



Usability Testing of a Single page Application using React

1. Introduction

This project involved looking at single page applications and usability testing. It will look at what they are, and what the two involve.

2. Objectives

- Investigate usability methods and choose a suitable method to test the usability of an existing website
- To look at Single Page Applications (SPA), and frameworks, and what they are.
- To create a single page application in React.js using the results from the usability test to improve its UI design and usability

3. Research

Single page applications are websites that are based on a single website and do not need to refresh the page when a new page is clicked on. Each page is ready to be displayed onclick because of the use of the JavaScript framework React.js.

Usability testing is used to find problems that exist in a website. Usability is measured in ease of use and user experience. The System usability scale method will be looked at and how it works.

4. System usability scale (SUS)

A SUS test is a questionnaire in which users are asked to complete a questionnaire with standard SUS questions, then answering using a Likert scale (1-5).

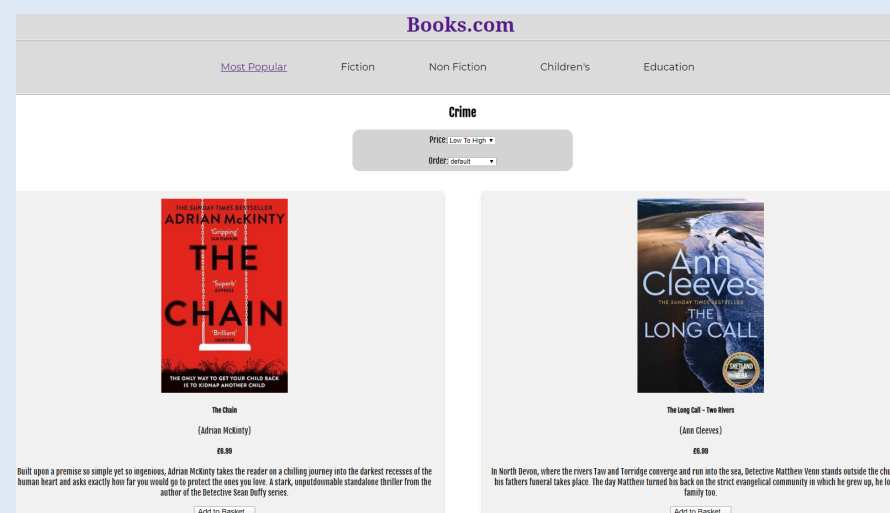
Users were asked to complete 5 tasks on the website wordery.com, which were finding books and information. Then asked to complete a SUS questionnaire and a 4 question questionnaire. The results from the SUS tests completed show an average score of 57 which can be improved using different techniques.

5. Design and implementation

The design of the application was based on the requirements created using the results from the SUS test and User stories created on the main features brought up from the test.

The implementation was done using a SQL server database which stored the Book data. This data was sent to the application using a node rest API. The application itself was created using React, the application used React router to create links that made the application a SPA.

```
//Get request for books with selected category
app.get('/api/books/:Category', function (req, res) {
  var query =
    "select * from BookShop WHERE Category = '" + req.params.Category + "'";
  executeQuery(res, query);
});
```



```
<Router>
  <Header />
  <Nav />
  <Switch>
    /* Routes for all pages needing links */
    <Route path="/" exact component={Home} />
    <Route path="/Basket" component={ShoppingBasket} />
    <Route path="/info" component={InfoPage} />
    <Route path="/:Category" component={BooksDisplay} />
  </Switch>
  <Footer />
</Router>
```

6. Cognitive walkthrough

A cognitive walkthrough was used to look at the usability of the application created. It was to see if the users would achieve the correct outcome and to see if the action performed, showed progress of completing.

The results showed the users would achieve correct outcome because of well laid out categories, the user would notice the correct option, and the user would know if an action was being completed because it would load quickly because of the SPA. Overall the cognitive walkthrough was a success.

7. Evaluation

Data collected in the SUS test and cognitive walk-through back up information researched such as SPA's make websites quicker and more efficient, also that usability testing are useful for finding out how good a feature is in terms of usability. Also React is very good for enhancing interfaces and good for creating single page applications.

8. Conclusion

The objectives that were set, were met by creating a SPA using React which helped enhance the interface and the usability was helped and improved by the SUS test that took place, the UI design and usability of the application created was then evaluated using a cognitive walkthrough and seen if the usability had been improved. It has also been proven that usability testing is a vital part of application development.