****

**Yarn Swap**

Interim Report

Name: Fiona Waters

Student No: 20095357

Programme: Higher Diploma in Computer Science

Module: Project

Project Supervisor: Richard Lacey

# 

[**Abbreviations/Figures**](#_8md8861q9iqt) **2**

[**1 Introduction**](#_dx4mlhsu2p9e) **3**

[1.1 Background](#_pyyyaz7xhj7y) 3

[1.2 Problem Definition](#_egol3jx76zma) 3

[1.3 Proposed Solution](#_7vg5e9xw6fz3) 3

[1.4 Project Goal](#_ibakz4tymo6q) 3

[**2 Research and Analysis**](#_r4wudjetcqr1) **5**

[2.1 Requirements analysis](#_yhfxum2xf7fd) 5

[2.3 Market analysis](#_ki3t2qgd1x9d) 5

[2.2 Feature analysis](#_xk147m9oqkg) 6

[2.2.1 Feature Comparison - Other Swap Web Applications](#_hfi70z2gkrg4) 6

[**3 Modelling**](#_ykczmh71wwb) **6**

[**4 Planning**](#_7dlfqdfb7csy) **6**

# 

# 1 Abbreviations/Figures

# 2 Introduction

## 2.1 Background

Not long after my first child was born, I started to crochet, then knit, and soon became obsessed with yarn and colour. Because of this I have a lot of yarn, I wish I could use it all but unfortunately I have acquired too much yarn, this is sometimes known as **S**tash **A**cquired **B**eyond **L**ife **E**xpectancy (SABLE).

## 2.2 Problem Definition

As a crafter, I have spent a lot of money on yarn over the past number of years. At the time of purchase I almost always have a plan, an idea of what I intend to use the yarn for. Just recently I crocheted a shawl with yarn purchased for that purpose in 2015, 8 years ago. But more often than not it doesn’t work out this way, and yarn is left untouched in my stash and may never be used. As well as this, it is almost certain that I will buy more yarn with the promise of new projects, and this goes on, and on. The thrill of a new project and some lovely new yarn outweighs the short sadness felt for the yarn left unused. You can see how this could be considered a problem.

## 2.3 Proposed Solution

What I have described above is not unusual, most crafters, like myself, will have yarn that they know they will never use. What I want to create is a place where crafters can list and view yarns and have the ability to swap or procure yarns from others that could otherwise be forgotten or wasted.

This is akin to a clothing swap, or a book swap, allowing people to reuse and make use of items already available rather than providing the need for more to be created or produced. This could play a small part in a circular economy.

## 2.4 Project Goal

The goal of this project is to create a progressive web application and CI/CD pipeline.

Web application

The web application will provide a place for crafters to create an inventory of their yarn, with descriptive listings and images. This, in the first instance, allows them to realise or remember what yarn is in their possession. They can then mark items as available to swap, and these yarns will be available for others to view. This is the main function of the proposed web application. A user can earn tokens for yarn they have sent to others, and can use these tokens in order to acquire yarn from other users.

I will use React and Chakra UI for the frontend, and Firebase Authentication and Firebase Realtime Database. This work will draw on learning from technologies covered in previous modules.

I have chosen to use some technologies that I have been learning and using during my work placement, therefore the backend web application will be written in Golang and will use test driven development with automated testing and a CI/CD Pipeline using Jenkins, an open source automation server.

CI/CD Pipeline

The pipeline will allow for automated building, testing and deployment of the application to OpenShift.

# 3 Research and Analysis

## 3.1 Requirements analysis

When deciding what tools and technologies to use for this project I felt very strongly about using those which I am learning at my work placement where possible. This would allow me to upskill and progress. The technologies I have chosen include a mix of those used in the company where I am undertakng my work placement and those I have learned in other modules on this course.

### 3.1.1 Figure 1 - Technologies

| **Front End** | **Back End** | **Pipeline** |
| --- | --- | --- |
| React | Golang | Jenkins |
| ChakraUI | Gin Web Framework | Docker |
|  | Firebase(Auth + Database) | Kubernetes |
|  |  | OpenShift |

### 3.1.2 Front End

**React**

Choosing React probably needs no explanation. It is one of the most widely used technologies for front end development and while there are many other options I would like to further my learning in this area and feel that it is the right choice for this application.

**Chakra UI**

There are so many component libraries that can be used with React. Some are straightforward and quite accessible. Having used other libraries including Material UI, and Semantic and Fomantic UI during the course I felt confident to choose another option and decided on Chakra UI. Chakra UI has a number of components that will work well for the application, and they are all well documented.

### 3.1.3 Back End

**Golang**

At my work placement, my team uses Golang. I have found it very interesting so far and would really like to continue to learn and gain more experience in using Go. I feel that using it for Yarn Swap will allow me to upskill in the language in new ways, while also learning more about its basic syntax.

**Gin Web Framework**

While researching the use of Golang to create backend API’s I came across the Gin Web Framework, and a related tutorial that would help with my learning in this area. This framework allows you to create a restful api using Golang.

**Firebase**

Firebase offer many products that are useful to software developers. The products I intend to use for this project are Firebase Authentication and Firebase Realtime Database. Firebase authentication will allow users to be authentication when they sign up and log in to the application. Firebase Realtime Database will allow the relevant application data to be stored and linked to it’s user.

I have used these on previous projects and find the Firebase documentation to be comprehensive and easy to follow and the resulting actions and processes to work smoothly and efficiently.

\*Project Data\*

All data used in, and stored for, this project will be test/dummy data created by me. No real data will be used.

### 3.1.4 Pipeline

The technologies used for the CI/CD Pipeline are:

**Jenkins** is an automation server with a user friendly UI, that will allow me to build a CI/CD pipeline to the required specifications.

**Docker** allows an application to be contained, along with all of its dependencies so that it can be deployed anywhere.

**Kubernetes** is the captain of the ship - it manages the containerised application.

**OpenShift** is built on Kubernetes and allows for added benefits including increased security and a user friendly UI.

I will use and explore these technologies while creating the CI/CD pipeline for Yarn Swap. These will be discussed more in the implementation section.

At my work placement these technologies are used for many pipelines that allow continuous integration and continuous development and deployment of the product that my team work on. I would very much like to further my learning and experience in this area so choose these technologies for this reason.

## 3.2 Market analysis

Following research I was unable to find another web application or site that offers the swapping or exchange of yarn. The closest I found was a site called Woolswap (<https://woolswap.com.au/>) which arranges swapping of yarn parcels from yarn lovers around the world. In this case those that sign up do not choose the yarn, but instead get a surprise parcel from their swap partner.

I did find sites with a similar purpose to mine but in other areas; fashion and literature. Below I will discuss some of these that I was able to compare my idea with.

**Nuw**

<https://www.thenuwardrobe.com/>

Nuw is a mobile application that allows users to swap items of clothing and other fashion items. I found this app to be inspiring and love how it looks, and its ethos. I have signed up but haven’t yet used it but it seems clear, simple and quite useful, attributes I believe to be quite important for an application.

The app allows users to sign up and earn a coin for each item they upload, they then use these coins to request items from other users and pay a small fee to the site on top of this for each swap.

**Bookswap**

[www.bookswap.co.uk](http://www.bookswap.co.uk)

Bookswap is a web application that allows users to swap books. The users earn points by offering books and can use these to acquire books that others are offering. If you do not have enough points you can purchase some for a small fee.

**Github**

I also conducted a search on Github for any open source projects that may be offering the same functionality but I did not find any relating to yarn. There were a number of book swap repositories written in a number of different languages including python and PHP, as well as a swap for college textbooks written in javascript. There were also some clothing swap repositories which mostly focused on connecting people living in the same area in order to swap clothes.

The swap model is quite versatile and could be utilised for any item. While this project is focusing on yarn, the application could be altered for any item that can be reused or swapped.

## 3.3 Feature analysis

Figure 1 below shows the comparison of features on the applications discussed in the market analysis section 2.2 above and the features proposed for Yarn Swap.

### 3.3.1 Figure 2 - Feature Comparison - Other Swap Web Applications

|  | **Upload Items** | **Browse Listings** | **Earn points** | **User ratings** | **Wishlist** | **Member chat** |
| --- | --- | --- | --- | --- | --- | --- |
| Bookswap | ✔ | ✔ | ✔ | ✔ | ✔ | x |
| Nuw | ✔ | ✔ | ✔ | x | ✔ | ✔ |
| Yarn Swap  Proposed features | ✔ | ✔ | ✔ | ✔ | ✔ | Forum |

As you can see, the applications reviewed have a lot of the same features including the uploading of items, the ability to browse listings, and to earn points/coins/tokens. Some allowed for users to be rated and some allowed for members to chat. I propose the following features for the Yarn Swap - Progressive Web Application:

* Upload items - a user will upload yarn listings.
* Browse listings - a user can browse listings.
* Earn points - a user will earn points/tokens when they add a listing and mark it as swappable, they will then use these to request yarn from other users.
* User ratings - after receiving an item from another user, you will be able to rate the transaction/user.
* Wishlist - a user will be able to create a wishlist, which will contains items they hope to obtain.
* Forum - a user will be able to start a topic and ask a question or for advice on the forum. A user will also be able to contribute to other users topics.

# 

# 4 Modelling

## 4.1 Data Modelling

Scenario:

Yarn Swap needs a data model which will contain user details, listing details, forum details, swap details, admin user details and record the swap outcomes.

* Each user has a unique userId, a username, a first name, a last name, an email address, a password, a profile picture, coins, number of swaps completed and a role.
* Some users are admin users - in this case we want to record the usersId and their role.
* Each listing has a unique listingid, a userid, a username, images, a brand, a colourway, a weight, a fibre content, a meterage(m), a unitWeight in grams, a dyelot, a yarn destiny, a wishlist marker.
* Each forum topic has a topic name, a heading and posts.
* Each swap has a unique swapId, swap status, swapper userId, swappee userId, swapper userName, swappee userName, swapper email, swappee email, listingId. We also want to record the outcome of the swap.
* Each user can create many listings, a listing can be created by one user.
* A user can create many forum topics, forum topics can be created by 1 user.
* A user can participate in many swaps, a swap must have 2 participants(users).
* An admin user can review many listings, a listing can be reviewed by 1 admin user.
* A user can participate in many swaps, a swap must have 2 participants(users).

### 4.1.1 Figure 3 - ER Diagram showing attributes and relationships.

# 

# 

# 5 Planning