Software localization manager
- Software Localizer
- Sr. Director - Engineering
- Sr. Director Service Delivery
- Sr. Director, Community Marketing
- Sr. International Program Manager
- Sr. Manager of Strategic Alliances
- Sr. Manager, User Experience
- Sr. Product Manager, Globalization
- Sr. Software Engineer
- SVP
- SVP
- SVP Product Development
- Technical Manager
- Technical Writer
- Technical Writing & Localization
- Test Analyst
- Test Lead
- Training and Implementation Specialist
- Translator
- Translator, Localization Professional
- Unicode Software Engineer
- Vice President Technology
- Vice President, Global IT
- Vice President, Localization
- VP
- VP Global Business Development
- VP Marketing
- VP Marketing
- VP marketing
- VP of Engineering
- VP of Engineering
- VP Products
- VP Software Development
- VP, Engineering
- VP, Strategic Accounts

H.4. Participating companies

- 3TERA
- Achievo Corporation
- ADAPT Localization Services
- Adaquest
- ADC
- ADERANT Holdings
- Adobe
- Advanced Micro Devices
- Albanian Translations
- Alchemy
- Algeron
- Alpha CRC
- Alterian
- Altova
- Amgen
- Anchiva Systems
- Appistry
- Asia Online
- ACM
- Atlassian
- Atlas
- Atypon Systems
- Autodesk
- Avaya
- Avocent
- B2B Technologies
- Baliho
- Batutis
- Bayer Healthcare
- Bentley
- CA
- CaridianBCT
- Centrifify
- Chevron
- Cisco
- Clearswift
- CLS Communication AG
- CollabNet
- Continuous Computing
- Corvallis
- Coupa
- Covario
- CreateSpace
- Diebold
- eBay
- eDT
- ELEKS Software
- Embarcadero Technologies
- Envision
- EQHO Communications
- Eurus
- Excel GITS
- Exigen Services
- Expedia
- Ficorp
- Fog Creek Software
- Formtek
- Fortify Software
- FreedomPay
- Funambol
- Gemini Mobile
- Glovia
- Google
- Handango
- HP
- IBM
- IER
- Infosys
- Ingo International Investments
- InMage
- InnerWorkings
- IntelPeer
- Intelliworks
- Intergraph
- INVITEC
- Jonckers
- Jungle Communications
- Kaseya
- Keonnect.com
- Kilgray Translation
- Kx
- Langix
- Lawrence Livermore National
- LISA
- Localisation Research Centre
- Logon
- LUZ
- Mastercard
- MEDIO Systems
- Mentor Graphics
- MGO-Traducciones
- Micro Focus / Borland
- MicroLink
- Microsoft
- NVIDIA
- OpenTable
- Opera
- OSIssoft
- Overhead
- Parasoft
- Paypal
- Playtech Bingames
- Prisma International
- Protegrity
- PTC (Parametric Tech. Corp)
- Qualcomm
- Qumu
- Raytheon
- Really Strategies
- Right Hemisphere
- Rosetta Stone
- RoyaltyShare
- Rubric
- Salesforce
- SAP
- SAS Institute
- Saylent Technologies
- Scania
- SDL
- Siemens
- SignalDemand
- Signiant
- SMOBILE Systems
- SolarWinds
- STAR Group
- SteelEye
- SugarCRM
- Sun
- Symantec
- Synopsys
- TechSmith
- Tektronix
- Telltale Games
- TIBCO Software
- Tivoli Software
- TOIN
- TotalView Technologies
- Trading Technologies
- Translatewiki.net
- Trend Micro
- VersionOne
- Vizrt
- Vocera Communications
- Warelords
- Welocalize
- WHP
- Yahoo!
- YouMail

H.5. Survey results

Abbreviation	Strongly Agree	Somewhat Agree	Neither	Somewhat Disagree	Strongly Disagree	Sample Size
Learn	95	139	26	9	7	276
Apply	142	94	26	10	5	277
Integrate	105	118	36	11	6	276
Course	40	97	64	54	21	276
Abroad	9	47	82	83	53	274
Technology-Aware	151	111	7	3	1	273
Technology-Apply	105	118	38	9	3	273
Definitions	142	94	26	9	3	274
Standards	73	117	52	23	7	272
Colors	65	108	61	32	9	275
Visual	78	114	52	27	5	276
Noise	63	97	74	34	6	274
Laws	91	115	47	19	4	276
Translation-Aware	110	113	26	8	3	260
Translation-Apply	67	111	48	27	5	258
Direction-Aware	138	89	23	7	2	259
Direction-Apply	82	112	43	18	3	258
Collation-Aware	124	96	28	9	3	260
Collation-Apply	77	107	53	18	3	258
Dates-Aware	149	86	17	5	2	259
Dates-Apply	88	109	41	17	4	259
Currency-Aware	137	91	26	3	2	259
Currency-Apply	85	109	46	14	4	258
Ethics	84	102	58	10	6	260

Table 26 - Summary of survey results.

H.6. Graphs of the survey results

Figure 51 - Industry responses to "Learn."

Figure 52 - Industry responses to "Apply."

Figure 53 - Industry responses to "Integrate."

Figure 54 - Industry responses to "Course."

Figure 55 - Industry responses to "Abroad."

Figure 56 - Industry responses to "Technology-Aware."

Figure 57 - Industry responses to "Technology-Apply."

Figure 58 - Industry responses to "Definitions."

Figure 59 - Industry responses to "Standards."

Figure 60 - Industry responses to "Colors."

Figure 61 - Industry responses to "Visual."

Figure 62 - Industry responses to "Noise."

Figure 63 - Industry responses to "Laws."

Figure 64 - Industry responses to "Translation-Aware."

Figure 65 - Industry responses to "Translation-Apply."

Figure 66 - Industry responses to "Direction-Aware."

Figure 67 - Industry responses to "Direction-Apply."

Figure 68 - Industry responses to "Collation-Aware."

Figure 69 - Industry responses to "Collation-Apply."

Figure 70 - Industry responses to "Dates-Aware."

Figure 71 - Industry responses to "Dates-Apply."

Figure 72 - Industry responses to "Currency-Aware."

Figure 73 - Industry responses to "Currency-Apply."

Figure 74 - Industry responses to "Ethics."