Contents

Overview

A simple overview of the app to support usage and workshops.

The application has multiple versions. Select a version using the admin drop down or admin menu screen. The 'help' menu shows high level changes for each version. This document describes them in more detail. Note this document may have 'spoilers' in terms of deliberate bugs to find etc.

"The Pulper" is a CRUD app (Create, Read, Update, Delete) for Pulp Magazine and book collections.

It comes pre-populated with data.

It should support multiple users, but this depends on what people do with it. Each user is given a unique session and the data is stored on a per user basis, so users should not be able to interfere with each other. This has the risk of consuming more memory, if many people use the application. Inactive sessions are removed after about 10 to 15 minutes of inactivity, and the user will be provided with a new session.

- https://eviltester.com/thepulper
- https://thepulper.herokuapp.com/

- https://github.com/eviltester/TestingApp/releases

Version 001

Version 001 is a read only version of the application.

Released in 2018.

It contains information about pulp Series:

- Doc Savage
- The Avenger
- The Spider

The application supports:

- Home
  - Main Menu with a cut down set of options
  - Help Menu
- Books
  - Table listing of all books with link to the Book entity view page
  - List (ul, li) listing of all books with link to the Book entity view page
- Authors
  - List of all authors, with link to their entity view page.
  - FAQ list, which is a list of all authors with link to their entity view page, and a link to frequently asked questions page
- Publishers
  - List of all Publishers with link to their entity view page

- Series
  - List of all Series with link to their entity view page
- Years
  - List of all Years with link to a list of books published in that year
- Search
  - search by title using form
  - search by title using dialog
  - prompt for "are you sure you want to search?" can be toggled in the search form
  - exit button on page to take to main page
  - help button shows an alert explaining search functionality
- Reports
  - printable list of books as table, books as list, author names, publisher names, series names, and publication years
- View Pages
  - Entity Views
    - Book
      - shows each book details
      - has links to the publisher, year published, etc. view pages
      - link to book as list
    - Author

      - shows each author details
      - has a link to all books by the author
    - Author
      - shows each author details
      - has a link to all books by the author
- Entity FAQ Views (Author)
  - only accessible from the list
  - a list of questions for each entity, which link to a search engine search to answer the question e.g. What books did ‘author’ write?

Version 001 Exercises

- use the app, model it, and perform exploratory testing, take notes as you go
- documentation coverage - does the documentation match the application? anything missing or incorrect?
- help coverage - does the help file in the app match the app?
- take note of the URLs as you use the application
  - change the parameters in the urls, does the application handle ‘invalid’ parameters well?
  - try changing the urls and see if you can find ‘hidden’ or unreleased functionality

Version 002

Version 002 adds the ability to create, amend and delete entities.

Released in 2018.

There is an 'admin' function to experiment with the prototype filtering and paging functionality.

Version 002 Exercises

- explore the CRUD functionality for each entity (Book, Series, Year, Author, Publisher)
  - can we create each entity?
  - can we delete each entity?
  - can we amend each entity?
  - can we read each entity correctly after amendment?
  - is there appropriate validation?
- explore the experimental 'filtering' functionality in the admin menu, which parts could we use in the main application? Which parts need work?

Version 003

Version 003 has automation hooks and responsive GUI features.

Released in 2018.

- exposes the Publishers, Series, Years and Books FAQs as drop down menu list items.
- adds more ids etc. to make the GUI easier to automate:
- hamburger menu used for mobile site access

Version 003 Exercises

- do all the FAQs work well enough?
- do the drop down menus match the main menus?
- responsive and adaptive web design
    - does the hamburger menu work effectively?
    - is the site responsive enough?
- automating
    - are there enough automation hooks?
    - are any parts of the application hard to automate?

Version 004

Version 004 has additional inline list functionality.

Released in 2018.

Inline amend links in the list:

- Publishers
- Series
- Authors
- Books (table and list)

Inline delete links in the list (confirm required):

- Publishers
- Series
- Authors

Version 004 Exercises

- do all delete links have confirmation?
- do all amend links work effectively?

Version 005

Version 005 is a bug fix release for the “Show All X books.” and now this should show the correct number and entity name for the various lists and reports in the application.

Released in 2018.

Version 005 Exercises

- exploratory test reporting and list views and check that the ‘show all x’ works as expected

Version 006

Version 006 is a prototype MVP release.

Released in 2019.

The Create Author screen now uses Ajax XHTTP calls to the backend form processor to create an author.

This should make the application more up to date and dynamic.

Version 006 Exercises

- exploratory test the Create Author screen. This is experimental code. Any issues? Any Risks?
- can you conduct a code review on the JavaScript? Any risks or issues with this?
- is this easier or harder to automate? Any risks with this approach going forward in terms of automating?
- do you need any extra tools to help with this?
  - Browser Dev Tools? Proxy Tools? How might they change your testing?

Version 007

Version 007 is another prototype MVP release.

Released in 2019.

The Create Author screen now uses Ajax XHTTP calls to the api to create an author.

This should make the application even more up to date and dynamic.

Version 007 Exercises

- exploratory test the Create Author screen. This is experimental code. Any issues? Any Risks?
- can you conduct a code review on the JavaScript? Any risks or issues with this?
- is this easier or harder to automate? Any risks with this approach going forward in terms of automating?
- do you need any extra tools to help with this?
  - Browser Dev Tools? Proxy Tools? How might they change your testing?
- what risks does using the API vs Form submission change?

Version 008

We added a search API end point into the API.

Released in 2020.

The search form now uses Ajax calls to this end point.

This also uses cookies and local storage to track previous searches.

Version 008 Exercises

- exploratory test the Search screen. This is experimental code. Any issues? Any Risks?
- can you conduct a code review on the JavaScript? Any risks or issues with this?
- is this easier or harder to automate? Any risks with this approach going forward in terms of automating?
- do you need any extra tools to help with this?
  - Browser Dev Tools? Proxy Tools?
  - API GUI Tools?
  - How might they change your testing?

Version 009

We extended use of the API from the GUI to all the Create, Update and Delete functionality for all the entities.

Released in 2020.

Version 009 Exercises

- exploratory test the Search screen. This is experimental code. Any issues? Any Risks?
- can you conduct a code review on the JavaScript? Any risks or issues with this?
- is this easier or harder to automate? Any risks with this approach going forward in terms of automating?
- Can you test at the API level?
- do you need any extra tools to help with this?
  - Browser Dev Tools? Proxy Tools?
  - API GUI Tools?
  - How might they change your testing?

Version 010

Released in 2020.

We added bug fixes for the extended use of the API from the GUI.

Primarily on Book, Series and deletion from list.

- All deletions should be confirmed
- All updates are via POST calls
- All deletes are via DELETE calls

Version 010 Exercises

- exploratory test the Book, Series and list screens. Any issues? Any Risks?
- can you conduct a code review on the JavaScript? Any risks or issues with this?
- is this easier or harder to automate? Any risks with this approach going forward in terms of automating?
- Can you test at the API level?
- do you need any extra tools to help with this?
  - Browser Dev Tools? Proxy Tools?
  - API GUI Tools?
  - How might they change your testing?