Configuration-driven Table Library

by

Torrin Smith

Computer Science Department
College of Engineering
California Polytechnic State University
2015

Date Submitted: June 6, 2015
Advisor: Dr. Hasmik Gharibyan

TABLE OF CONTENTS

Sections

Appendices

ABSTRACT

The goal of this senior project was to write a configuration-driven table library to build tables of data into application websites with minimal work being done by the application and most of the work being done by the library. This senior project was executed with my employer, Amazon SLO. Amazon has an existing table library with many useful features that our team wanted to use in our applications. However, using this library requires a fair amount of integration and configuration work. This project was to create a wrapper around the existing library allowing easy creation of customizable tables in my team's applications. The backend portion of this project, parsing table configuration files and building table view definitions, was written in Java. Table view features, such as table cell types, were defined using JSP. Table front-end/client-side actions were written using Javascript.

NOTIFICATION

This project was completed under a non-disclosure agreement with Amazon, so this report does not include all of the specific details of the implementation of the project.

I. Introduction

My team at Amazon SLO owns, develops, and maintains Amazon's two self-publishing websites, CreateSpace and Kindle Direct Publishing (KDP), where independent authors can self-publish their own print books and Kindle eBooks respectively [1,2]. Recently, my team has also been expanding its horizons by becoming more platform-oriented, building web development platforms that can be easily reusable for other Amazon Kindle content ingestion websites.

My senior project was to create a reusable, and highly configurable table library to build tables of data into applications. Part of the motivation for this project was to create a library that can be reused by a variety of teams at Amazon enabling them to build customizable tables into their websites with minimal work. Specifically, this table library focuses on the features needed to build "dashboard" tables, displaying a user's books (or other forms of user created content) on a site such as KDP. This library can be used to create a variety of tables, however.

II. Requirements

The requirements of this project were to create a configuration-driven library to quickly build tables into applications. The application provides JSON configuration to the library, as well as JSON data, both of which the library parses, performs computations and logic, as necessary based on the data and configuration, returning a view to display the table. The requirements also called for the library to be able to create customized cells defined by the configuration. Another requirement was for the library to be easily extendable, allowing the specific applications (the users of the library) to be able to easily add their own features, which would easily integrate with the base features of the library. The library also had to support intricate customized cell types to support complex cells used in the KDP bookshelf page. These complex cells needed to be able to be defined in configuration, as well as vary based on the data provided for each row of the table. As a proof of concept for the project, the last requirement was to use the library to create a newly designed KDP bookshelf to prove that a complex table can be created by the library, defined by configuration, and that minimal work is needed by the application, most being done by the library.

III. Tools and Teamwork

The server-side code for this project (the majority of the library) was written in Java, using JDK 1.7. The front end code was written in Javascript. The Eclipse IDE was used for development. Amazon internal tools were used for source code revision control.

I worked primarily by myself on this project. For the initial requirements and design of the project I worked with my manager and another senior developer. Also, as I incrementally developed the project, my manager periodically reviewed my code check-ins, suggesting fixes and improvements as needed. For the proof of concept for the library, the redesign of the Kindle Direct Publishing bookshelf page, my manager helped finish some of the last work as the deadline for the project approached. I worked with another employee on the requirements for the backend data index where the data for the KDP bookshelf table was queried from. I also worked with some other senior developers on my team with design choices for the project.

IV. Results

The results of this project were significant, in that I delivered a library meeting the above mentioned requirements, as proven by the redesign on the KDP bookshelf using my library. Although there are some areas for improvement in the library, which will be discussed later, the overall outcome was positive. The successful results of the project will be looked at by addressing the individual requirements for the project.

The first main requirement of the project was for the library to be configuration-driven, reading in and parsing JSON configuration, and using this configuration data to create a table. The library successfully meets this requirement by parsing JSON configuration into internal data structures which are then used by the library to create the table view. The table configuration defines such things as a header for the table, individual column headers, and cell types for each column. These are all used to define the view for the table. The configuration can also define the sorting, filtering, searching, and pagination functionality of the table. The configuration defines which data fields are sortable, filterable, and searchable, and how to display the sort/filter/search inputs to the table (ie dropdowns for sorts and filters, text input for search). The number of entries per page for the table, as well as how and where the pagination buttons are displayed, are defined in the configuration as well. The configuration also defines the data provider to the table (such as the data index for KDP),which is queried by the library through searches, sorts, filters, etc, to get the data used in the table. The library parses all of these configuration values into internal data structures which are then used to build the table in the rest of the library.

The library also allows for customizable cells which are defined by the configuration. There are a variety of default cell "elements" including text, images, links, buttons, dropdowns, popover triggers, etc. The style of each of these elements can be customized through the configuration. A cell type is defined as a list of these elements where the elements for the cell will appear in the order defined. This allows for a large variety of customized cells. It is also simple to add more cell element types to the library, or extend the library in an application by adding element types. The library allows for elements in a cell to be conditional where the data for the row determines whether the element is displayed in the cell. This allows for cells in a column to differ from one row to another based on the data. These features allow for a large range of possible customization of tables using the library.

The library also handles the table actions based on the table configuration. The table created by the library kicks off a refresh action (which is handled by the library) for searches, sorts, filters, and paginations. The library builds a Lucene query for the refresh event that occurred, queries the data provider, and rebuilds the table with the updated data. The library provides a general data provider, which builds a lucene query, queries the backend system for the table data, and returns this data. The backend data system is another field defined in the table configuration.

The successfulness of these library features are illustrated by the redesign of the KDP bookshelf using this library. The new KDP bookshelf uses complex, customized cells, which are defined by the table configuration. All of the refresh actions for the table are now handled by the library, as opposed to being handled by the application as the old version of the bookshelf did. The JSON configuration is defined in the application, and the library does most of the rest of the work in creating the table. This proof of concept was a success in creating a more user friendly experience on the KDP bookshelf, reducing the complexity of the application code for KDP by having majority of the work for the table done by the library, and by creating noticeable improvements in the performance metrics that my team keeps for the KDP bookshelf page.

The old bookshelf and bookshelf redesign are shown in the Appendices. Appendix A shows an overview of the old bookshelf, in comparison to Appendix B showing an overview of the new bookshelf redesign. Appendices C and D show the old and new bookshelfves' row and cell contents, while Appendices E and F do the same for table sorting, searching, and pagination actions.

V. Further Improvements

After successfully completing this project by creating the configuration-driven table library and doing a proof of concept by building a complex, highly used, business-critical table for the KDP bookshelf using this library, there are several areas for further improvement in the library. These include using a library for Lucene queries, supporting JSP and JSON syncs, refactoring the customized cell building, documentation of the library, and some other minor improvements.

The first improvement that could be made to the library is to use a Lucene library for building the queries to the data provider. Currently, the library manually builds queries using string appends. Using a Lucene library would make the query building simpler, more maintainable, and less bug prone. Another improvement that would greatly speed up further development of the library, and applications extending the library, would be supporting JSP and JSON syncs. Currently, when the table element views written in JSP or the table JSON configuration changes, the local server must be rebuilt in order to reflect those changes. A JSP and JSON sync would mean that while the server is running, it automatically syncs the updates to the JSP and JSON so that a server rebuild is not needed. A lot of development time of this project was spent waiting for the server to rebuild to test changes to table configuration JSON or cell element JSP. Supporting syncs would greatly speed up further development of this library.

A general refactor of the customized cell building classes of the library would be another useful improvement. Early on in the iterative development of this project, I made some design decisions that worked when the cells created by the library were simple. As the supported cell types/elements became more complex and the support for cell customization increased, the code grew complex and complicated. In several areas, I forced some “hacky” implementations when I was feeling pressured on time to finish before the project deadline. Therefore, some general refactoring of certain areas of the library in a cleaner way, especially around the customized cell type building, would be a useful improvement to the library. This would also make the library more maintainable, as well as easier to develop in the future.

Documentation of the library is another important improvement. Two sets of documentation should be written for the library: user documentation and developer documentation. User documentation should be written to explain how to use the library features to future consumers of the library. This should include information such as how to define the table configuration and a list of features supported by the library, and how to add these features to a table using the configuration. It should also include information on the work needed to integrate an application with the library to be able to use it to create tables. The developer documentation should include a more in-depth explanation of the inner workings of the library. It should explain how the various features of the library are implemented, and an explanation of the general workflow of the library, from the parsing of the table configuration to creating the table view that is returned to the application for display on the web page.

VI. Conclusion

This senior project has been a great learning experience in many ways. I was able to deliver a successful product to my employer which is in use today for a traffic-heavy and business-critical page on KDP. While working on this project I have gained both technical programming knowledge, software engineering design and best practice knowledge, and more general knowledge- such as communicating progress, requirements, and estimates for a large scale project.

This project involved a lot of programming; I wrote possibly thousands of lines of code for the finished library. Through the large amount of programming done for the project, I gained much technical programming knowledge and have become a better programmer. Specifically, I learned about how to write programs in a general, reusable way. The library has a specific task of building tables of data, but has to implement this in a general way so that it can build arbitrary, customized tables based on the provided configuration. In most programming projects in school, and my previous projects at work, the task involves more of writing programs to have a specific result. However, this project involved a lot more generalization which is an important technical skill to have because the more general and customizable code you write, the more reusable it can be.

This project also involved many planning and design choices. With the pretty general requirements for the project, there were many available options for the implementation of the specific features, as well as the overall structure and design of the library. In this area of the project I worked with my manager and other senior colleagues, and learned important skills regarding design choices for a large scale project from the more experienced developers.

Due to this being a large scale and business-critical project for my team, I also learned more general things about working on a project in the software industry. This includes areas such as estimating, communicating progress, and deadlines. To estimate this project, I had to divide the large project into many different subtasks, evaluating these separately, to combine together for an overall estimate. Dividing this large project into subtasks also helped to improve my ability to communicate progress on the project and to develop incrementally. These are both essential skills to have while working on a large scale project in the software industry. I gained experience in handling deadlines, including how to communicate when a project falls behind schedule. Somewhere in the middle of the project, I came across and had to deal with some tasks that were not originally included in the project subtasks and estimates. This unexpected work resulted in delayed deadlines for the project. More general employment skills such as this are important skills to have as a software industry employee as well.

Overall, this has been a great project, with many important skills learned and an impressive outcome delivered. My work resulted in a completed and functional library, along with the redesigned KDP bookshelf using the library.

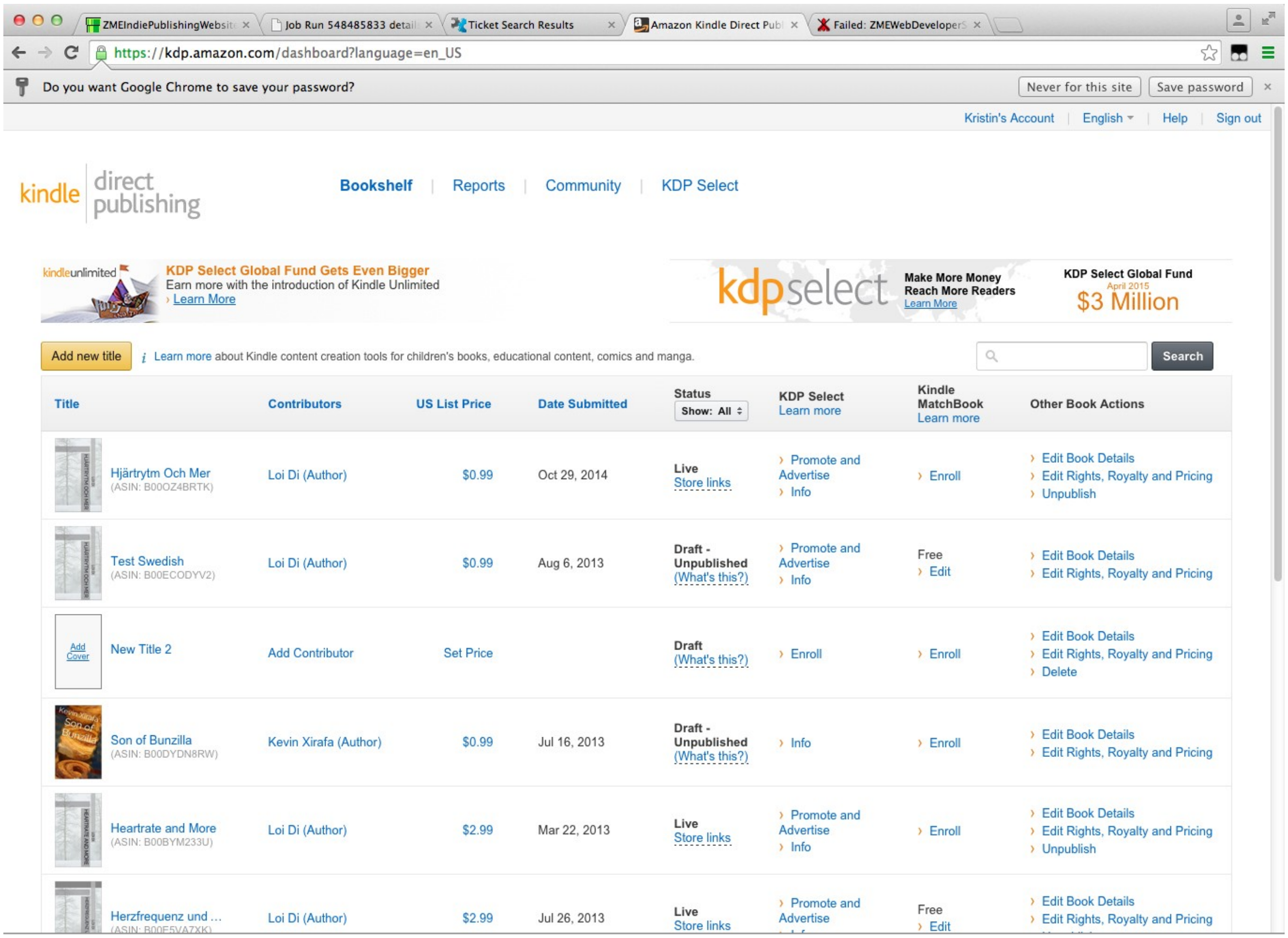
References

[1] https://kdp.amazon.com/help

[2] https://www.createspace.com/Help

APPENDICES

A. Old KDP Bookshelf Overview



Do you want Google Chrome to save your password?

KDP Select Global Fund Gets Even Bigger
Earn more with the introduction of Kindle Unlimited
› Learn More

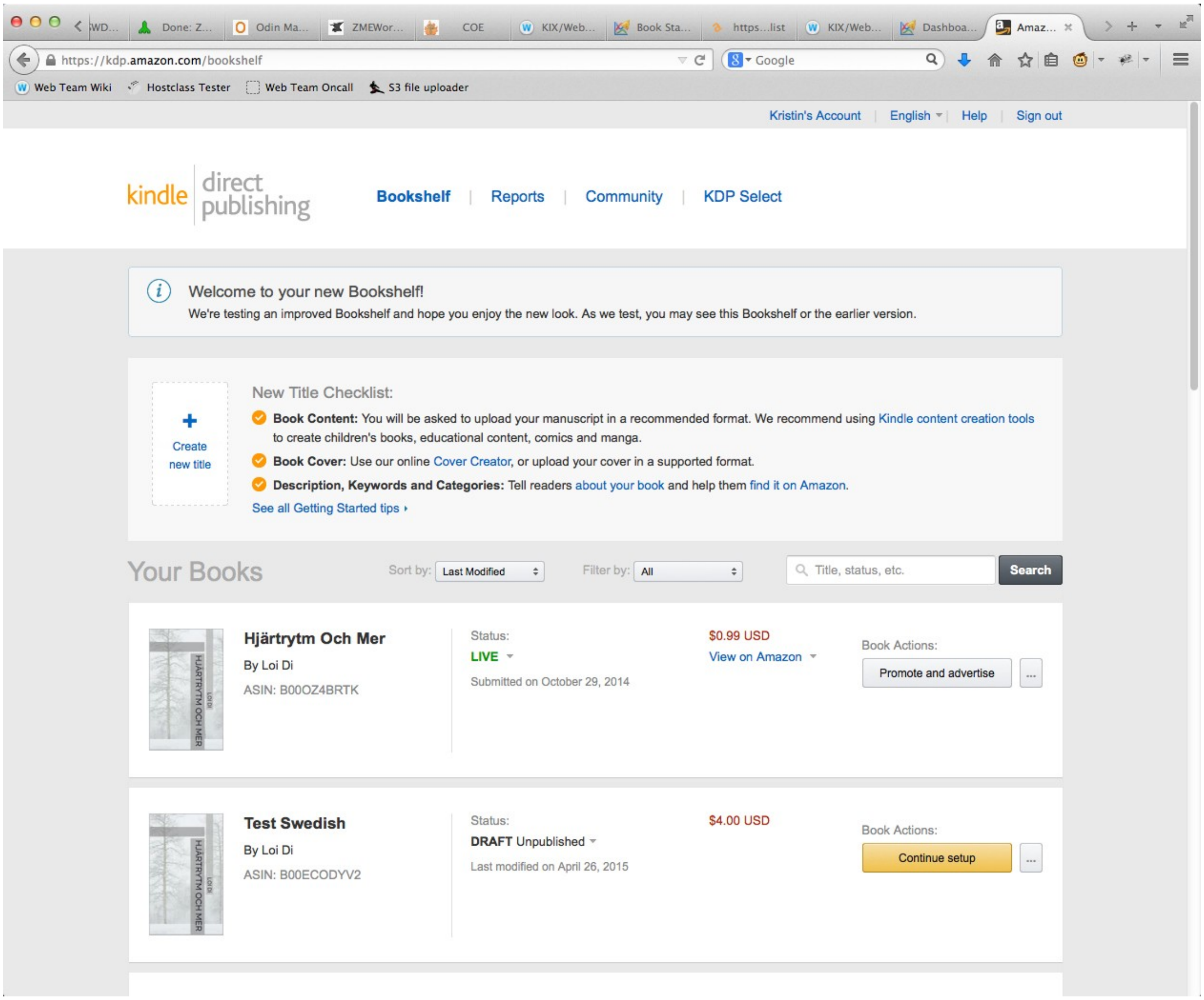
Make More Money
Reach More Readers
Learn More

KDP Select Global Fund
April 2015
$3 Million

i Learn more about Kindle content creation tools for children's books, educational content, comics and manga.

Title	Contributors	US List Price	Date Submitted	Status Show: All	KDP Select Learn more	Kindle MatchBook Learn more	Other Book Actions
Hjärtrytm Och Mer (ASIN: B00OZ4BRTK)	Loi Di (Author)	$0.99	Oct 29, 2014	Live Store links	› Promote and Advertise › Info	› Enroll	› Edit Book Details › Edit Rights, Royalty and Pricing › Unpublish
Test Swedish (ASIN: B00ECODYV2)	Loi Di (Author)	$0.99	Aug 6, 2013	Draft - Unpublished (What's this?)	› Promote and Advertise › Info	Free › Edit	› Edit Book Details › Edit Rights, Royalty and Pricing
Add Cover New Title 2	Add Contributor	Set Price		Draft (What's this?)	› Enroll	› Enroll	› Edit Book Details › Edit Rights, Royalty and Pricing › Delete
Son of Bunzilla (ASIN: B00DYDN8RW)	Kevin Xirafa (Author)	$0.99	Jul 16, 2013	Draft - Unpublished (What's this?)	› Info	› Enroll	› Edit Book Details › Edit Rights, Royalty and Pricing
Heartrate and More (ASIN: B00BYM233U)	Loi Di (Author)	$2.99	Mar 22, 2013	Live Store links	› Promote and Advertise › Info	› Enroll	› Edit Book Details › Edit Rights, Royalty and Pricing › Unpublish
Herzfrequenz und … (ASIN: B00E5VA7XK)	Loi Di (Author)	$2.99	Jul 26, 2013	Live Store links	› Promote and Advertise	Free › Edit	› Edit Book Details › Edit Rights, Royalty and Pricing

B. New KDP Bookshelf Overview



Welcome to your new Bookshelf!

We're testing an improved Bookshelf and hope you enjoy the new look. As we test, you may see this Bookshelf or the earlier version.

New Title Checklist:

- **Book Content:** You will be asked to upload your manuscript in a recommended format. We recommend using Kindle content creation tools to create children's books, educational content, comics and manga.
- **Book Cover:** Use our online Cover Creator, or upload your cover in a supported format.
- **Description, Keywords and Categories:** Tell readers about your book and help them find it on Amazon.

See all Getting Started tips ›

Your Books

Sort by: Last Modified

Filter by: All

Title, status, etc.

Hjärtrytm Och Mer
By Loi Di
ASIN: B00OZ4BRTK

Status:
LIVE
Submitted on October 29, 2014

$0.99 USD
View on Amazon

Book Actions:

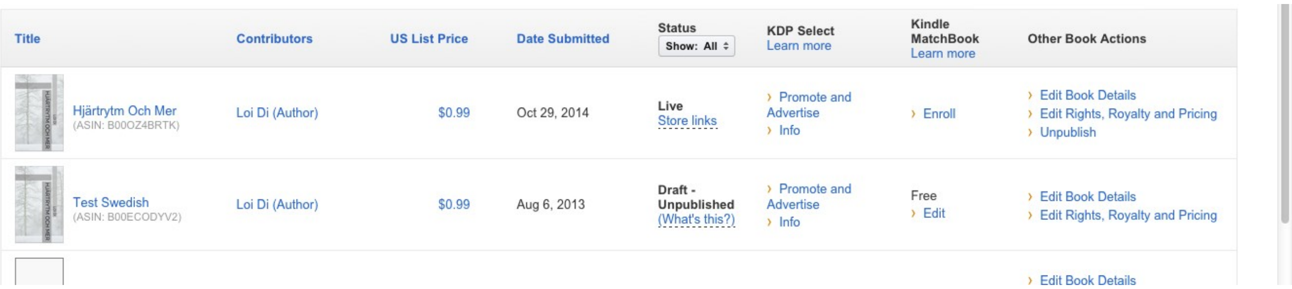
Test Swedish
By Loi Di
ASIN: B00ECODYV2

Status:
DRAFT Unpublished
Last modified on April 26, 2015

$4.00 USD

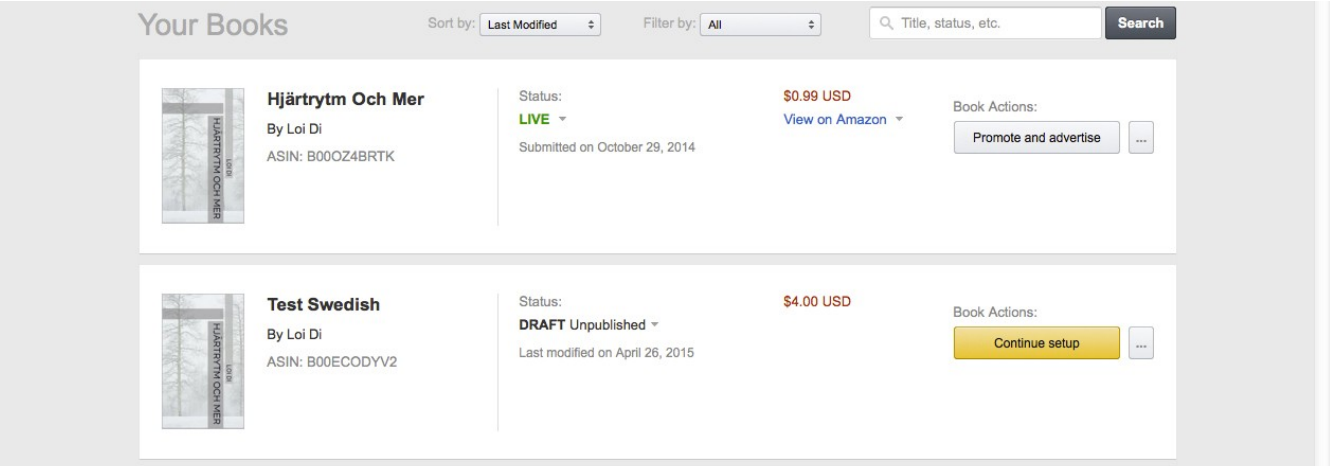
Book Actions:

C. Old KDP Bookshelf Row View



Title	Contributors	US List Price	Date Submitted	Status Show: All	KDP Select Learn more	Kindle MatchBook Learn more	Other Book Actions
Hjärtrytm Och Mer (ASIN: B00OZ4BRTK)	Loi Di (Author)	$0.99	Oct 29, 2014	Live Store links	› Promote and Advertise › Info	› Enroll	› Edit Book Details › Edit Rights, Royalty and Pricing › Unpublish
Test Swedish (ASIN: B00ECODYV2)	Loi Di (Author)	$0.99	Aug 6, 2013	Draft - Unpublished (What's this?)	› Promote and Advertise › Info	Free › Edit	› Edit Book Details › Edit Rights, Royalty and Pricing
							› Edit Book Details

D. New KDP Bookshelf Row View



Your Books

Sort by: Last Modified

Filter by: All

Title, status, etc.

Search

Hjärtrytm Och Mer

By Loi Di

ASIN: B00OZ4BRTK

Status:

LIVE

Submitted on October 29, 2014

$0.99 USD

View on Amazon

Book Actions:

Promote and advertise

...

Test Swedish

By Loi Di

ASIN: B00ECODYV2

Status:

DRAFT Unpublished

Last modified on April 26, 2015

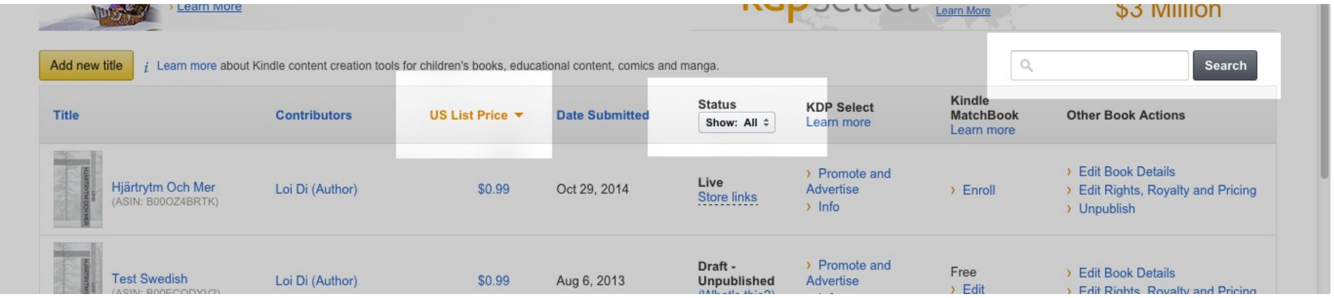
$4.00 USD

Book Actions:

Continue setup

...

E. Old KDP Bookshelf Search/Filter/Sort



› Learn More

Learn More

$3 Million

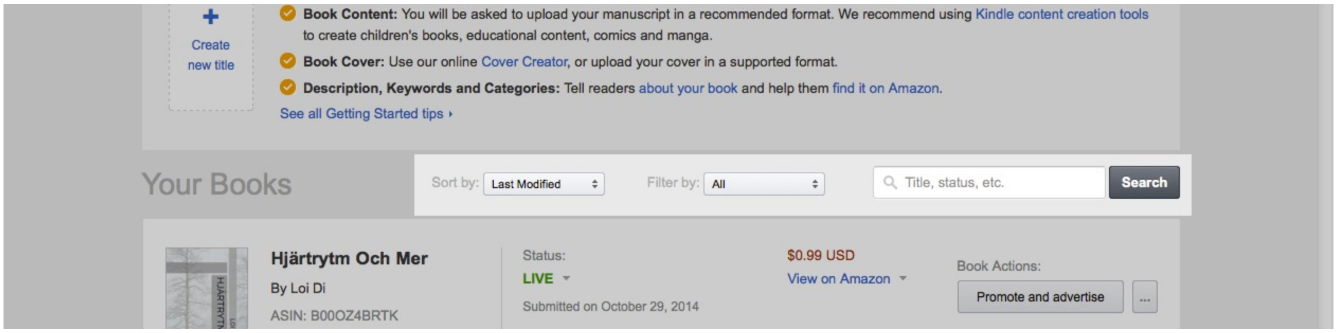
Add new title

i Learn more about Kindle content creation tools for children's books, educational content, comics and manga.

Search

Title	Contributors	US List Price	Date Submitted	Status Show: All	KDP Select Learn more	Kindle MatchBook Learn more	Other Book Actions
Hjärtrytm Och Mer (ASIN: B00OZ4BRTK)	Loi Di (Author)	$0.99	Oct 29, 2014	Live Store links	› Promote and Advertise › Info	› Enroll	› Edit Book Details › Edit Rights, Royalty and Pricing › Unpublish
Test Swedish	Loi Di (Author)	$0.99	Aug 6, 2013	Draft - Unpublished	› Promote and Advertise	Free › Edit	› Edit Book Details › Edit Rights, Royalty and Pricing

F. New KDP Bookshelf Search/Filter/Sort



Create new title

Book Content: You will be asked to upload your manuscript in a recommended format. We recommend using Kindle content creation tools to create children's books, educational content, comics and manga.

Book Cover: Use our online Cover Creator, or upload your cover in a supported format.

Description, Keywords and Categories: Tell readers about your book and help them find it on Amazon.

See all Getting Started tips ›

Your Books

Sort by: Last Modified

Filter by: All

Title, status, etc. Search

Hjärtrytm Och Mer

By Loi Di

ASIN: B00OZ4BRTK

Status:

LIVE

Submitted on October 29, 2014

$0.99 USD

View on Amazon

Book Actions:

Promote and advertise

...