**BIT Campus Reconnaissance**

This project aims to provide a user-friendly mobile application to digitize the traditional way of filling forms using pen and paper. Every year BIT asks the newly enrolled students to fill a paper form containing questions about the polytechnic campus. The objective of this form is to make sure that every student knows all the important information about the campus.

I developed a mobile application which makes the students to fill the same form in digital format. Anyone can easily download the mobile application and register with their email address and can fill the Reconnaissance form easily.

There are two different parts in this project – a web application and a mobile application.

1. **Web application** – Using this application, the administrator can add questions for the reconnaissance form and can generate the reports in PDF format that contains all the submitted responses of the students for those questions. This web application also provides APIs for the mobile application for accessing data relevant to the user in the database.
2. **Mobile application** – This is the application that will be used by the Polytechnic students. It fetches the questions from the database using the APIs provided by the web application and displays them in a well-designed user interface screen of the mobile application. The students provide answers to these questions which are recorded in the database.

**Part 1 - Web application**

Technologies used: Django, SQLite, jQuery

The main objective of the web application is to add questions and store them in the database and to provide APIs for the mobile application to access those added questions from the database. This application contains two main modules that will be used by the administrator (modules 1 and 2 below). Also, it provides APIs for the users of the mobile application to use.

1. Reconnaissance Module
2. Report Module
3. APIs to be used by the Mobile Application

**Reconnaissance Module:**

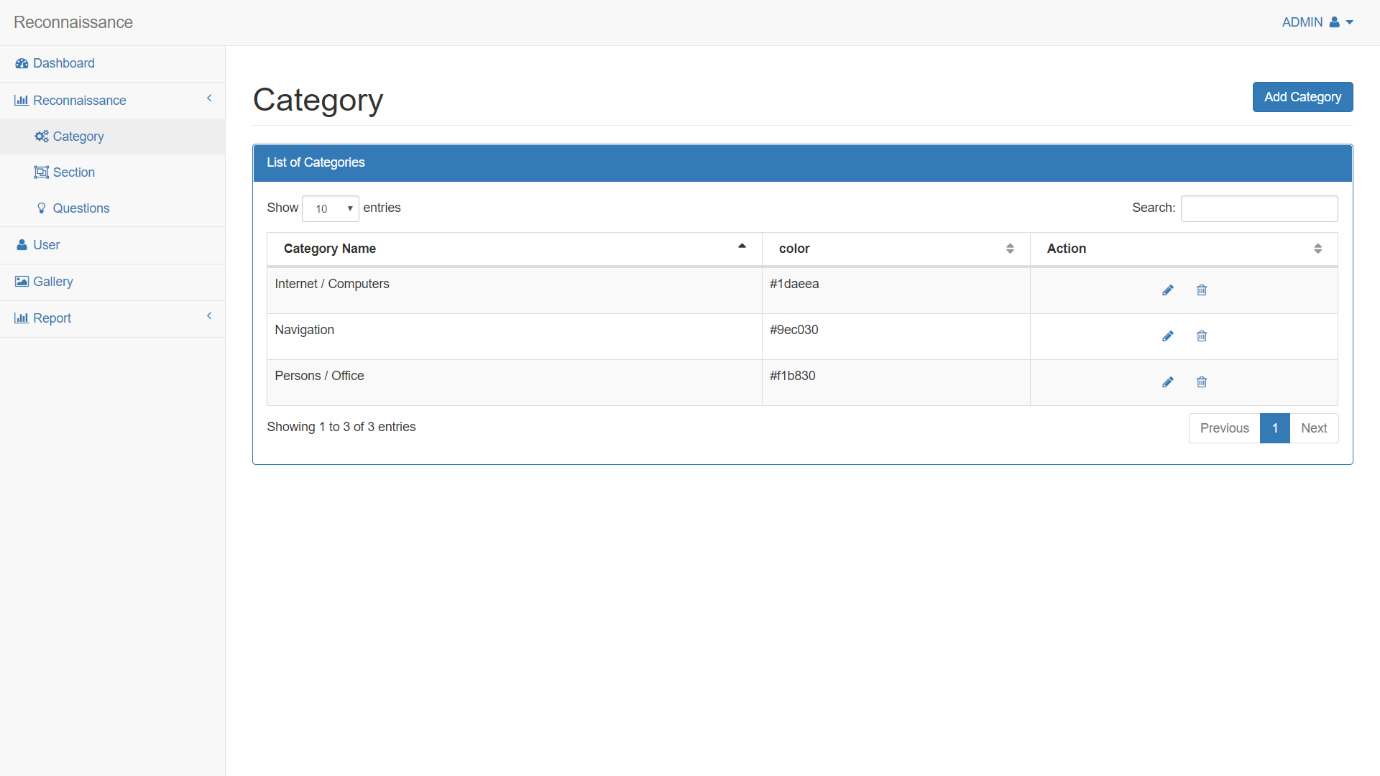
This module classified all the questions into different categories. Each category is sub-divided into sections and each section will have one or more questions. There are three types of questions available. They are:

A) **Text** – This type of questions have text boxes to type the answers. The students can type their answers in the given text box in the mobile application.

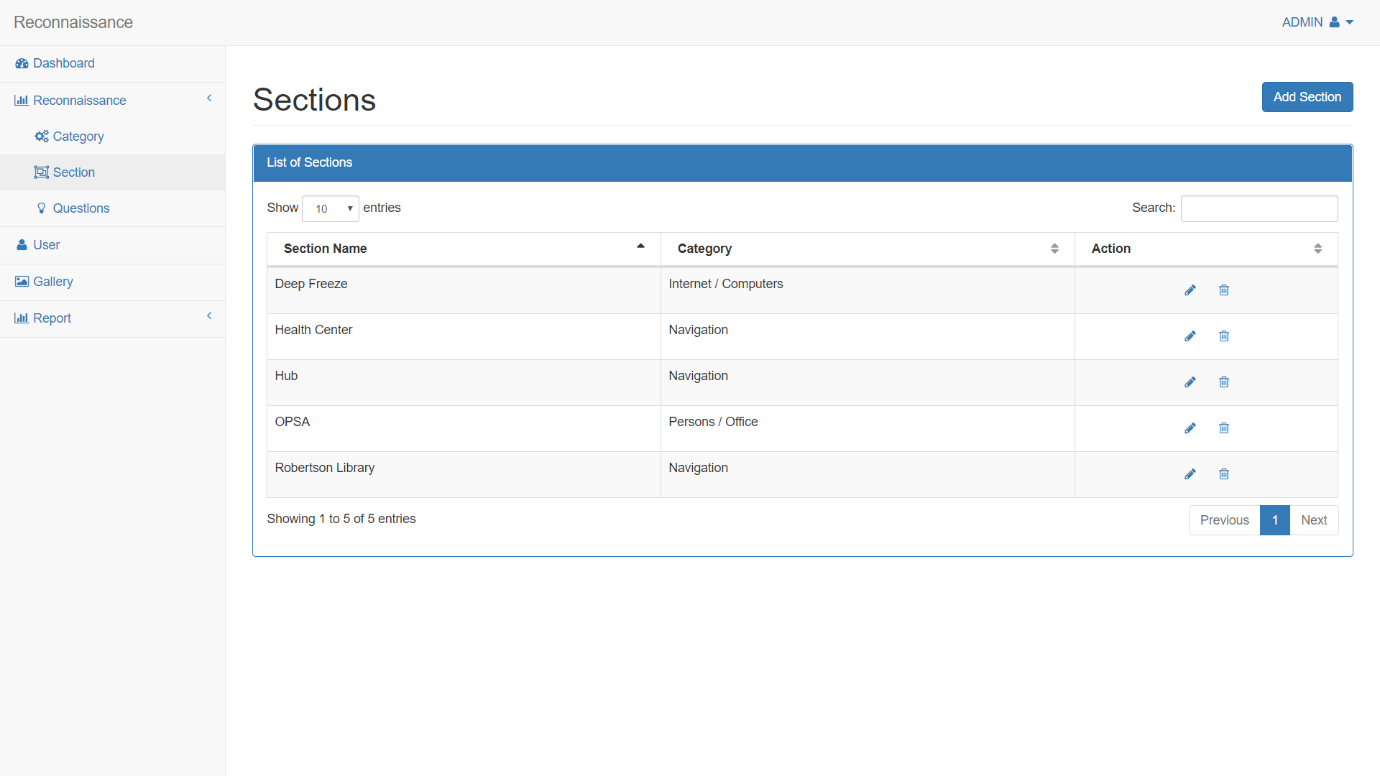
B) **Link –** This type of questions contain an URL which redirects the user to the specific web page. Students can go through that page to obtain some information regarding those questions by clicking on the links.

C) **Task** - This type of questions ask the students to find some locations in the campus. They can answer these questions by finding the place and scan the QR code pasted in that location.

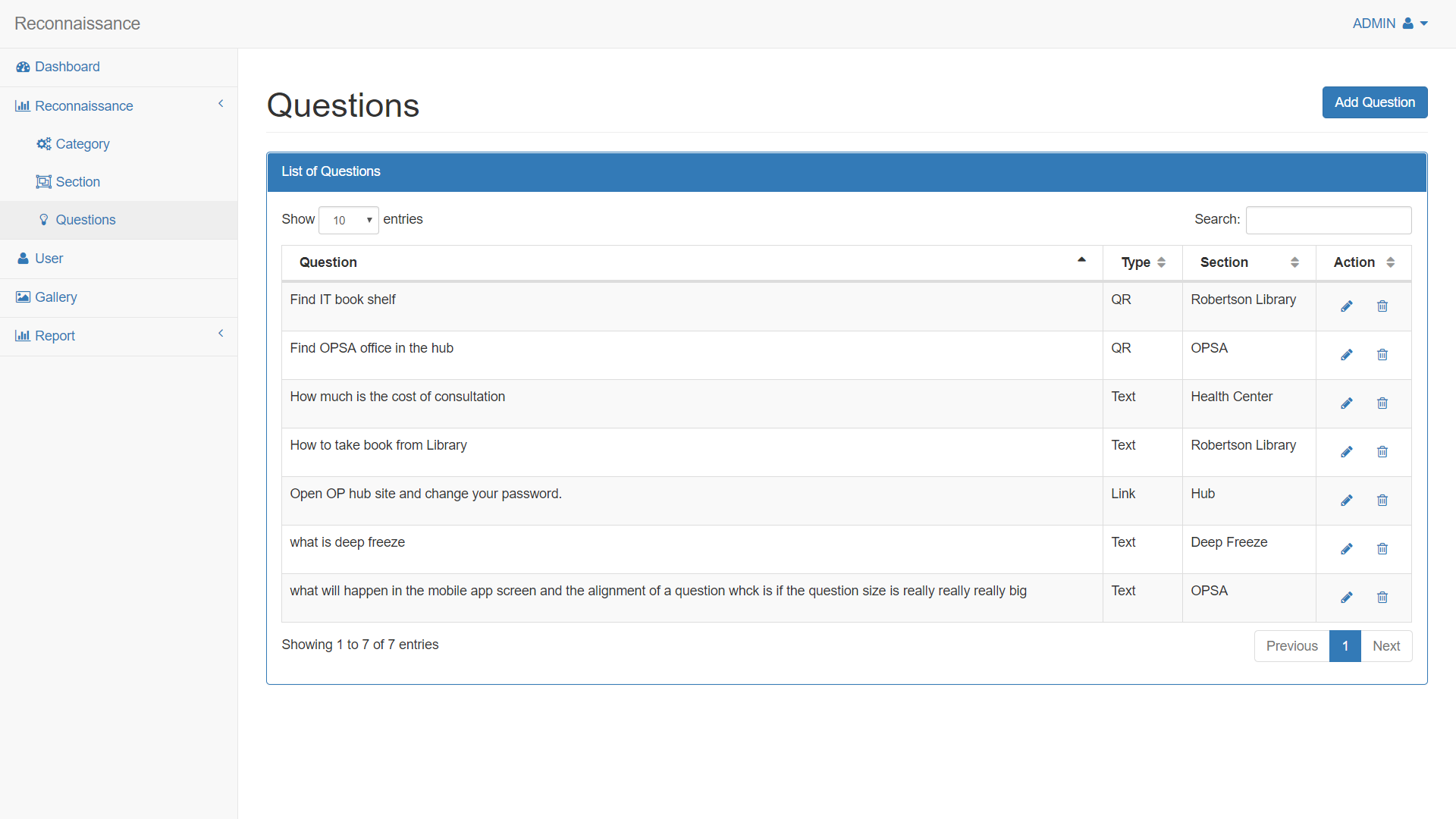
In this module, the admin user can add, update and delete the a) categories b) sections and c) questions. The screenshots for these three features are given below.



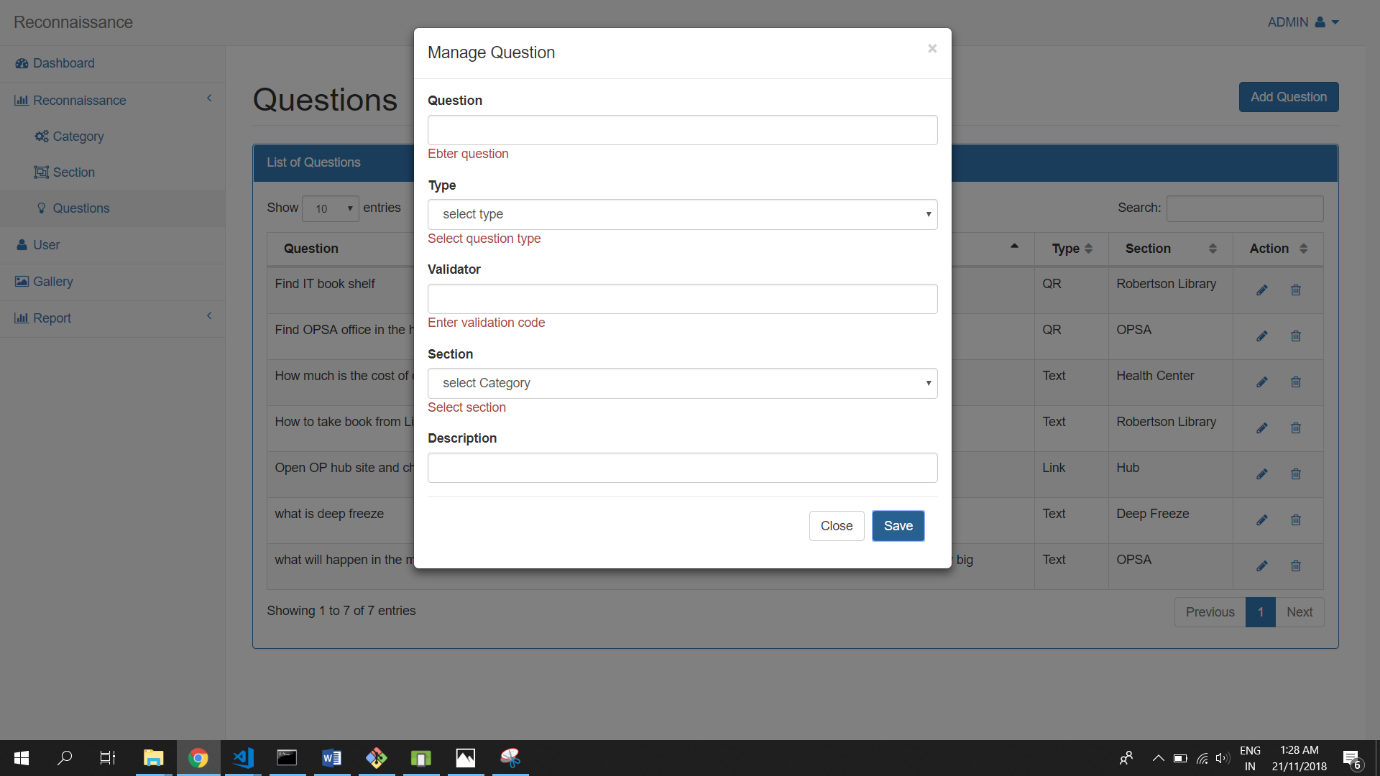
Category page



Section page



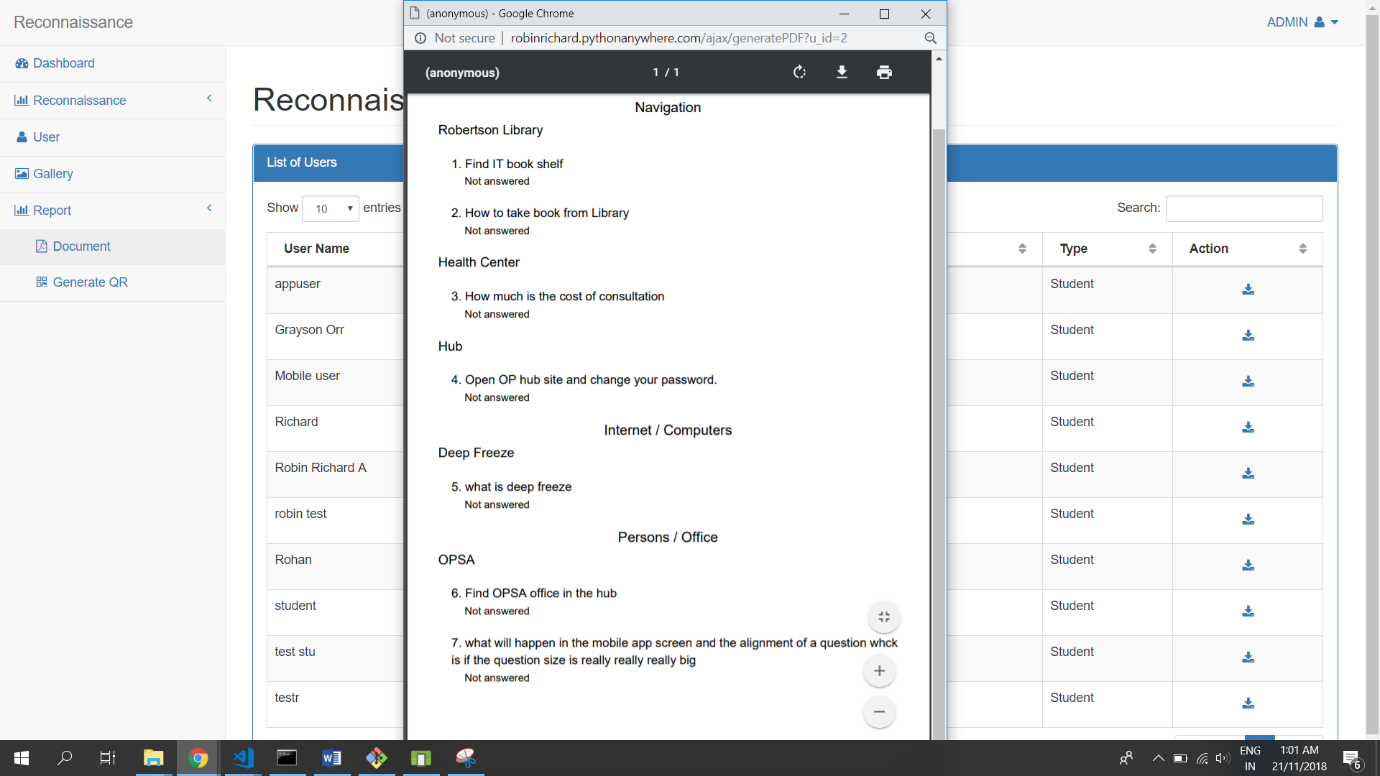
Questions page



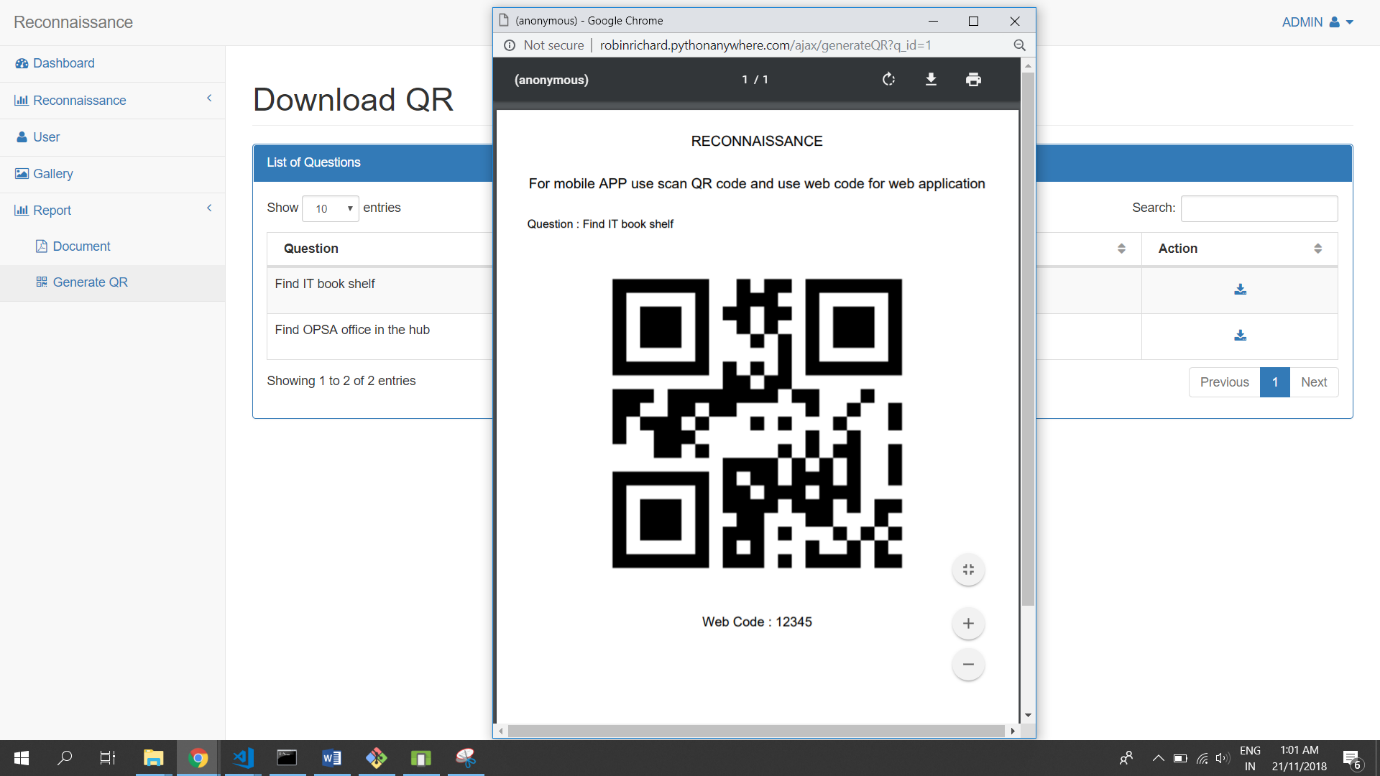
Add question popup page

**Report Module:**

Using this module, the admin user can generate two different PDF reports. Firstly, the admin can generate PDF with QR code, which will be used to validate the task type of questions. Secondly, the admin user can generate a PDF report for each student with all the questions and responses of the students for those questions.



**Sample report of questions and student responses**



**Sample report of QR generation**

**List of APIs :**

In addition to the above-mentioned two modules, this web application also has several APIs for the mobile application users for various functionalities such as login verification, list categories, section and questions, add student response to the database and fetch user data. Note that there is no user-interface for this module as it provides method descriptions. The list of APIs provided are given below.

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Functionality name** | **API** | **Parameter** |
| **1** | Login verification | “api/applogin” | Username and Password |
| **2** | Get all questions | “api/apiQuestions” | UserID |
| **3** | Reset previous answer | “api/ resetAnswer” | Reconnaissance table Unique ID |
| **4** | Update answers | “api/updateAnswer” | QuestionID,UserID,Answer |
| **5** | File upload | “api/ apiUpload” | UserID and Image File |
| **6** | Get user details | “api/getUserDetails” | UserID |
| **7** | Get answers | “api/getAnswers” | QuestionID and UserID |
| **8** | Get gallery images | “api/getGallery” | None |

**Part 2 - Mobile application**

Technology used: React Native

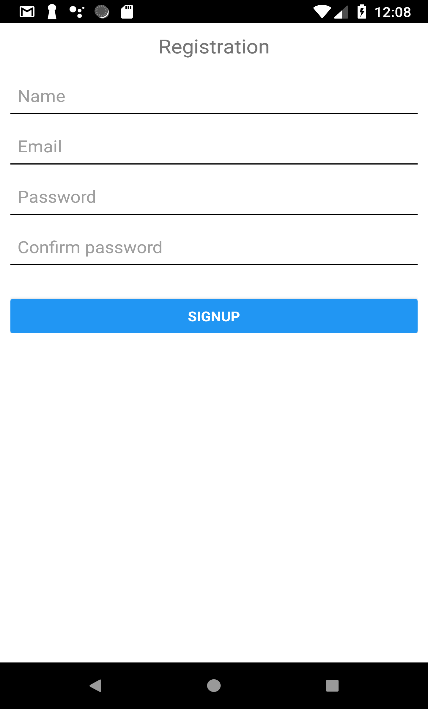
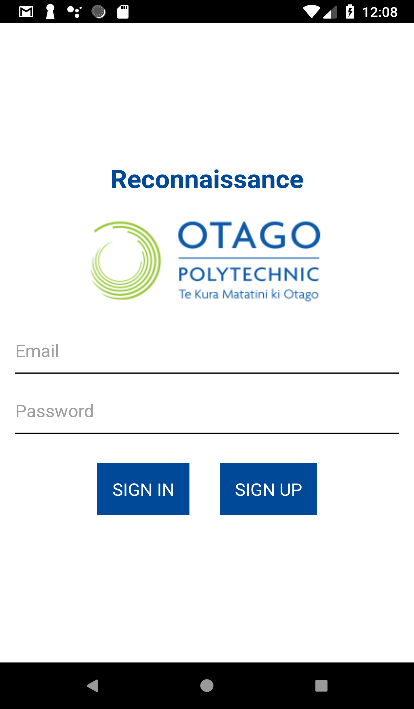
The main objective of this mobile application is to provide a user-friendly interface to fill the reconnaissance form. This application fetches data from the database. It uses the APIs of previously mentioned web application to access data from the database and insert data in to the database.

This application contains four main functionalities.

1. User registration and login
2. Display category with answered and unanswered question counts
3. Display section with completion status
4. Display reconnaissance questions based on its type

**Registration and login screens:**

The students can easily register in this application and login with their email and password. The user needs to activate their account before they login by clicking the link that has been sent to their email at the time of registration. Users can upload profile picture of their choice.

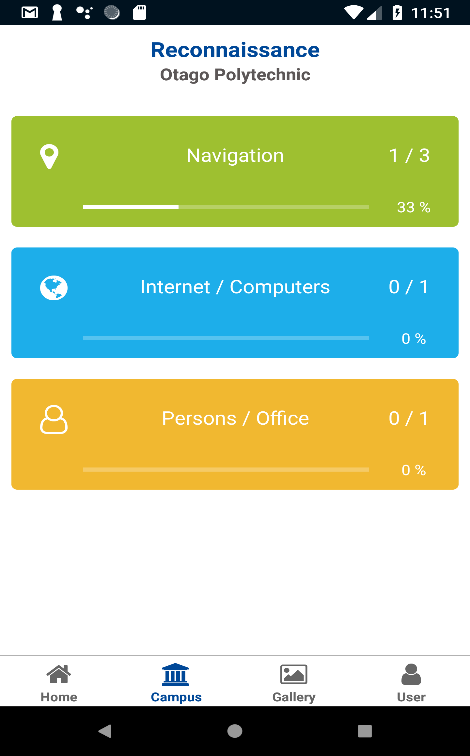


Login screen

Registration screen

**Category screen:**

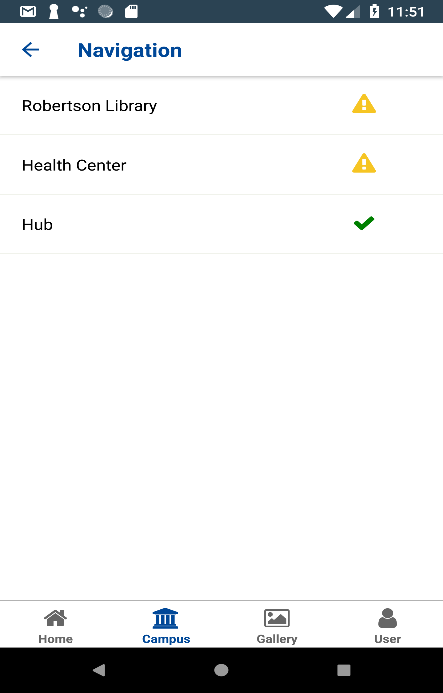
All the categories are displayed with icons and background colour mentioned in the database. Along with the category name, count of total questions in that category and number of questions a student has answered on that category is shown. A progress bar denoting the percentage completion and a pop up describing the functionalities of the screen are also included in this screen.

****

Category screen

**Section screen:**

All sections under the selected category are displayed in a list view provided by React Native. Each section displays the status that shows whether all the questions in that section are completed or not (see next snapshot).



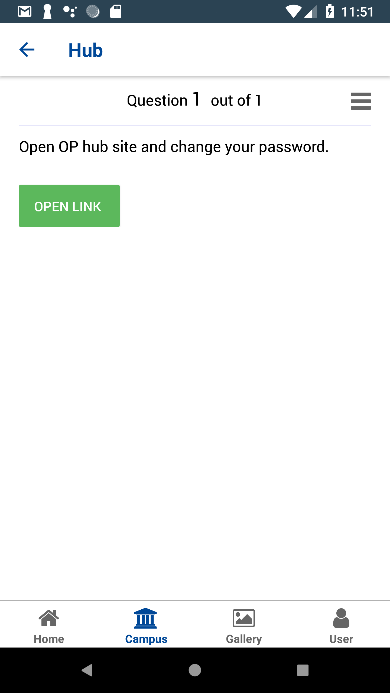
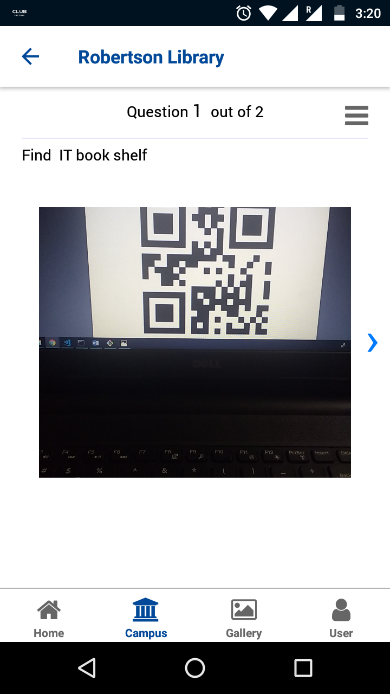
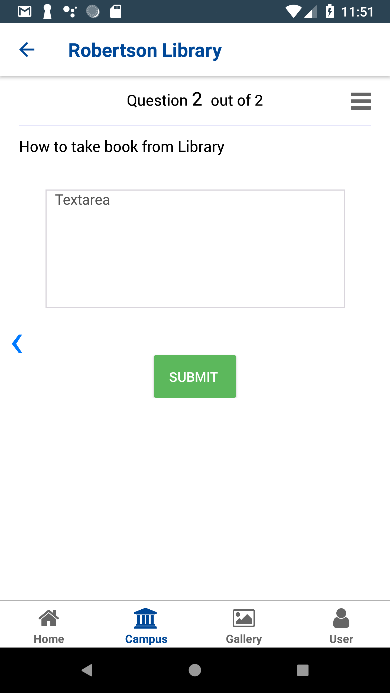
Section screen

**Reconnaissance screen:**

This screen provides different controls based on the question type.

1. Multiline textbox is provided for text type questions (see Figure a below)
2. Link button for link type question (see Figure b below)
3. QR reader for QR task type question (see Figure c below)

Students can view the description for these questions and provide answers by entering text or clicking on the link or taking a picture of the QR code. They can also reset their submitted answers by clicking on the menu (on the top-right of the screen) which provides the reset option.

****

1. Multi-line text box for providing answers for text type questions

c) A QR reader for QR task type question to take a picture of QR code at a particular location.

b) A Link button for link type question for students to obtain information by following the link

**How to run the application?**

Clone the repo using the following URL :

<https://gitlab.op-bit.nz/BIT/Project/Mobile-Development/BITCampusReconnaissance.git>

1. Django web application installation and running commands
   1. cd Reconnissance
   2. pip install -r requirements.txt
   3. python manage.py runserver

The server will run on <http://127.0.0.1:8000>

Login Credentials:

User name: admin

Password: 123

1. React Native mobile application installation and running commands
   1. cd BITCampus
   2. npm install
   3. React-native run-android (run any android emulator or connect mobile device using USB to the machine before running this command)