**Software Requirements**

**Specification**

**For**

**Knit-A-Bit**

**Version 1.0 approved**

**Prepared by Shannen Bravo-Brown,**

**Courtney Boortz, Jacqueline King,**

**Christopher Draper, and Rebecca Wilson**

**CEN4020 Team Rowers**

**4/24/2017**

[**1. Introduction**](#_na5rzjwolwr9) **3**

[**1.1 Purpose**](#_hfq51zxtbnpm) **3**

[**1.2 Scope of the System Specified**](#_nijulml556uq) **3**

[**1.3 Definitions, Acronyms, and Abbreviations**](#_hojf1z7x2a14) **3**

[**1.4 References to Supporting Documents**](#_a8fmdoavihmq) **6**

[**1.5 Overview of rest of SRS**](#_1p0f3z1h4pxp) **6**

[**2. General Description**](#_na5rzjwolwr9) **7**

[**2.1 Produce perspective**](#_zf7ki18ldtd2) **7**

[**2.2 Product Functions**](#_ebu8jne5yexa) **7**

[**2.3 User Characteristics**](#_5zph7e1hh01s) **8**

[**2.4 General Constraints**](#_se2lr7qr0eo7) **9**

[**2.5 Assumptions and Dependencies**](#_j66zjwlnlmwf) **9**

[**3. Specific Requirements**](#_5vsi9xf4k4wx) **10**

[**3.1 User Interfaces**](#_slniy9qatwk9) **10**

[**3.2 Hardware Interfaces**](#_bs5ctxc908ie) **10**

[**3.3 Software Interfaces**](#_x3litcm1afvm) **12**

**3.4**  [**Communications Interfaces**](#_zasqfzk9kd3r) **13**

[**4. Non-functional Requirements**](#_na5rzjwolwr9) **14**

[**4.1 Performance**](#_nzkr76v1hhid) **14**

[**4.2 Interface**](#_oqxorwg7496o) **14**

[**5. Diagrams**](#_wqyphwm71o6b) **16**

[**5.1 Use case Diagram**](#_tjqkp0le2czw) **16**

[**5.2 UML Class Diagram**](#_tziros5lo1ga) **17**

# 1. Introduction

## 1.1 Purpose

The purpose of this SRS is to describe and document the creation of the mobile application ‘Knit-A-Bit’. To do this, it will provide a detailed walkthrough of the features of the system, the interfaces for the system, the functions of the system, and the environment the system will work in. This document is intended for the Knit-A-Bit development team to keep a record of the functionalities of the application. It will also act as a guide for stakeholders, namely Dr. Gaitros, to assess the application’s structure and design.

## 1.2 Scope of the System Specified

The Knit-A-Bit application has been developed with the intent of facilitating the production of knitted projects from user created patterns. The product will allow the user to create and store knitting patterns, and then to start active projects from those patterns. The knitter will be able to select an active project to become the current project. They will then be able to access this project on the working page where they can increment counters and view a row-by-row guide of their project. The user will be able to begin multiple unique projects from the same pattern. The application will not teach a user how to knit or allow them to export their patterns from the application. All patterns and projects will be stored on the device for later access.

In creating this application, the development team's goal is to provide an easy-to-use resource for knitters that allows them to consolidate their own knitting patterns and projects onto an android device of their choosing. By doing this, a knitter’s productivity and organization will increase as they no longer need to keep track of multiple pdf or printed patterns and their associated projects. In addition, the row-by-row guide of Knit-A-Bit reduces confusion as knitters can leave projects for long periods of time and know exactly where they left off. Overall, the objective of the Knit-A-Bit application is to make knitting easier for its users.

## 1.3 Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| Android Studios | A development environment geared towards creating and publishing apps for android devices. It allows users to both drag and drop in widgets and write XML code to implement them. The user can then implement the functionalities of the widgets by using Java. |
| Android | A mobile operating system that was created to be run on mobile technology devices such as phones and tablets. |
| Pattern | A schematic of stitches and associated information that, when followed, will produce a knitted object. |
| Project | A knitted object that a knitter is creating based on a pattern stored through KAB. A project can be marked as current, active, or completed. |
| Knitter | A Knit-A-Bit user. |
| KAB | Knit-A-Bit |
| Current | When a project is marked as current, indicated by a little star on its project menu page, it will appear on the working page. There can only be one current project at a time. |
| Active | When a project is not current or completed, it is active. This means that it can become current and be edited. |
| Completed | When a project has reached its pattern’s totalRows, it will be marked as completed, indicated by a check mark on its project menu page, and can no longer become current or be edited. |
| Row-by-row | A method to display the instructions of a pattern, whereby only a single row’s instruction is shown at a time. When the knitter increments a counter, the instruction for the next row of the pattern will be displayed. |
| Row | A single line of stitches made by a knitter. In a straight knitting project there are right and wrong side rows corresponding to whether the row number is odd or even. In a circular project there is no distinction. |
| Stitch | The result of a single movement in knitting which creates a loop of yarn. There are two primary stitches: the purl stitch (p) and the knit stitch (k). |
| Section | A collection of rows that create a mini-pattern which may be repeated throughout a larger pattern. |
| Working Page | Also called the current project page, the page that displays the row-by-row instructions, as well as any project counters, of the project marked as current. |

## 1.4 References to Supporting Documents

Any included diagrams created for the KAB app will be placed in section five for ease of reading and document organization, outside references to information related to the project and knitting will be listed out below.

Craft Yarn Council: The linked page is for understanding how to read a knitting pattern. We used this kind of language to accept, store, and display user inputted stitches.

Leinhauser, Jean. "How to Read a Knitting Pattern." How to Read a Knitting Pattern | Welcome to the Craft Yarn Council. Craft Arn Council, n.d. Web. 15 Apr. 2017. <<http://www.craftyarncouncil.com/tip_knit.html>>.

Vogue Knitting: The linked page is a chart displaying the standard yarn weight system. This relates to how the gauge is calculated and the different types of user inputted yarn weight that are accepted.

"Standard Yarn Weight System." Vogue Knitting. Vogue Knitting, n.d. Web. 15 Apr. 2017. <<http://www.vogueknitting.com/pattern_help/how-to/techniques_abbreviations/standard_yarn_weight_system>>.

New Stitch a Day: The linked youtube channel has tutorials ranging from the most basic to advanced knitting patterns and techniques. This was used to help the members of the KAB team who did not know much about knitting learn enough to properly work on this project.

Newstitchaday. "New Stitch a Day: Knitting and Crochet Video Tutorials." YouTube. YouTube, n.d. Web. 15 Apr. 2017. <<https://www.youtube.com/user/newstitchaday>>.

## 1.5 Overview

The rest of this SRS will go over the general description of the KAB project including summaries of: the products relation to the environment, functionality, classes made, constraints on the project, and the assumptions and dependencies made. The SRS will then continue to list and describe the functional and nonfunctional requirements of KAB.

# 2. General Description

## 2.1 Produce perspective

The KAB app will exist as an Android application. The app will be downloaded and then run on an android device for the Nougat and Marshmallow versions of the Android operating system. Once the app has been downloaded it will be a totally stand alone piece of software. The app will not be part of a larger system as the app contains all functionalities and data that it needs.

## 2.2 Product Functions

Through KAB, the knitter will be able to create, edit, and save projects and patterns. They can input new patterns using the template feature, where they create different sections. These sections can then be put in order by the user to create the greater pattern.

Once a pattern is created, it can be viewed on the pattern library page in a list-view. A pattern’s entry in the library will display all of its relevant member data. The knitter can then select a pattern to create a project from which will move them to the project creation page. There the knitter will be prompted for additional information related to the project, like the exact type of yarn they used and its color.

After a project is created, it can be viewed on the project library page in a list-view. A project’s entry in the library will display all of its relevant member data and an indicator of whether it is current, active, or completed. From this page, a knitter can designate the current project from any of the active projects. It will then appear on the working page of the app, where a row-by-row guide through the project and a number of counters are displayed. The row-by-row instructions will update as the knitter increments the primary counter, which indicates the total number of rows completed.

## 2.3 User Characteristics

It will be assumed that the user knows how to read and write english to the 6th grade level. It will be assumed that the user has a basic proficiency in knitting and that they know all of the stitches and abbreviations in the patterns they input. In addition it is assumed that the knitter knows how to cast on and bind off in whatever way their pattern requires.

## 2.4 General Constraints

The application will be constrained by the hardware of the android device that downloads it. The application is lightweight enough that speed and functionality should not be affected by the phone it is downloaded on. The amount of patterns that can be stored will be limited by the amount of memory in the android device. The interface of the application will be limited to the touch screen of the phone and the on screen keyboard of the phone. Any and all user inputs must come from one of these two interfaces.

The application is being designed in Android Studio, so the functionality of KAB will be constrained by what Android Studio allows.

## 2.5 Assumptions and Dependencies

It will be assumed that the user will be using an android device that is running a current version of the android operating system that can support the applications created by the current version of android studio. It is also assumed that the user’s phone has a functioning touch screen and functioning on-screen keyboard. The user must also have a basic understanding of mobile applications, as well as be able to use said applications.

We assume that the user is familiar with the basics of knitting.

Because the patterns are saved locally on the mobile device, the number of patterns and projects that can be saved through the KAB application depends on the size of the device’s available storage.

# 

# 

# 3. Specific Requirements

This section contains all of the functional and quality requirements of the system. It gives a detailed description of the system and all it’s features.

## 3.1 User Interfaces

#### 3.1.1 Ease of Use

**ID: 1A**

The interface shall be easy for users to read and navigate. Text will be easily distinguishable against the background and navigation buttons will be clear to the user.

#### 3.1.2 Grabbing Input

**ID: 1B**

The user shall be able to input information about projects and patterns through the on screen keyboard being inputted into text fields or a drop down menu.

#### 3.1.3 Page Flow

**ID: 1C**

The application shall start the user off on a home screen. From the home screen the user shall be able to go to the projects library, the patterns library, or their current project. From the libraries the user shall be able to see all their projects or patterns listed out in some format.

## 3.2 Hardware Interfaces

#### 3.2.1 Authorizations Needed

**ID: 1A**

The application shall only require authorization to store and retrieve data from the phone.

#### 3.2.2

**ID: 1B**

Saved instances of the application shall be stored on the mobile device.

## 3.3 Software Interfaces

#### 3.3.1 Creation of Project

**ID: 3A**

The application shall allow the user to create a project using the template page of the application. A project requires at least one pattern, a name, a size, yarn weight, needle size, and number of colors.

#### 3.3.2 Creation of Pattern

**ID: 3B**

The application shall allow the user to create a knitting pattern. A knitting pattern requires a name, a number of stitches, if the pattern is for a straight or curved fabric, and the series of stitches required to make that pattern. The user shall also be allowed to set section by section the series of rows needed to create a pattern. The user shall be allowed to set row by row the series of stitches needed to create a section.

#### 3.3.3 Saving Data

**ID: 3C**

The application shall store information about the projects and patterns so that those projects and patterns can be accessed at a later date. No data related to projects and patterns shall be lost when the app is closed or opened. The data shall be stored on the phone and not elsewhere.

#### 3.3.4 Grabbing User Input

**ID: 3D**

The application shall take user input and use that input to create patterns and classes. The application shall also take user input to navigate in between the pages of the app.

#### 3.3.5 Editing

**ID: 3E**

The application shall allow the user to edit both patterns and projects. The user shall be able to edit any part of the pattern or project. The edits the user names shall be saved to memory. The user shall not be allowed to edit completed projects. Editing patterns shall not affect projects that user that pattern.

#### 3.3.6 Deleting

**ID: 3F**

The application shall allow the user to delete any pattern or project. If a pattern is deleted projects that use that pattern shall not be affected. If a project is deleted patterns used by that project shall not be affected.

#### 3.3.7 Current Project

**ID: 3G**

Only one project shall be designated as the current project. The current project shall be accessible from the homescreen. The current project shall be the project displayed on the working page. The current project shall be the project the row by row instructions uses. The current project shall also display the row number that the knitter is currently on out of the total rows in the project. This information shall be saved to the phone so that the knitter can continue working on the project at anytime.

#### 3.3.7 Row by Row Counter

**ID: 3H**

The working page of the application shall have a row by row counter. The row by row counter shall keep track of which row of stitches the user is currently on out of the total number of rows in the project. The application shall show instructions on how to knit the current row. The user shall be able to increment this counter. Once the counter has reached it’s max value the project shall be marked as completed.

#### 3.3.9 Libraries

**ID: 3I**

The application shall allow the user to view a library of their created patterns and projects. There shall be one library for patterns and one library for projects. The pattern library shall show the user relevant data for each pattern. From the pattern library the user shall be able to edit, delete, or create a pattern. The project library shall show the user relevant data for each project and indicate if the project is active, complete, or current. The project library shall allow the user to edit the current and active projects, delete active and completed projects, and set active projects to be the current project.

## 3.4 Communications Interfaces

#### 3.4.1 Data Storage

**ID: 4A**

The mobile application connects with the phone device in order to store data onto the phone in the shared preferences file. The objects saved through the application will be converted, through JSON, to GSON strings that can be easily and quickly retrieved.

# 

# 

# 

# 4. Non-functional Requirements

## 4.1 Performance

### 4.1.1 Android Device Restrictions

**ID: 41A**

The app should remain lightweight enough that it will be able to run on any fully functioning android device made within the last three years.

### 4.1.2 Storing projects and patterns

**ID: 41B**

The app should store the data it needs relating to patterns and projects in a efficient and effective way.

### 4.1.3 Pattern Storage Limits

**ID: 41C**

The app should allow the user to hold as many patterns as their phone can handle without a hard limit placed within the app and without degrading the app’s quality

## 4.2 Interface

### 4.2.1 Navigation of App

**ID: 42A**

The interface should remain as clutter free as possible by including only a few buttons per screen and not overloading the screen with excess information.

### 4.2.2 Theme

**ID: 42B**

The interface should try to keep a “knitting” aesthetic by using images of yarn and knitting needles.

### 4.2.3 Aesthetic Appeal

**ID: 42C**

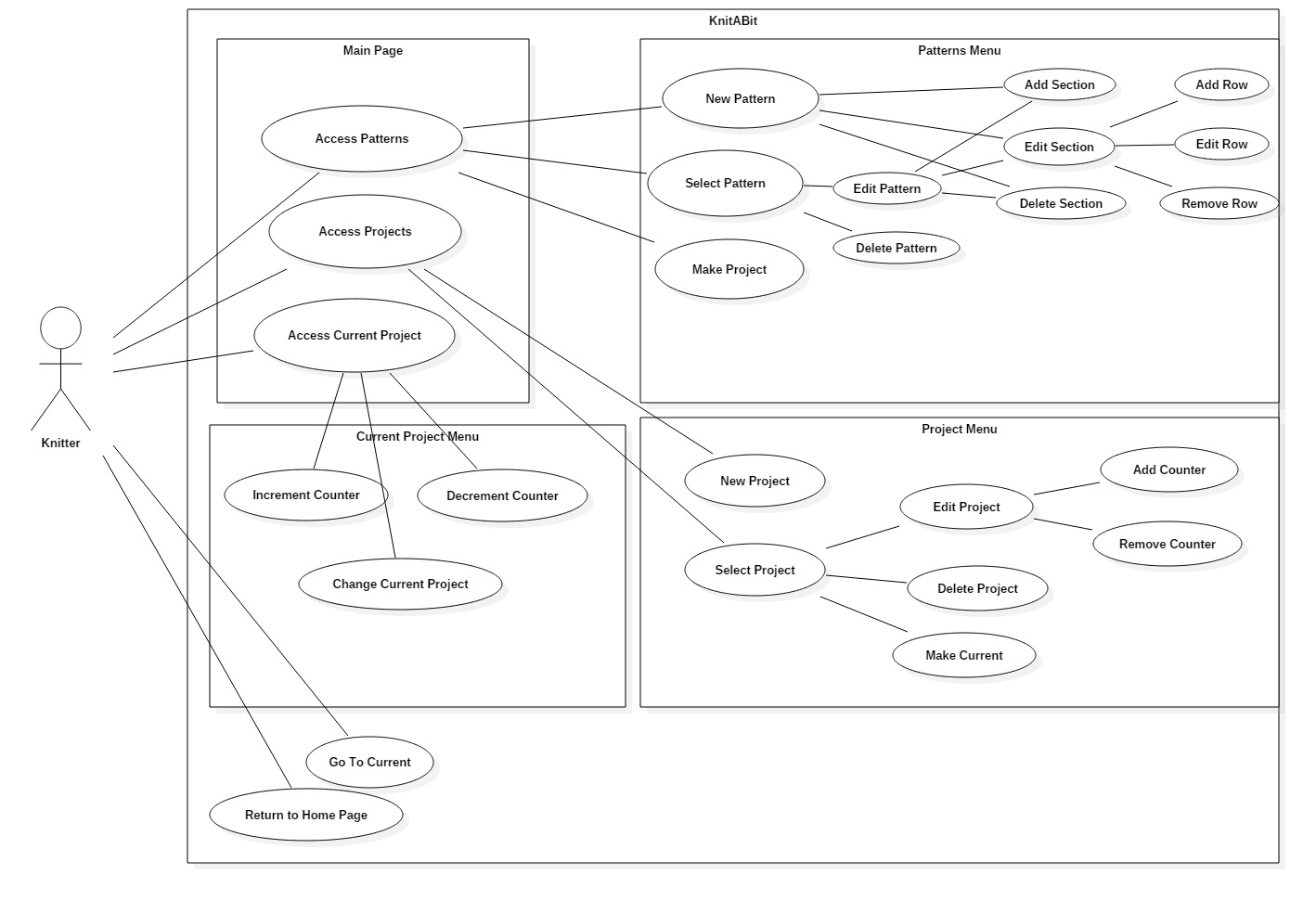
The interface should remain relatively bright by not using large patches of dark coloring for both ease of reading and general aesthetic.

# 

# 

# 5. Diagrams

## 5.1 Use case Diagram



## 5.2 UML Class Diagram

