Steepl

Collin Willis and Derek Lundy

Capstone Design Specification

Grand Canyon University

Instructor: Professor Mark Reha

Revision: 2

Date: 4/14/22

**Table of Contents**

* 1. ***Design Planning Summary***
  2. ***Overview of Design Concepts***
     1. ***Overview***
     2. **User Interface Wireframes**
     3. **Sitemap**
  3. ***Detailed Solution Architecture***
     1. **User Interaction Flow Chart**
     2. **Video Retrieval Process**
     3. **Logical/Physical Diagram**
     4. **Non-Relational Google Firestore Database Diagram**
     5. **Class Diagram**
     6. **Security**
  4. ***Project Deliverable Acceptance Log***
  5. ***Hardware and Software Technologies***

***Revision and Signoff Sheet***

**Change Record**

|  |  |  |
| --- | --- | --- |
| **Date** | **Editor** | **Revision Notes** |
| **11/21/21** | **Derek & Collin** | **Initial Version** |
| **4/14/22** | **Derek & Collin** | **Final Version:**  **- Added an item to the end of Design Planning Summary**  **- Adjusted Overview of design concepts to accurately reflect final project design**  **- Removed an unused diagram**  **- Fixed deliverable acceptance log to align with final product and statuses**  **- Corrected table of contents** |

***1.0.0. Design Planning Summary***

Steepl functions as an online church. Steepl provides a registered user with topic-organized, Christian content ranging from Bible verses and stories to videos by pastors along with comments by other users. Steepl’s goal is to provide a unique and highly personalized online church experience. Highly personalized content allows users to interact with God at their own pace and about the subjects most important to the user.

Many people do not attend church for a plethora of reasons ranging from health to availability. Steepl’s goal is to help provide a constant, free church experience to those who would not attend church in the first place as much as frequent attenders.

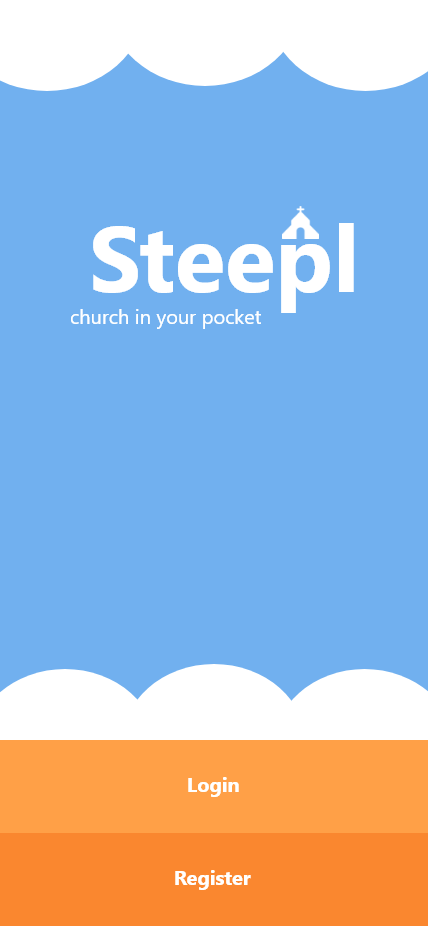
Steepl is a mobile app that gives a user access to topic-organized Christian content to provide a meaningful, online substitute or replacement for church.

Steepl should cater to the non-Christian and offer a powerful introduction to Christianity in a user-friendly manner.

***1.1.0. Overview of Design Concepts***

Steepl’s high-level design is relatively simple. Steepl’s data is stored in a non-relational Google Firebase, the application, used on a mobile device, is designed leveraging the Container/View design pattern. The application’s state will be managed using React Native Redux with reducers and a store.

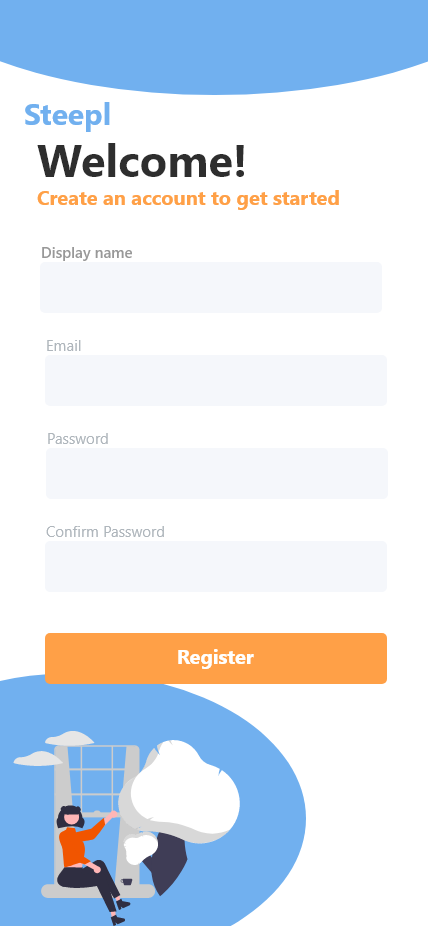
**1.1.1. User Interface Wireframes:**



Landing/Entry Page

Specifications/Requirements:

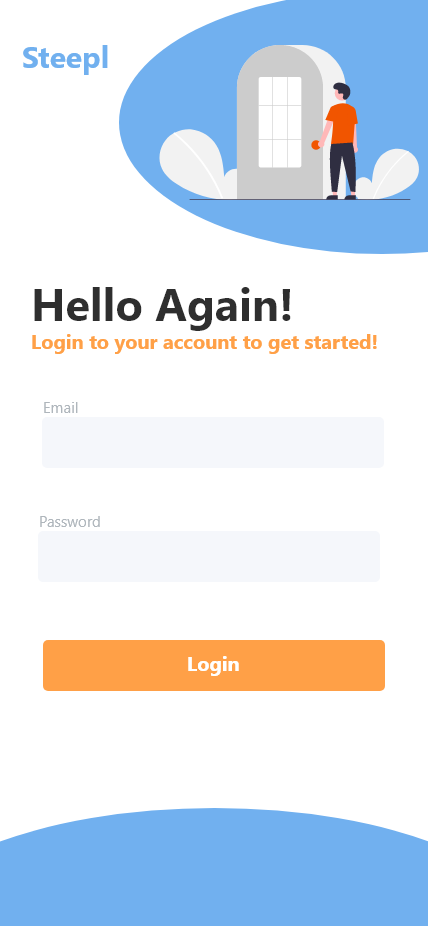
* User access register page
* User access to access login page



Register Page

Specifications/Requirements:

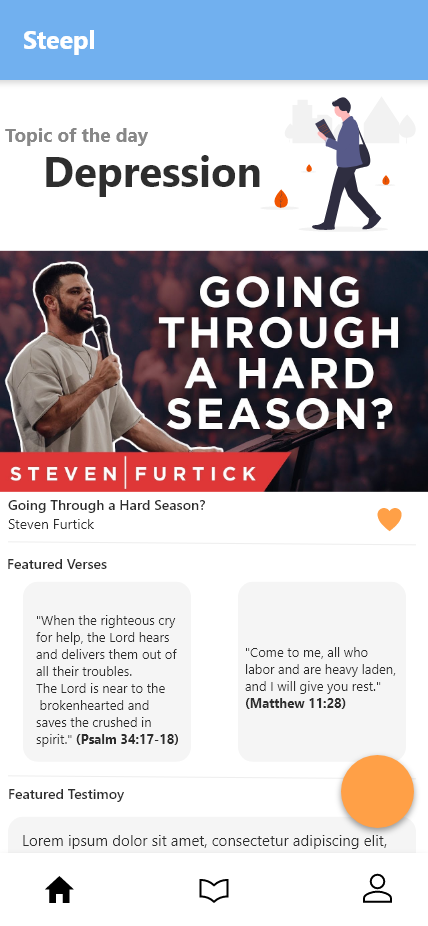
* Fields to input all required user data
* Labels for each field
* Self-descriptive titles
* Register functionality with a completed form
* Data validation per business requirements



Login Page

Specs/Reqs:

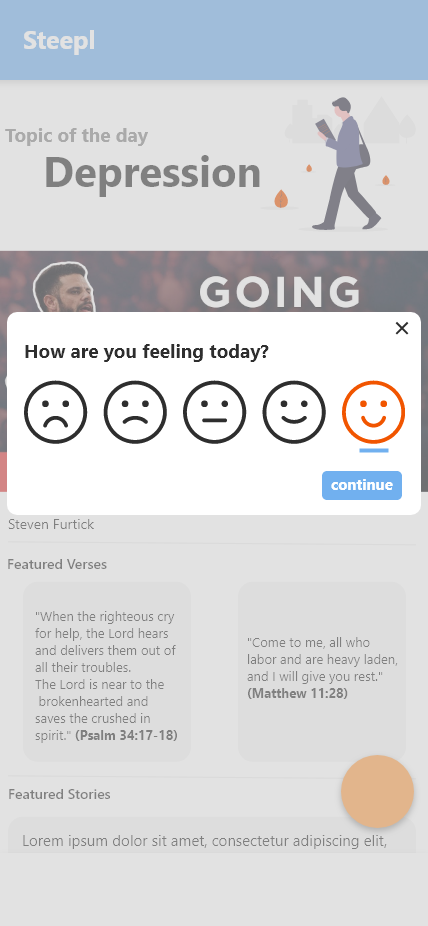
* Form fields for required data to perform a login
  + Email
  + Password
* Login functionality with completed form
* Data validation for fields
* Clear and concise prompts



Home Page

Specs/Reqs:

* Display navigation bar with access to:
  + Home Page (current)
  + HAYF Form
  + Topic Selection Page
  + Profile Page
* Display featured content in fragments throughout page
* Featured content acts as navigation to its content view page
* Featured content fragments can be favorited



How Are You Feeling (HAYF) Form

Specs/Reqs:

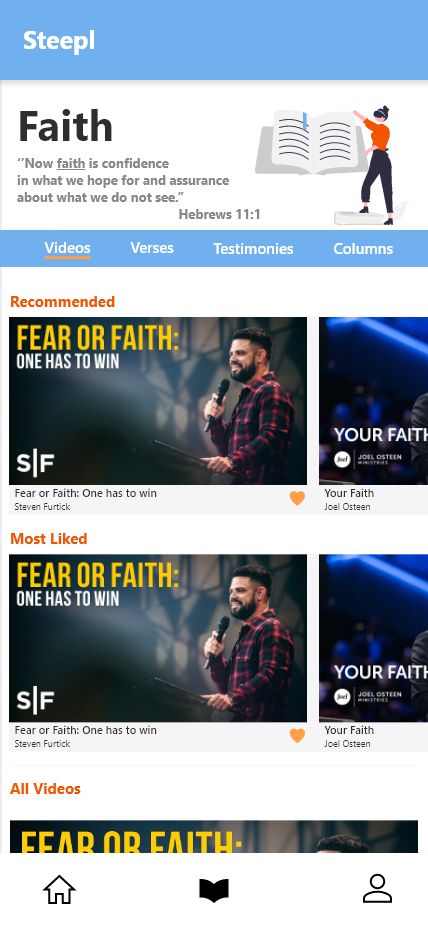
* Accessed from home page
* Clear, easy-to-understand, and concise prompts
* Radio-style, single option feeling form
* Submit functionality sends user with form data to Topic Selection page



Topic Selection

Specs/Reqs:

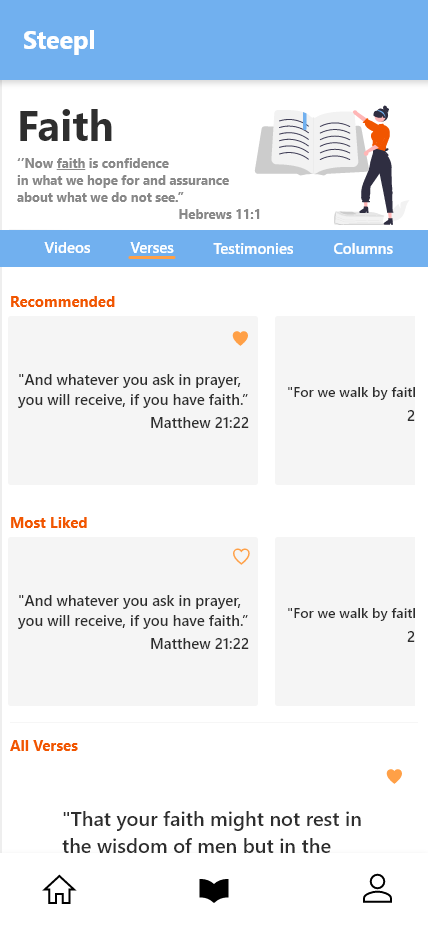
* Listed topics per business requirements/deliverables that act as navigation to a topic’s content
* List topics dynamically to accommodate HAYF form submissions
* Concise instruction
* Navigation bar



Topic Videos

Specs/Reqs:

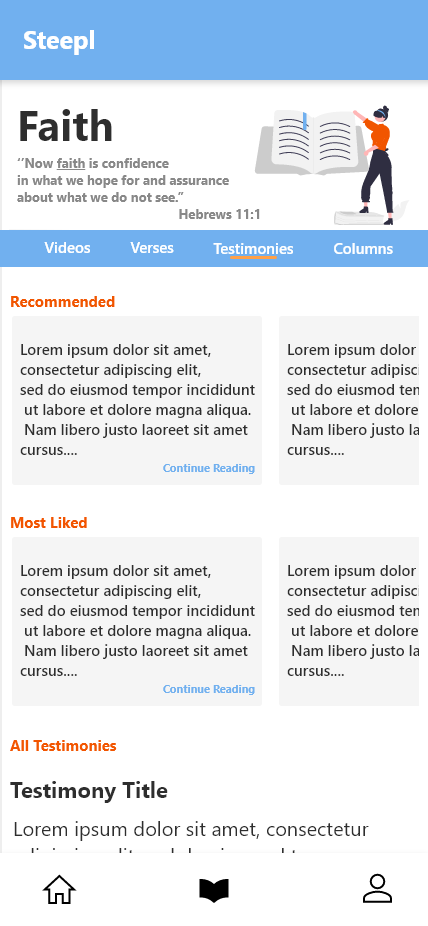
* Current topic title clearly displayed
* Efficient and intuitive navigation between types of content for current topic content types
* Display current topic’s content in fragments
  + Fragments can be organized in meaningful ways (recommended, most liked, all)
* Content in fragments can be favorited
* Navigation bar



Topic Verses

Specs/Reqs:

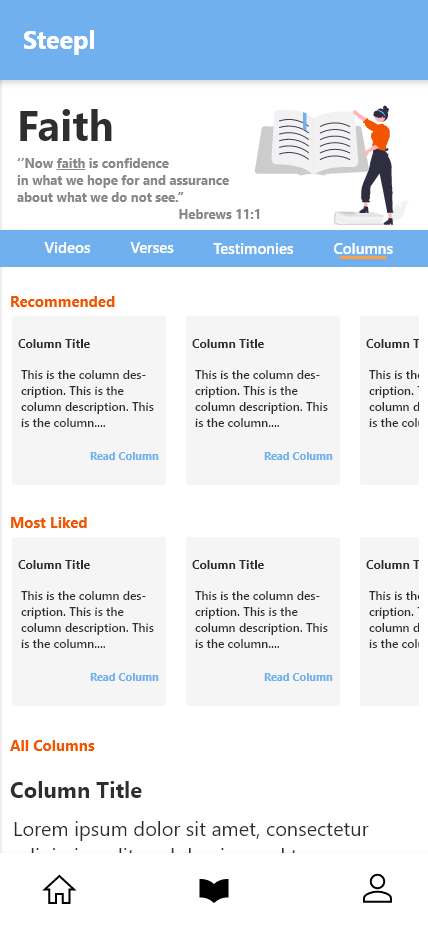
* Current topic title clearly displayed
* Efficient and intuitive navigation between types of content for current topic content types
* Display current topic’s content in fragments
  + Fragments can be organized in meaningful ways (recommended, most liked, all)
* Content in fragments can be favorited
* Navigation bar



Topic Testimonies

Specs/Reqs:

* Current topic title clearly displayed
* Efficient and intuitive navigation between types of content for current topic content types
* Display current topic’s content in fragments
  + Fragments can be organized in meaningful ways (recommended, most liked, all)
* Content in fragments can be favorited
* Navigation bar



Topic Columns

Specs/Reqs:

* Current topic title clearly displayed
* Efficient and intuitive navigation between types of content for current topic content types
* Display current topic’s content in fragments
  + Fragments can be organized in meaningful ways (recommended, most liked, all)
* Content in fragments can be favorited
* Navigation bar



Profile Page

Specs/Reqs:

* Display user display name clearly
* Display email
* Display favorited content in a scrollable format
  + Side scrolls switch content type
  + Vertical scrolls adjust content
* Label stating the displayed content are favorited content
* Navigation to user settings

**1.1.3 Sitemap**

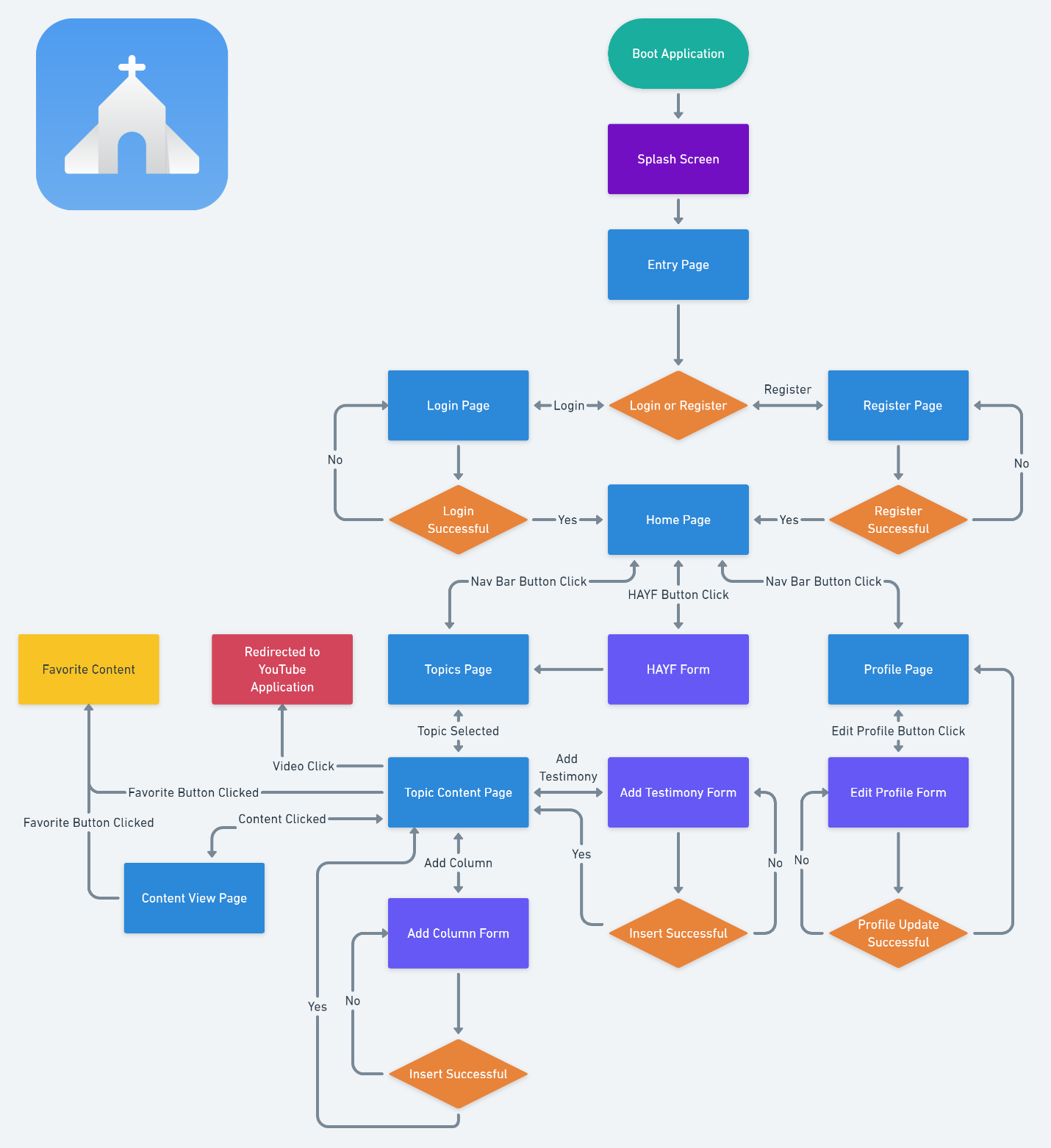
**Diagram

Description automatically generated**

This is a sitemap for Steepl. This diagram shows all pages in the app and how they are connected to each other.

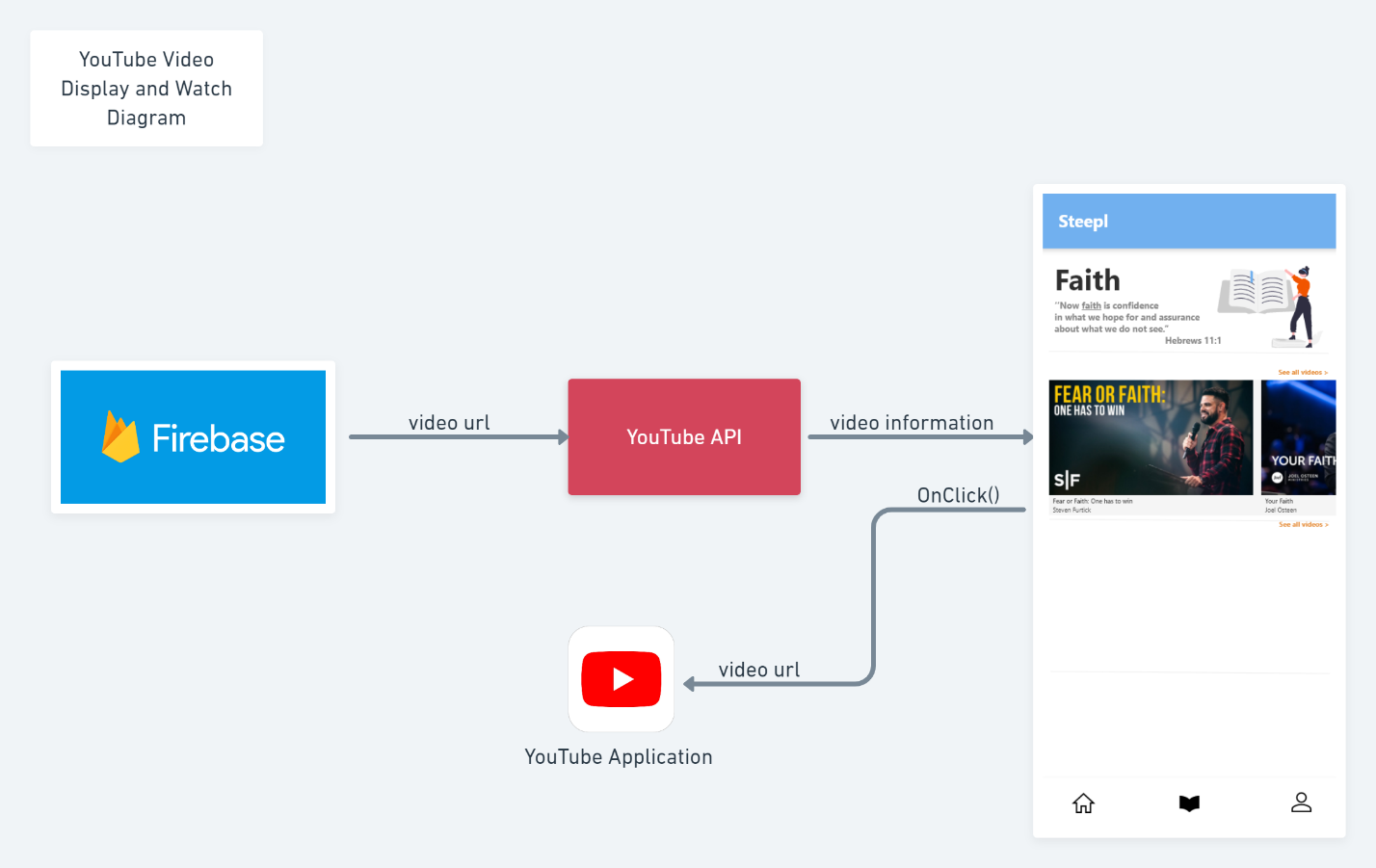
* 1. ***Detailed Solution Architecture***

**1.2.1 User Interaction Flow Chart**

****

This diagram shows the flow of the possible ways a user could interact with the application. This is *not* a sitemap.

**1.2.2 Video Retrieval Process**

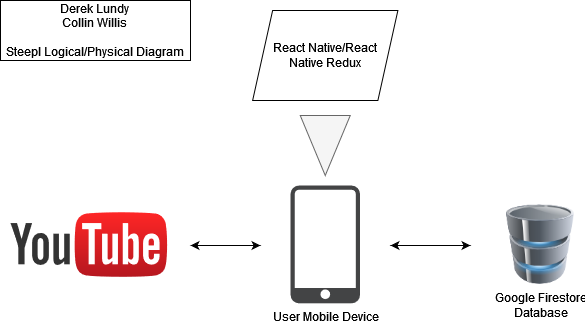
****

This diagram demonstrates the method behind how Youtube videos and their properties like title, thumbnail, creator, etc. are interacted with by the Steepl application.

Firebase stores a Youtube video ID and that is used with the Youtube Data API to return required data about the video’s properties. When a video is clicked the Steepl application will open the Youtube application on the user’s device by combining a YouTube URL with the ID to open the clicked video.

**1.2.3 Logical/Physical Diagram**

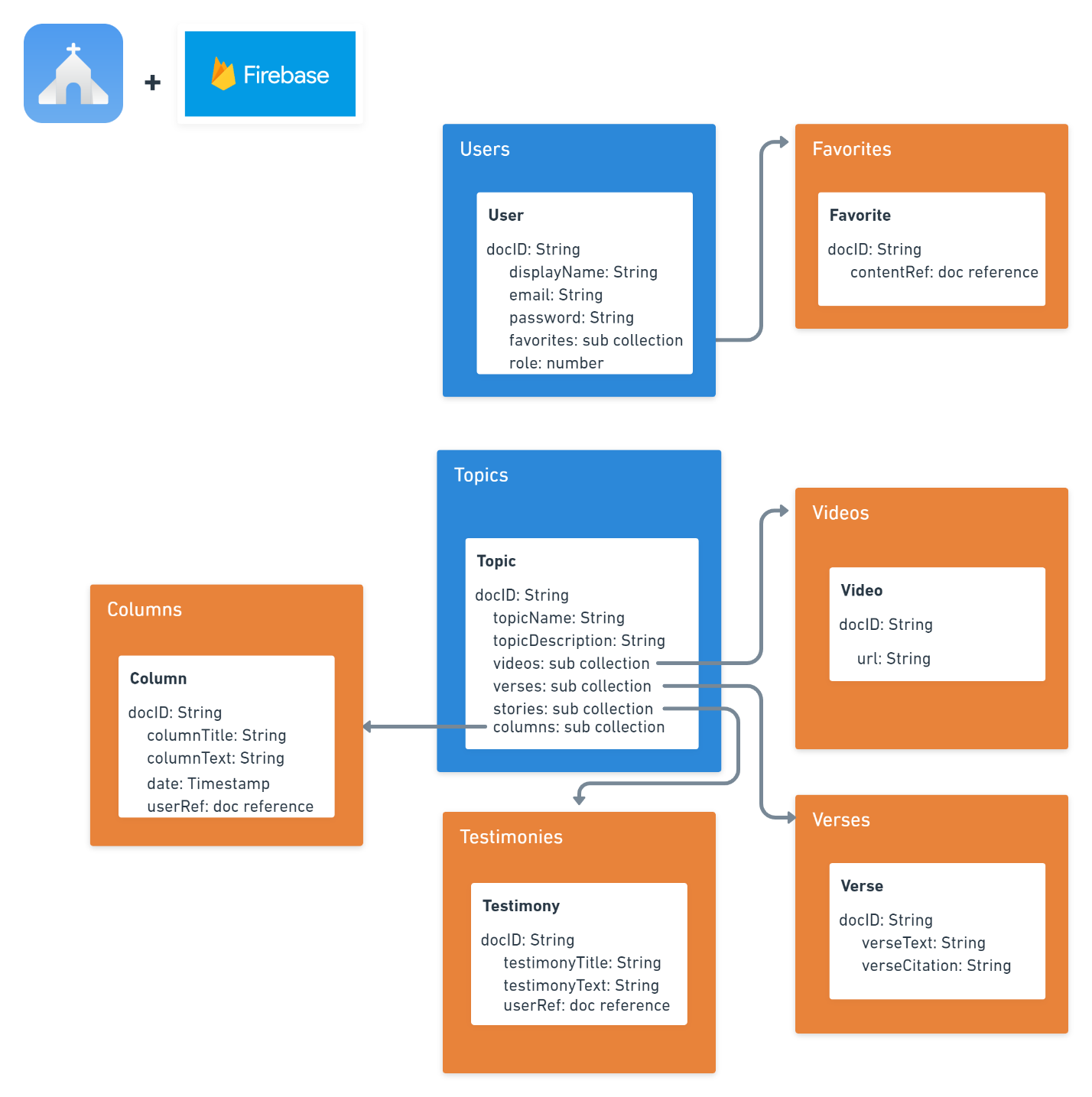
***Note: the logical and physical diagrams are too alike to make separate diagrams. One serves the function of both as the system is simple.***

****

This diagram describes the physical hardware interactions used for the application to function and the logical nature of data transfers.

Steepl uses React Native to give a user a UI to request data with, the data is stored in the cloud in a Google Firestore database, data is requested from the device via React Native, managed using React Native Redux, and if necessary, Youtube and their Youtube Data API is then used to collect more required data. After all pieces are combined, enough systems are available and interacting together to provide the envisioned user experience.

**1.2.4 Non-Relational Google Firestore Database Diagram**



This diagram represents how the data will be stored in the database. Steepl’s database is non-relational which means there is no clearly defined structure and relationships to stored data; however, this diagram helps to define what data is minimally necessary for each document stored in a collection and how the documents and collections will reference each other.

**1.2.5 Class Diagram**



Steepl uses the View/Container design pattern, therefore a model is not used in the typical object-oriented sense; however, this diagram helps to understand the structure of the classes that will be necessary for the application to function. Essentially, there are two main layers: the API layer and Actions layer. The API layer functions as the layer that interacts with the database directly like a DAO in OOP N-Layer. The Actions layer functions as a business layer, importing and using the API layer objects to access the database and performing necessary functions on views.

**1.2.6 Security**

Security is extremely important for any application. Identifying and storing only necessary data is the first step Steepl will take with security, limiting the data collected to only the essentials for if there were a compromise, nothing valuable would be lost. Steepl will leverage Google Firebase’s authentication features to securely register and authenticate existing users. Firebase also has built-in security for the actual storage of the data. NoSQL databases do not use SQL queries and therefore things like SQL injections are not a concern. The API layer in the architecture is separate from the actions layer so there is a degree of separation in the applications code. Data is secured behind the cloud’s admin functionality and therefore data validation is all that is necessary.

***1.3.0 Project Deliverable Acceptance Log***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Description** | **Comments** | **Evaluator** | **Status** |
| **1** | Database creation | This does not include data. | Professor Mark | Complete |
| **2** | Page layouts | Fully thought-out page layouts | Professor Mark | Complete |
| **3** | Create all views for layouts |  | Professor Mark | Complete |
| **4** | Register Module |  | Professor Mark | Complete |
| **5** | Login Module |  | Professor Mark | Complete |
| **6** | Home Module |  | Professor Mark | Complete |
| **7** | How Are You Feeling Module |  | Professor Mark | Complete |
| **8** | Topic Module |  | Professor Mark | Complete |
| **9** | Video Module |  | Professor Mark | Complete |
| **10** | Verse Module |  | Professor Mark | Complete |
| **11** | Testimony Module |  | Professor Mark | Complete |
| **12** | Column Module |  | Professor Mark | Complete |
| **13** | Profile Module |  | Professor Mark | Complete |
| **14** | Testimony Insert Module |  | Professor Mark | Complete |
| **15** | API Module |  | Professor Mark | Complete |

***1.4.0 Hardware and Software Technologies***

|  |
| --- |
| **Github** |
| **Android (supported platform)** |
| **IOS (supported platform)** |
| **React Native** |
| **React Native Redux** |
| **Google Firebase** |
| **Youtube Data API** |
| **Visual Studio Code** |
| **XCode** |
| **Android Studio** |
| **Jira** |
| **Node.js** |
| **Javascript** |
| **Whimsical** |
| **AdobeXD** |
| **Draw.io** |