Computer Science and Engineering Senior Theses   Student Scholarship

6-6-2015

Sankara Eye Records: eye-health tracker for children in India

Francis-Joseph Cuenca
Santa Clara University

Amy Truong
Santa Clara University

Recommended Citation

SANTA CLARA UNIVERSITY
DEPARTMENT OF COMPUTER ENGINEERING

Date: June 6, 2015

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY

Francis-Joseph Cuenca
Amy Truong

ENTITLED

Sankara Eye Records: Eye-Health Tracker for Children in India

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF

BACHELOR OF SCIENCE IN COMPUTER SCIENCE AND ENGINEERING

_____
Thesis Advisor

_____ 6/9/15
Department Chair

Sankara Eye Records: Eye-Health Tracker for Children in India

by

Francis-Joseph Cuenca
Amy Truong

SENIOR DESIGN PROJECT REPORT

Submitted in partial fulfillment of the requirements
for the degree of
Bachelor of Science in Computer Science and Engineering
School of Engineering
Santa Clara University

Santa Clara, California
June 6, 2015

Sankara Eye Records: Eye-Health Tracker for Children in India

Francis-Joseph Cuenca
Amy Truong

Department of Computer Engineering
Santa Clara University
June 6, 2015

ABSTRACT

Sankara Eye Records is a web application created for Sankara Eye, an Indian-based outreach organization that provides eye care services for underserved communities. Our goal is to make the process of creating and editing screening site records and schoolchildren eye health records quick and efficient. This will enable Sankara Eye to track childrens eye health over their school years. Our system will include a web application to be used by trained Sankara Eye personnel at screening sites and a backend database to store all their data.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Sankara Eye is an Indian-based outreach organization that operates and manages eye care hospitals all over India primarily for low-income families. In order to provide services to these under-served communities, trained Sankara Eye personnel, or field screeners, are sent into the field to examine patients who have eye problems and are in need of care at the hospital. Although this process has been servicing Sankara Eye and their patients well, Sankara Eye wants a better system to keep track of schoolchildren's eye health records that is user-friendly and efficient. Currently, the process of tracking the student's eye health records requires schoolteachers to first screen their students for any eye conditions. If the schoolteacher identifies that a child in his or her class has symptoms of a certain eye condition, the teacher must record the childs name, his or her basic information, and his or her eye condition on paper and submit the form to Sankara Eye. Field screeners will then be notified prior to their visit to the screening site: a school, camp, or an aganwadi camp. They will do a second screening on the students who have been identified with eye problems and need further assistance. The field screeners will need to manually fill out a form for each student and electronically scan and send the form to the nearest clinic. They have to fill out forms for every student they screen every year, because these eye health records are not updated throughout their school years. The child's age and class level stay the same on their forms. Therefore, screeners need to repeat this process annually. The current process is tedious and time-consuming for the field screeners. Our solution is Sankara Eye Records, a web application that automates this process and do so in a more efficient way for the field screeners. Sankara Eye Records allows the screeners to easily input and update information about the screening sites and the students through a simple interface. With a tap of a button, the screener will be able to input the student's name, age, gender, grade level, guardian name, home address, demographic, and details on their eye condition,

quickly and efficiently. They can input additional information about each student into their file and determine if the student needs to be brought in to the hospital for additional testing or surgery. This web application also features a tutorial on how to use the application. Sankara Eye Records will allow Sankara Eye screeners to easily access and input schoolchildren's eye health records in real time.

Chapter 2

System Requirements

2.1   Requirements

After speaking with the Sankara Eye President of Community Outreach, Bharath Balasubramaniam, we agreed upon the following requirements for our project. There are two types of requirements: functional and nonfunctional. Functional requirements describe what our system needs to do and nonfunctional requirements describe how the functional requirements will be implemented.

Functional

Critical

- The system will store screening site or student information into the database.

- The system will remove screening site or student information from the database.

- The system will categorize students by the screening site they attend or live at.

- The system will input information by keyboard input.

- The system will automatically update the age of each student based on their birth date.

- The system will automatically update the grade of each student every year.

Recommended

- The system will authenticate the user before allowing access to sensitive information.

- The system will be supportive of eight different Indian languages and English.

Non-Functional

Critical

- The system will be portable to different mobile platforms.

- The system will be easy to use.

- The system will be fast and responsive.

- The system will be intuitive.

Recommended

- The system will have low distribution cost.

Chapter 3

Use Cases

Use cases define the steps required to accomplish a specific goal in a system. From our understanding of our list of requirements, our mobile application will revolve around four main use cases as illustrated in Figure 3.1: add screening site or student records, edit screening site or student records, search screening site or student records, and how to use the application.

3.1   Use Cases

Case 1: Add screening site or student record

Goal: The screener will have the ability to add screening site or student records. Screening site information includes the name of the screening site, date of screening, name of the principal or headmaster, the screening site address, and contact details of the screening site and the person in charge. Student information includes student's first name, last name, age, gender, birthdate, grade, and other basic information as well as details regarding their eye condition.

Actor: Sankara Eye screener

Preconditon(s):

1. Have to be logged in with valid credentials.

Postcondition(s):

1. An alert will be displayed to notify the screener of a successful addition to the database.

Steps Required:

1. Open the application.

2. Select a language.

3. Log in with valid credentials.

4. If adding new screening site records, navigate to the side menu and select + Anganwadi Profile', + Camp Profile', or + School Profile'. If adding new student record, navigate to the side menu and select + Student Profile'.

5. Input screening site or student information into input fields.

6. Select Add button.

7. Confirm decision.

Exception(s):

1. Screening site or student is already added into the system.

Solution(s):

1. Error message will appear and screening site information will not be added into the system.

Case 2: Edit screening site or student record

Goal: The screener will have the ability to edit existing screening site or student records.

Actor: Sankara Eye screener

Preconditon(s):

1. Have to be logged in with valid credentials.

2. Screening site or student record need to be edited must already exist in the system.

Postcondition(s):

1. An alert will appear to notify the screener the screening site or student record was successfully updated in the database.

Steps Required:

1. Open the application.

2. Select a language.

3. Log in with valid credentials.

4. Search for the record that needs to be edited. If a screening site record needs to be edited, select the side menu and under search, select the type of screening site that needs to be edited. If a student record needs to be edited, select the side menu and select student master search.

5. Select the record and tap edit at the top right corner.

6. Add the information that needs to updated in the input fields.

7. Tap Edit School or Edit Student at the bottom of the input fields.

8. Confirm decision.

Exception(s):

1. Multiple students with the same name are suggested when the screener is searching for a particular student.

Solution(s):

1. When a list of students with the same name is suggested, the screener will be able to identify which the student record they are looking to edit by the student's profile picture or guardian information that is also displayed.

Case 3: Search screening site or student record

Goal: The screener will have the ability to search an existing screening site or student record.

Actor: Sankara Eye screener

Preconditon(s):

1. Have to be logged in with valid credentials.

2. Screening site or student record to be edited must already exist in the system.

Postcondition(s):

1. Screening site or student information that was searched will be displayed on the screen.

Steps Required:

1. Open the application.

2. Select a language.

3. Log in with valid credentials.

4. If a screening site record needs to be viewed, select the side menu and under search, select the type of screening site record. If a student record needs to be viewed, select the side menu and select student master search.

5. Type in record name in the search bar at the top.

6. Select the record to be viewed.

7. View screening site or student information.

Exception(s):

1. Multiple students with the same name are suggested when the screener is searching for a particular student.

Solution(s):

1. When a list of students with the same name is suggested, the screener will be able to identify which the student record they are looking to edit by the student's profile picture or guardian information that is also displayed.

Case 4: How to use the application

Goal: The screener will view a tutorial on how to use the application.

Actor: Sankara Eye screener

Preconditon(s):

1. Have to be logged in with valid credentials.

Postcondition(s):

1. The last page in the tutorial will notify the screener they have finished viewing the tutorial.

Steps Required:

1. Open the application.

2. Select a language.

3. Log in with valid credentials.

4. Select the side menu and select Tutorial'.

5. Swipe left to learn how to use the application.

6. Select Back' at the top left any time to leave the tutorial.

Exception(s):

1. User still has unanswered questions regarding the use of the application.

Solution(s):

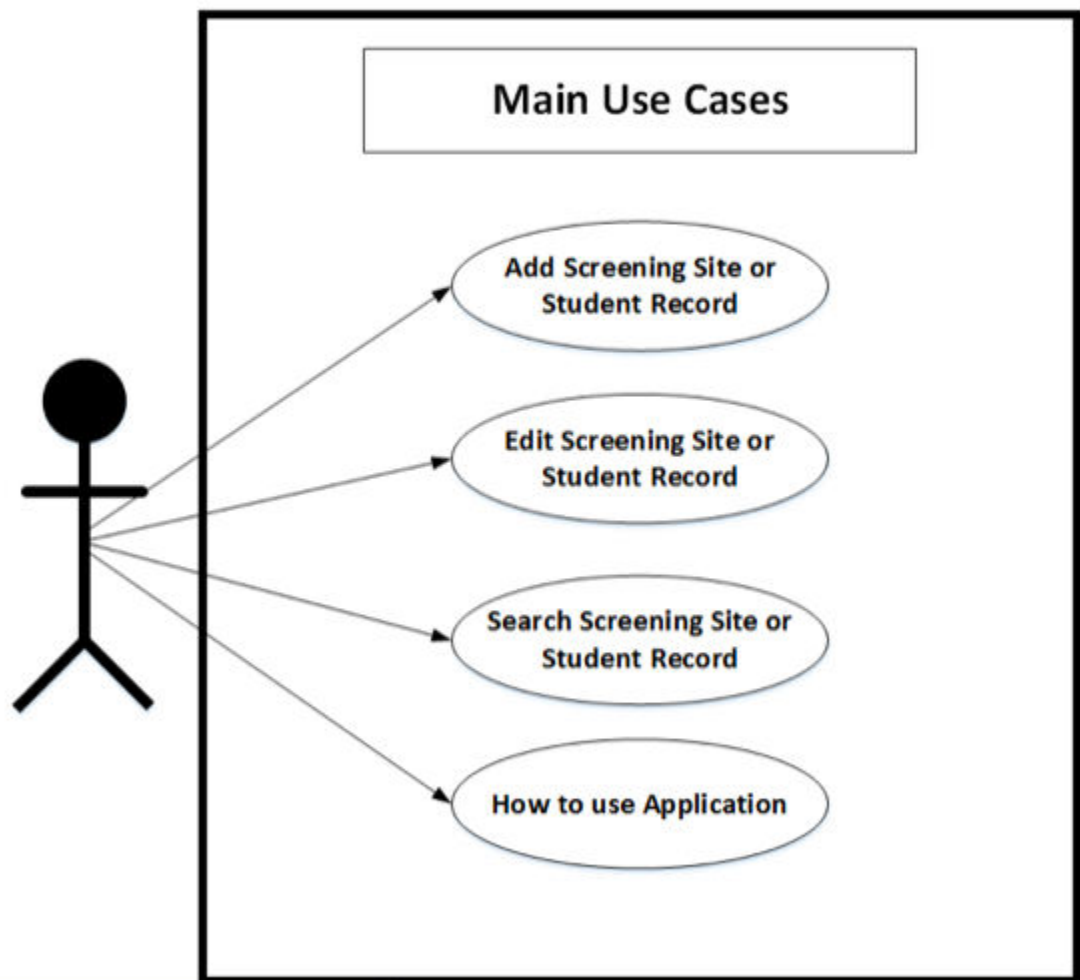1. User has option to email the creators of the application for assistance.

Figure 3.1: Use Cases for Screener

Chapter 4

Design and Implementation

4.1 Screener Interface

Our goal is create a user interface that is simple, functional, and intuitive for the screener. The following screenshots show different views of the web application and the different functionalities the screeners will have the option to use. Those functions include choosing a language, logging into the web application to access information, and add or view a screening site or student record. The screenshot below (Figure 4.1) shows the first screen the screener will see when they open the web application. They will have the option to use the web application in nine different languages. The option to choose a language allows screeners from different Indian states to navigate the web application with ease.

Figure 4.1: Choose a Language Screen

Figure 4.2: Login Screen

The screenshot in Figure 4.2 is the login screen where the screener will authenticate him or herself. They must enter their unique login and password in order to access and add information to the database. If they have not yet created an account, they can click on the Create Login button to create a unique username and password.

Figure 4.3: Create Account Screen

Next screenshot (Figure 4.3) displays the create account page. Here the screener can create a unique username and password. It is required they enter a valid pin number that only Sankara Eye personnel knows in order to secure the screening site and student information.
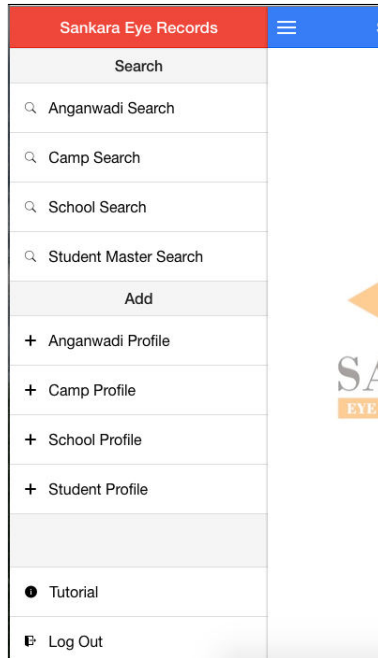
11

Figure 4.4: Side Menu

The screenshot shown in Figure 4.4 is the side menu screen where the screener can choose to either search or add a screening site or a student record, learn how to use the web application, or log out. After the screener has tapped School Search, the screen shown in Figure 4.5 is presented showing a list of schools that Sankara has already screened. Similar screens will be displayed if the screener chose to search for an Anganwadi site or a camp site. Next, the screener can select the screening site they want to view. The screener will see the screening site profile (Figure 4.6). The profile will include information such as the screening site name, the date the screening site was screened, the headmaster name, the screening site address, the screening site phone number, and the co-sponsor name.

The screener can then choose to go back to the list of search suggestions to search and/or view a different screening site record. He or she can also click the side menu to add a new screening site. To edit a screening site profile, the screener can tap on the top right button on the screening site profile page (as shown in Figure 4.6). The screener will then see a screen (Figure 4.7) with a form they can fill out.

Figure 4.5: View List of Screening Sites

Figure 4.6: View a Screening Site Record

Figure 4.7: Edit a Screening Site Record

Figure 4.8: Add a Screening Site Record

To add a new screening site record, the screener must select the type of screening site record they would like to add from the side menu. Figure 4.8 shows what the screen looks like if a screener chose to add an Anganwadi site record. To view the list of students who attend a specific screening site, the screener simply has to tap the button View Student List' at the top of the screening site record shown in Figure 4.6. The screener will then be presented with a list of students as seen in Figure 4.8. They will have the option to go back to the screening site record or search and view a

14

student record by tapping on the student name. After tapping on a student name, the screener will be shown the student record (Figure 4.10), which will display the student basic information and eye health information. To edit a student record, the screener can tap on the top right button on the student record screen (as shown in Figure 4.9).

Figure 4.9: View List of Students

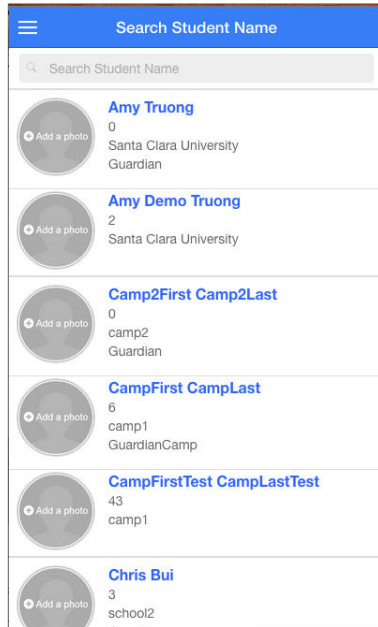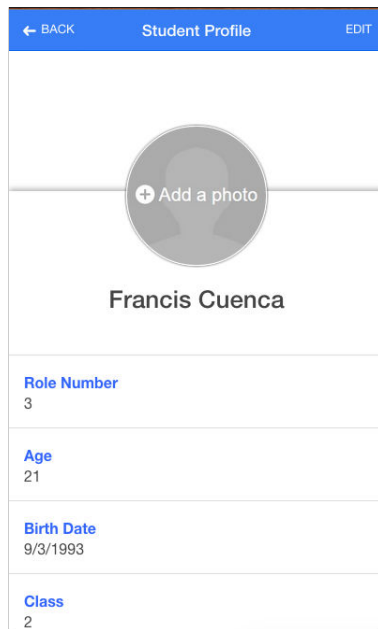Figure 4.10: View a Student Record
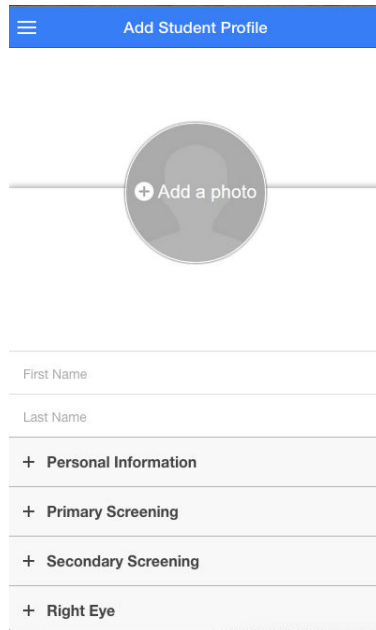
Figure 4.11: Add a Student Record

To add new student record, the screener will tap the side menu and select + Student Profile' and be shown the screen in Figure 4.11.

4.2 System Design

The system design of our web application involves an interaction between the web application and the database as shown in Figure 4.12. Through the use of the web application, the screener will be interacting with the the database. The screener will be able to log into the web application, access the screening site or the student information by sending a requesting from the database, add a new screening site or student record by storing the information to the database, and learn how to use the web application by viewing the tutorial that is already integrated on the web application itself. To log into the web application, the screener will send an authentication request to the back end by providing their first and last name and their personal identification number. The back end will then take the provided information and compare it with the information in the database to authenticate the screener. To add a screening site or a student record, the screener will type in the information into the input form and the backend will then store the information into the appropriate table in the database, depending if the screener is adding a screening site record or a student record. To edit a screening site or student record, the screener sends a request to the backend to retrieve the record from the relevant database, update the record, and save the new information. The backend

will then store the newly provided information in the appropriate table. To search for a screening site or student record for viewing, the screener needs to send a request to the backend to retrieve the information from the database. Lastly, to learn how to use the application, the screener needs to open the tutorial that is already integrated on the web application.
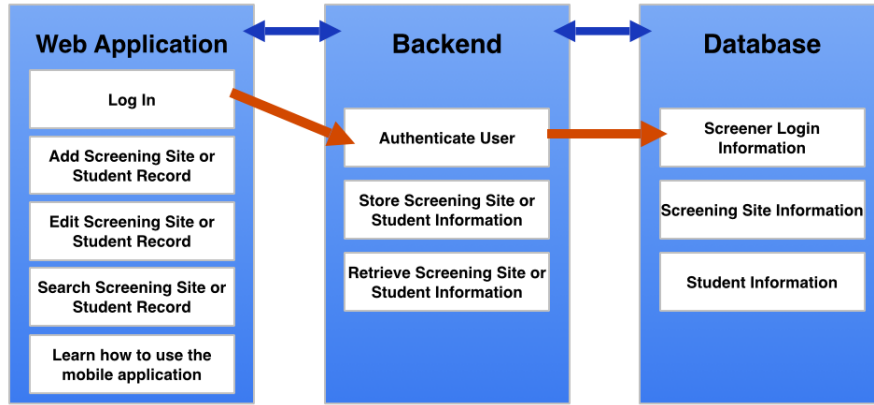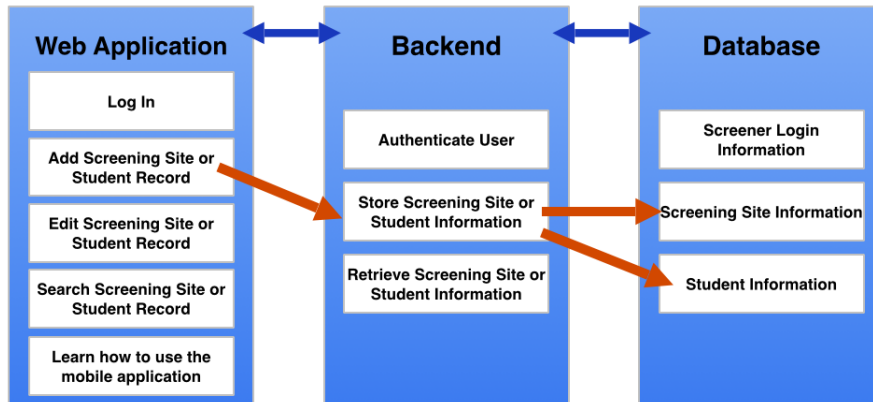

Figure 4.12: System Diagram to Log In

Figure 4.13: System Diagram to Add a Screening Site or Student Record
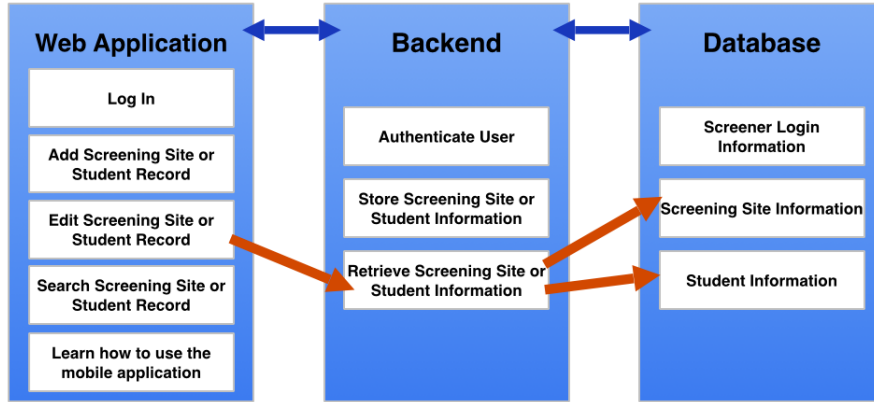
17

Figure 4.14: System Diagram to Edit a Screening Site or Student Record - Retrieve Record
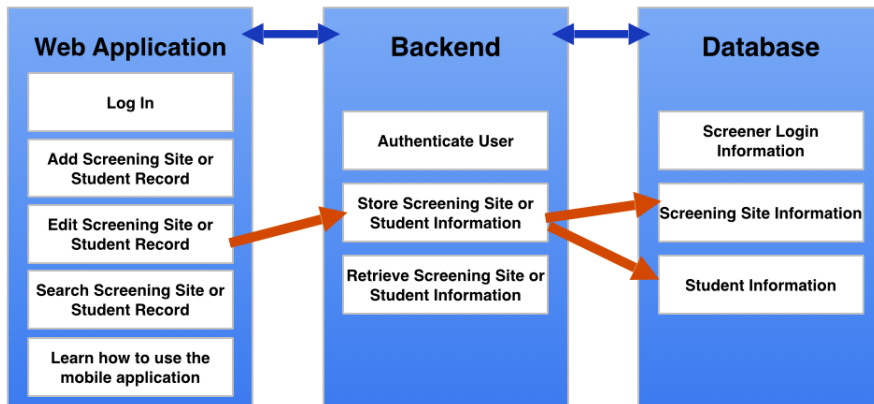

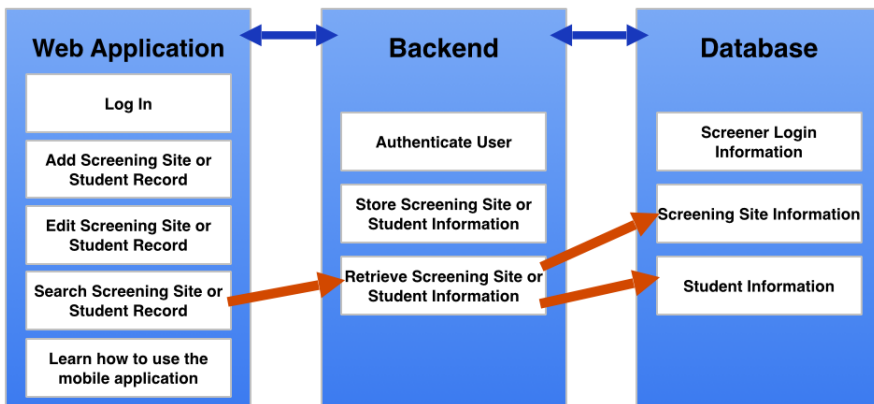Figure 4.15: System Diagram to Edit a Screening Site or Student Record - Store Newly Updated Record

Figure 4.16: System Diagram to Search a Screening Site or Student Record

4.3 Activity Diagram

The system begins when the user first opens the application. Once the application is opened, the user must select a language to use throughout the application. After selecting a language, the user must log in with their unique credentials. Once logged in, the user will enter the home page and have three options: select the how to use the application button, select one of the add buttons to add a screening site or student record, or select one of the search buttons to search and view a screening site or student record. If the user were to select how to use the application, they would be able to walk through a step-by-step tutorial on how to use the application. When they are finished, they would be able to go back to the previous page where they can again choose between how to use the application, add a screening site or student record, or search for a screening site or student record. If the screener was to select one of the add buttons, they will have the option to add a new Anganwadi site, school site, camp site, or student record. If the screener decided to add a new screening site record, they will fill out a form with the screening site name, date of screening, headmaster name, screening sight address, phone number, co sponsor name, co sponsor contact number, and person in charge contact number. If the screener decided to add a new student record, they will input the student's first and last name, age, gender, birthdate, grade, other basic information, and the student's eye health information. After adding a new record, whether it was a screening site or student record, the screener will be returned to the home page. The screener will then have the same three options as before: add a screening site or student record, search for a screening site or student record, or learn how to use the web application. If the screener were to select one of the search buttons, they will have the option to search for an Anganwadi site, school site, camp site, or student record. If the screener were to select the search a screening site or student record, he or she will have to type in the record name in the search field. The screener will be given a list of results to choose from and he or she can tap the record they wish to view. While viewing the record, he or she may choose to edit the record. If the screener is viewing a screening site record, he or she can view the site's student list. Once finished with their desired task, they would be able to go back to the home screen where they can log out or repeat the above process again. Figure 4.12 illustrates the workflow of our overall system.
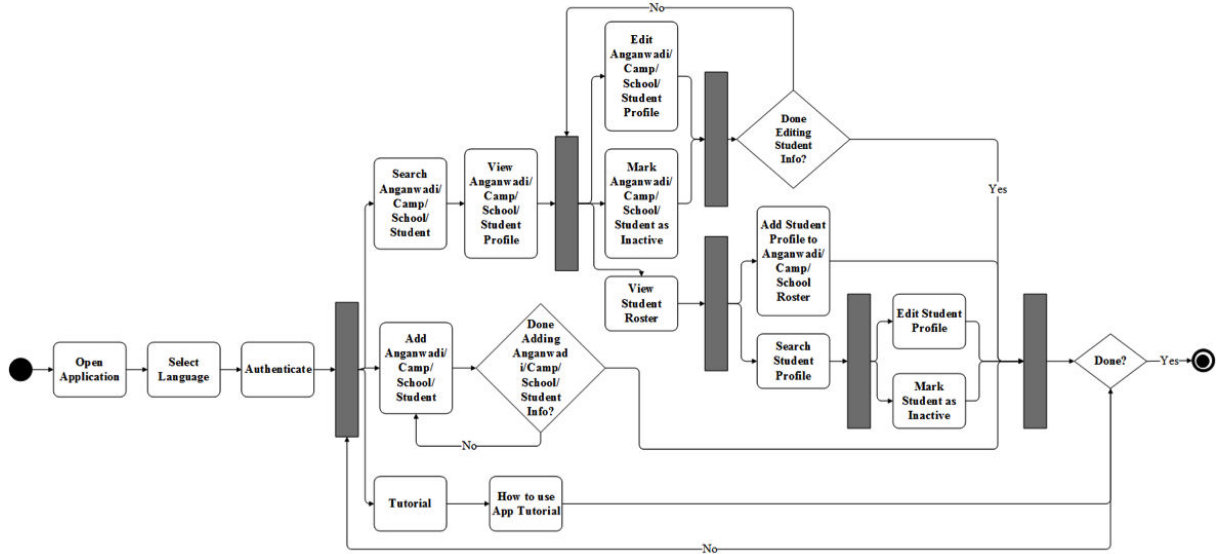
Figure 4.17: Activity Diagram

4.4 Technologies Used

- **HTML -** markup language

- **CSS -** style sheet language

- **JavaScript -** client-side scripting language for websites

- **Ionic Framework -** framework for creating hybrid mobile apps, a library of HTML, CSS, and JS components for building interactive apps, and optimized for Angular JS

- **Angular JS -** a JavaScript framework

- **Heroku -** cloud platform as a service (PaaS)

- **PostgreSQL -** object-relational database management system provided by Heroku

- **PHP -** server-side scripting language for web development

- **GitHub -** git repository service for revision control and source code management

4.5 Design Rationale

- **Search Suggestions -** We implemented search suggestions so that the user does not have to type out the full name of the student or of the screening site, but they could filter the information by the name, screening date, or role number. We did this so that the user can quickly and easily access the information.

20

- **Ionic Framework -** We decided to use the front end ionic framework which allowed us to stylize our application since we are not web design majors. This framework allows us to create a more user-friendly application that is both responsive and easy to use.

- **Heroku -** We decided to use Heroku, which is a cloud platform as a service, so that we could build and run our web application on the cloud. Initially, we decided to create a mobile application, but seeing as though a web application would be easier to deploy, we decided to switch from a mobile application to a web application. Heroku also allowed us to use the PostgreSQL database which is part of its platform and standalone service.

Chapter 5

Testing

Throughout our development process, we tested the functionality of our system to ensure that our major design choices are fulfilling our requirements and are being properly implemented. We conducted unit tests and integration tests throughout our design and implementation stages and conducted acceptance tests after we have completed most of our implementation.

5.1 Unit Testing

To ensure that our system has no bugs or flaws, we tested key AngularJS functions to make sure that each basic functionality was working properly throughout each stage of development. Major test cases include: storing and receiving to and from a database, authenticating a user prior to allowing them access to any information, and providing search suggestions when a screener is searching for a record.

5.2 Integration Testing

Major features and modules that needed to be integrated included database design, authentication, and search suggestions. After designing our database using PostgreSQL, we integrated the ability to enter information from the web application interface and storing the information into a PostgreSQL database using PHP. When we finished integrating PostgreSQL and our web application interface, we then integrated an authentication feature to the system so that only authenticated screeners will be able to have access to the mobile application. Lastly, we integrated search suggestions, where the screener is given a list of possible records as they are typing their query.

5.3   Acceptance Testing

To ensure that the system works properly and meets all the requirements, we performed acceptance tests with users that fully understand how our system is supposed to work and what is required of our system. Once we believe our system is operational, we conducted acceptance tests with our customer, Sankara Eye, to make sure that our web application meets all the functional requirements. We then asked for feedback regarding the aesthetics, usability, and functionality of our system. Acceptance testing continued until we have fully satisfied all the requirements and have developed the product envisioned by Sankara Eye.

Chapter 6

Project Management

6.1 Project Schedule

Our Gantt chart (Appendix A.1) depicts the development timeline of our project. We have three charts, one for each quarter, each of which is then divided into ten weeks. The Gantt Charts show which tasks each member is responsible for and when each task needs to be completed. For tasks related to the implementation of the project, we divided the tasks between the members. As for deliverable tasks, such as presentations and reports, we will work together as a team.

6.2 Risk Analysis

In addition to our Gantt charts, we also organized a table of possible risks (Appendix A.2). We included the consequences, probability, severity, and impact of those risks. For each risk, we provided a mitigation strategy in case the risk does occur during our project development, we will have a planned solution.

Chapter 7

Societal Issues

7.1   Ethical

Regarding the ethics of our application, we decided to adhere to the ACM Code of Ethics, specifically these four main points: strive to achieve the highest quality, effectiveness and dignity in both the process and products of professional work, respect the privacy of others, acquire and maintain professional competence, and articulate social responsibilities of members of an organizational unit and encourage full acceptance of those responsibilities. We decided to focus on these four main points because we believe that they encompass the main aspects of our application while taking ethics into account.

The first point is striving to achieve the highest quality, effectiveness and dignity in both the process and products of professional work. With this in mind, we understood that our application would be deployed in India, so we wanted to create an application to the best of our ability that would actually be used by Sankara Eye. We also wanted to create an application that met all their requirements, so we strived to achieve the highest quality of our application by keeping in constant communication with our customers.

The second point is respect the privacy of others. We understood that our application would handle sensitive data such as screening site and student information. With this in mind, we tried to create an application that was secure by having the users log in with valid credentials. We also decided to add the aspect of having a PIN that only Sankara members will know so that they can create an account.

The third point is acquire and maintain professional competence. We understood that when we would email Sankara or have to Skype with them, we would do so in a professional manner. We did this because we understood that they were our customers and should be treated as professionals.

The last point is articulate social responsibilities of members of an organizational unit and encourage

full acceptance of those responsibilities. Since this was a two- person team, we knew that we had to split up the work evenly and did so based on the list of requirements and based on our strengths.

7.2 Social

Since we are creating a product for a customer in a developing country, we must be culturally aware of our customer's concern and suggestions as well as the potential users of our product. In order to do so, we must meet the requirements that we agreed upon with Sankara Eye. We must also make sure our product is easy to use and affordable since we are working with a non-profit social organization.

7.3 Political

We believe that political consideration is out of scope for our web application. But we do realize that our product will impact the way Sankara Eye keeps track of student's eye health records, and as a result, students can get quicker treatment and maybe even more effective treatment since it will be easier for Sankara members to access and store records.

7.4 Health and Safety

Since Sankara Eye Mobile is a web application that keeps track of screening site and student records, our product must be secure in a sense so that screening site or student information does not get compromised. One way we made our application secure was by creating a Login page that will verify if the user is a Sankara member. In order to even create an account, the user must know the Sankara PIN' which only Sankara members will know. Although the school and student information is not too sensitive since the main information just consists of eye information, guardian names, and phone numbers, so security is not as big of a deal as other applications out there. When thinking about the health issues about our application, we realized that if the information is not stored correctly or input correctly, then it could lead to a wrong diagnosis or treatment of a student. This is why we tested to make sure the information was properly stored in our database and does not incorrectly manipulate the user's input.

7.5 Manufacturability

Our product is a frugal innovation project that will be deployed to our customers in India, Sankara Eye. We built a web application because it requires little to no installation for Sankara Eye members. They have been giving us suggestions and testing our project once in a while to make sure we were on the right track.

After creating the web application, we realized that maybe a different platform would maybe have allowed for less code and easier development. We first began by creating a mobile application using PhoneGap and the ionic framework, however, we realized that it would be easier to deploy the application if we made it as a web application. When we first started the project, we thought that we would not need any funds, but soon we realized that we would need to purchase a database that is hosted somewhere in the cloud. As of now, we are use a PostgreSQL database hosted on the Heroku server that is currently free but only hosts a limited amount of data. Once the application is deployed, they will need to upgrade to a database that contains more storage space and will cost about 10 USD.

Problems we see arising is the future would be when the Sankara Eye has to upgrade the database. Since we are passing on this project to fellows who will deploy our application in India, we hope that they will be able to upgrade the database without a problem and make the necessary changes which we will be including in our instruction manual.

7.6 Economic

We believe that economic considering is out of scope for our web application since the only cost that is necessary is the cost for the database, which ranges depending on how much storage space they want.

7.7 Sustainability, Environmental Impact

We believe that our web application is sustainable in a sense that our customers, Sankara Eye, can use our application for many years since it will store and keep track of student's eye health in India. By creating an application that can access student's eye health records, it caters to their needs since they wanted a convenient and easy way to track student's eye health records, and we believe our web application does that.

With the use of our web application, there is no longer a need to fill out paper forms for every child's

student record. Therefore, there are not any environmental issues related to our product.

7.8 Usability

Since we created our web application to be used by a social enterprise in a developing country, we knew we had to create a product that was easy to use by anyone with or without any prior technical experience. Our product is straightforward, easily learned, and easily used by the end user. Our web application is user-friendly because it only requires the user to simply input information into input fields and tap buttons to save and confirm.

7.9 Lifelong Learning

This project definitely prepared us for the time when we will have to learn on our own. We have had prior experience in web programming, but never had we had to build a web application. We had to refresh our knowledge of common web technologies as well as learn Angular JS and study which frameworks best fit our project needs. We also had to take into account that we are not just solving a given problem by our customer, we also have to consider the usability and accessibility. This project inspired us to study new material, because there are a lot more features we can add if we had more time to learn more different technologies applicable to our product.

7.10 Compassion

After meeting with Sankara Eye Representatives, we were able to learn about their mission to provide eyecare for everyone in India. Their compassion inspired us to create Sankara Eye Records to the best of our abilities. We hope with the use of the web application we can support Sankara Eye in reaching their goal by 2020.

Chapter 8

Conclusions

8.1 Summary

Sankara Eye Records is a web application aimed to help Sankara Eye keep better track of schoolchildren eye care records. The trained Sankara Eye screeners can create, edit, delete, or search for records of the sites and students they have screened. The screening site record also allows the screener to view the list of students that attend or live at that site. With the mobile application translated into eight different Indian languages, Sankara Eye can screen children and view their eye health quickly and efficiently.

8.2 Lessons Learned

Throughout the development of our project, we have learned the following lessons about managing a project and producing a product for an international outreach organization.

- We learned that our original plan in the fall quarter was too large to be accomplished fully by the end of spring quarter. We decided to scale down the project and complete the agreed upon requirements. Unfortunately, the extra features that we were eager to integrate into our system cannot be accomplished this year. Despite the lack of extra features, such as camera accessibility and voice recognition, our system is user-friendly and allows the screener to add, edit, or search for a screening site or student record. We can integrate the extra features in the second version of our system.

- We learned it is important to keep in constant contact and be in communication with our customer. In order to develop a product that our customer would be satisfied with, we learned it was important to verify any modifications or changes we make with them to guarantee a

correctly developed product.

8.3 Future Improvements

Features we would like to integrate into our system in the future would first allow the user to access the camera on their mobile devices to take profile pictures of the schoolchildren to add to their eye care record. The screener can also take pictures of the eye condition of the schoolchildren and add them to their eye care records. The second feature would be a reliable voice recognition API to allow the screener to quickly input information by speech rather than by keyboard. We would also like to find an API that works well with the eight Indian languages we have translated our mobile application into - not just the English language. Third, we would like to allow the screener to view statistics based on the information stored in the database. For example, the screener can view what is the most common eye illness among a certain gender or find out what the number of screening site have been screened in a certain region. Last, we would like to allow the user to access the database on a desktop-friendly website so they can perform more specific queries if needed.

Appendix A

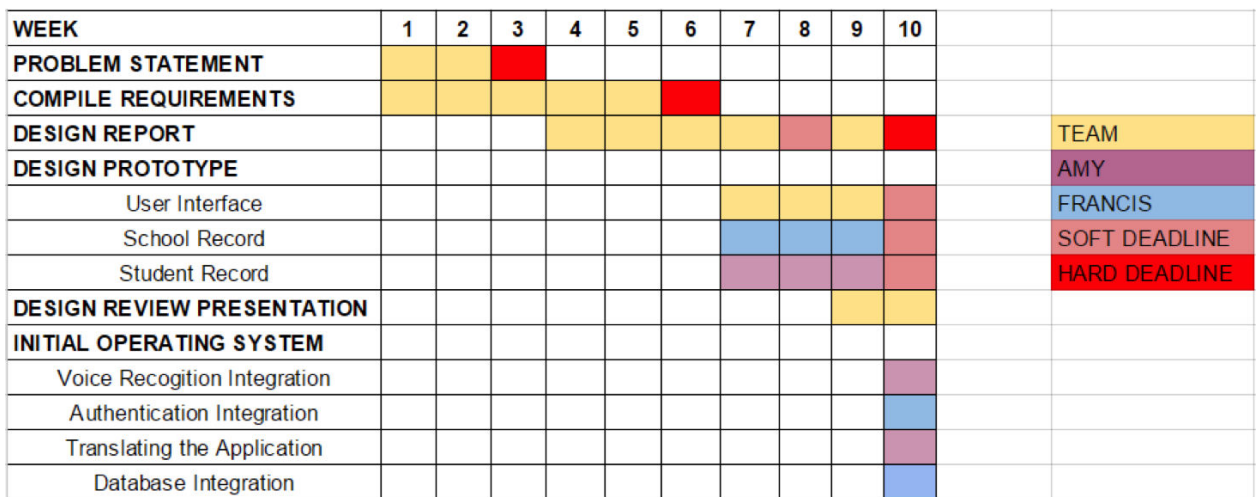# Project Management

A.1   Project Schedule

WEEK	1	2	3	4	5	6	7	8	9	10		
PROBLEM STATEMENT												
COMPILE REQUIREMENTS												
DESIGN REPORT												TEAM
DESIGN PROTOTYPE												AMY
User Interface												FRANCIS
School Record												SOFT DEADLINE
Student Record												HARD DEADLINE
DESIGN REVIEW PRESENTATION												
INITIAL OPERATING SYSTEM												
Voice Recogition Integration												
Authentication Integration												
Translating the Application												
Database Integration												

Figure A.1: Fall 2014 Quarter Gantt Chart

WEEK	1	2	3	4	5	6	7	8	9	10		
DESIGN REVIEW												
REVISED DESIGN REPORT												
OPERATIONAL SYSTEM												TEAM
Database Integration												AMY
Voice Recogition Integration												FRANCIS
Authentication Integration												SOFT DEADLINE
Translating the Application												HARD DEADLINE
TESTING												
Unit Testing												
Integration & System Testing												
Acceptance Testing												
Configuration Testing												
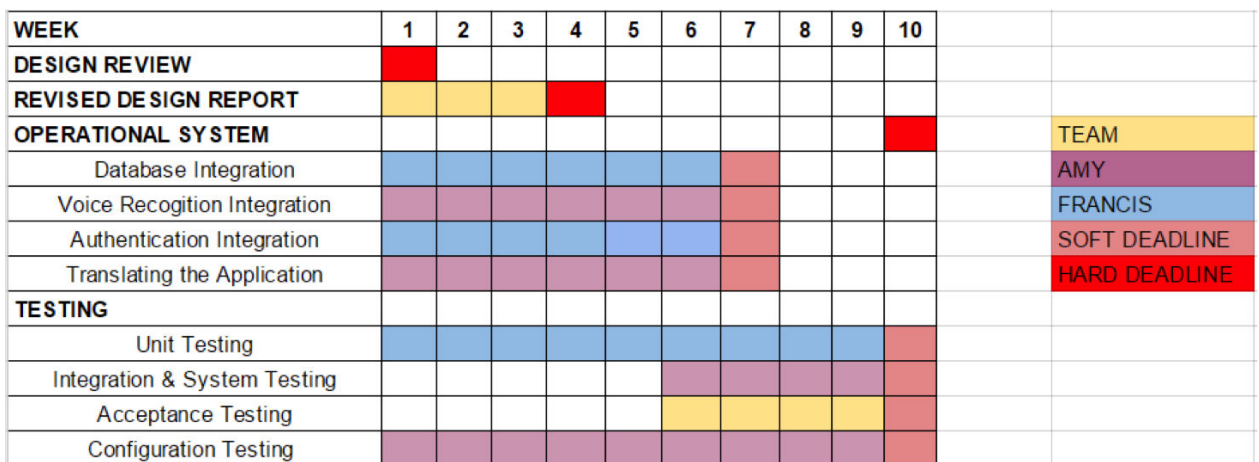
Figure A.2: Winter 2015 Quarter Gantt Chart

31

WEEK	1	2	3	4	5	6	7	8	9	10		TEAM
FINAL OPERATING SYSTEM												HARD DEADLINE
DESIGN CONFERENCE												
SENIOR THESIS												

Figure A.3: Spring 2015 Quarter Gantt Chart

A.2 Risk Analysis

Risk	Consequence	P	S	I	Mitigation Strategy
Incomplete/ Misunderstood Requirements	Fail to produce the product that the customer had envisioned	0.8	8	6.4	Periodically check with Sankara Eye representatives to confirm the developments of our project. Keep project requirements in mind throughout design process
Team members become busy or sick	Delay project completion and miss deadlines	0.5	8	4	Ensure both team members understand the project in order to take over the other member's incomplete tasks.
Data Loss	Lost work and delay project completion	0.1	10	1	Backup code and data periodically.
Problems with server	Delay project completion and miss deadlines	0.1	8	0.8	Initially test server with a small set of data prior to saving data to the database. Backup data periodically.

Table A.1: Risk Analysis Table

Appendix B

Setup and Installation

B.1   General

- **On your desktop or laptop browser**

  Simply enter the following URL in Firefox, Chrome, or Safari: https://sankara-eye-mobile.herokuapp.com

- **On your mobile device**

  iOS using Safari:

  1. Enter the URL in the browser: https://sankara-eye-mobile.herokuapp.com

  2. Click on the share button in the bottom of the screen as seen in Figure B.1 below:
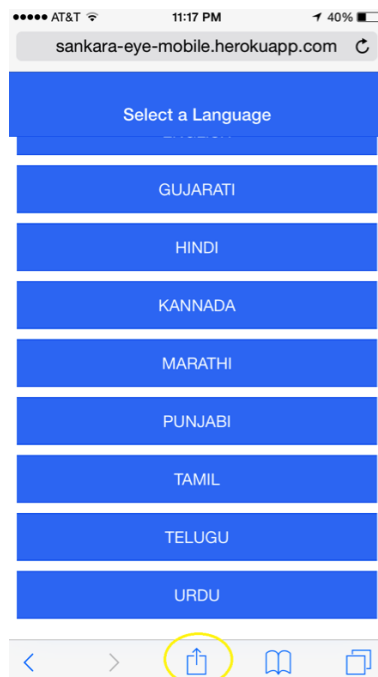
Figure B.1: Click the Share Button

3. Click on Add to Home Screen' as seen in Figure B.2 below:

Figure B.2: Add to Home Screen

4. Click on the Add' button on the top right corner

5. The web application is now added to your homescreen and can be easily accessed by clicking on the Sankara icon as seen in Figure B.3
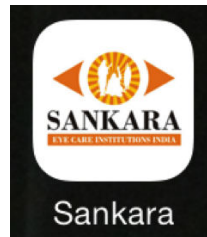
Figure B.3: Sankara Icon on iOS

• **Android using Google Chrome**

1. Enter the URL in the browser: https://sankara-eye-mobile.herokuapp.com

2. Click the button on the top right corner of browser and click on Add to Homescreen' as seen in Figure B.4 below
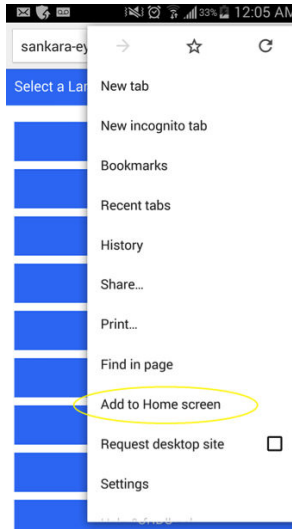
Figure B.4: Add to Homescreen

3. Then click on Add'

4. The web application is now added to your homescreen and can be easily accessed by clicking on the Sankara icon as in Figure B.5
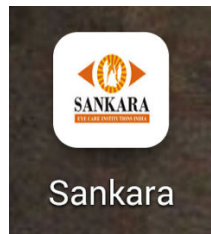

Figure B.5: Sankara Icon on Android

Appendix C

User Manual

C.1 General

To learn how to add/edit/search for screening site or student information, simply open the tutorial in our web application as seen in Figure C.1 below.
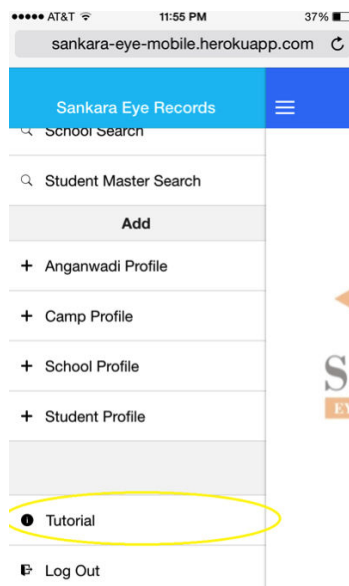

Figure C.1: Tutorial in the Side Menu

Upgrade the Database

Currently, the web application is using the Heroku Postgres Database, specifically the free version that only allows for 10,000 rows to be saved into the database. When more student information is added, the database size will need to increase. In order to upgrade the database, follow these steps:

1. Login to Heroku website: https://id.heroku.com/login

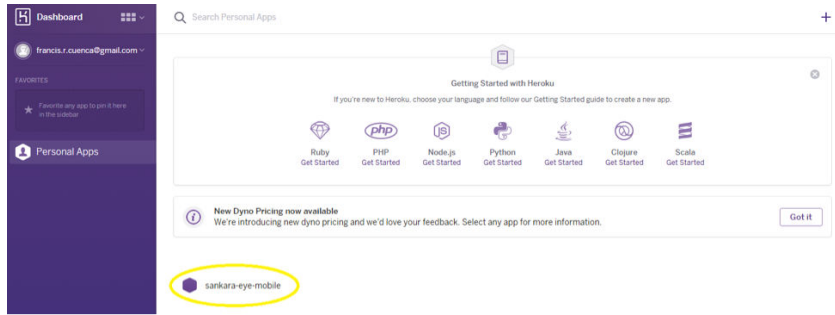2. Click on the sankara-eye-mobile app as seen in the Figure C.2 below

Figure C.2: Heroku homepage

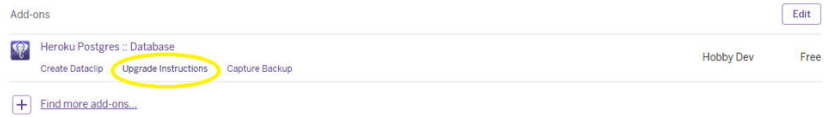3. Click on the upgrade instructions' button as seen in Figure C.3 below and follow the instructions

Figure C.3: Upgrade Instructions