

COMPUTER SCIENCE

APP DEVELOPMENT

OVERVIEW

Application development is responsible for the creation of various softwares relevant in an increasingly technologically demanding world. This section deals with a method in the use of Thunkable and Firebase in developing a dormitory pass application. Application development is a tedious process often requiring keen insight not only on the method of addressing an issue, but also the issue being addressed. Identify and focus on stakeholders prior to beginning the project.

A. [Thunkable and Firebase](#) (Roxas, Subong, Fuentes, Salazar, Briones)

The flowchart shows the steps in developing Fygo. These steps include the gathering of information from the manual leave pass system for developing Fygo, developing flowcharts that would determine the logic of Fygo, developing the subsystems, debugging each subsystem, integrating each subsystem, testing and gathering to the interns, dorm managers and security guards for the feedback.

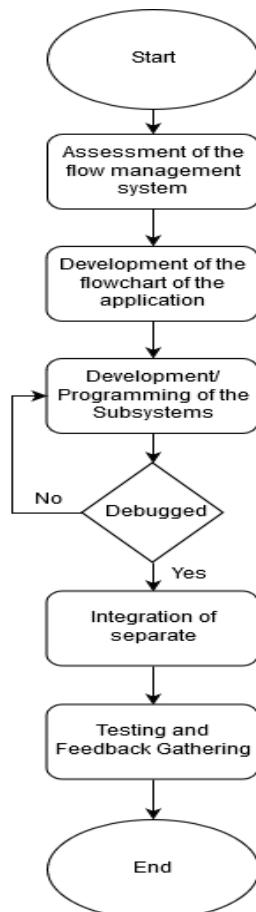


Figure 28. Flowchart of the timeline of the developmental phases of the application.

User interface (UI) is a vital concept when developing an application. The first impression of users is usually based on the UI of the application. If its UI is less attractive, users would tend to be bored in using the application. As stated by Nam (2011), the manager of a developed mobile application which is the interface of the system, UI is implemented for smartphone applications in the Android platform to support easy installation and usability with Graphic User Interface (GUI). Disregarding an application's function, its UI must follow a set of standards in order to be accessible. While doing work on Fygo proper, a user interface was designed so as to streamline development of what the front-end of the application would look like.

B. [Fygo](#) (Roxas, Subong, Fuentes, Salazar, Briones)

The development phase of Fygo is composed of 6 phases: assessment of the conventional method, development of application operation flowchart of Fygo, development of the Fygo subsystems, debugging, integration of subsystems, and testing and feedback gathering.

The assessment of the conventional method identified the different persons involved and their respective tasks for the current dormitory leave pass system. The application operation flowchart for Fygo was then modeled from this system. This flowchart served as a guide to develop the application. Another flowchart was designed to help visualize the final product of Fygo.

Separate subsystems were created for each type of user (interns, dorm managers, and security guards) to perform different, specific tasks. There are a total of 5 subsystems: logging in, registration, for interns, for dorm managers, and for security guards.



The login subsystem is the primary screen in which the users first encounter upon opening Fygo. On this subsystem, a user logs in using their username and password. The application will then identify if the user is registered and determine whether the user is an intern, a dorm manager, or a security guard then redirects the user to his/her respective subsystem. The login subsystem is an additional feature added to the system of Fygo.

New users who want to create a new account can proceed to the registration screen. The registration is only applicable to interns only. There is no need for security guards and the dorm manager to create a new account. Premade accounts have been created for the dorm manager and the security guards. The registration subsystem is also an additional feature added to the system of Fygo.

After the intern logs in, the application would redirect to the screen where he/she could file a leave pass. The subsystem designed for the interns are based on the paper-form leave pass that have been used in the current dormitory leave pass system.

When the dorm managers log in, Fygo would redirect to the screen consisting of three options: the list of requests, log records, and pending registrations. The list of requests screen contains a list of interns who filed for a leave pass. Log records contain the dates in which interns logged in and logged out. Lastly, the pending registrations screen contains a list of new interns registering for a new account.

When the security guards log in to Fygo, they would be able to see approved leave passes. They could also log in or log out the interns who have approved leave passes.

Each of the subsystems was debugged separately for easier spotting of any minor errors. After successfully debugging, the subsystems were integrated to become a functioning application. The screens for the different subsystems were also made and programmed. The application included a registration screen and a main menu or login screen.

The application was tested by 40 interns who were tasked to file a leave pass. The number of testers were determined by the Minimum Sample Size Needed for Interval Estimate of a Population Proportion. The average number of leave passes filed each weekend was used as the population size, and the confidence level and confidence interval was set to be 95% and 5, respectively. The application was also tested by a dorm manager who was tasked to approve the leave passes filed by the interns. Lastly, the application was also tested by a security guard who was tasked to log in and log out the approved leave passes. The time spent for each user to accomplish a task was recorded.

The ISO 9241-11 standard measure of usability has been used to evaluate Fygo. The application was evaluated using the effectiveness (completion rate), time-based efficiency, overall relative efficiency, and satisfaction of each user.



DESIGN

OVERVIEW

A proper workflow breaks down the key process/es an application undergoes to serve its function. This provides the programmer with workable goals that makes the application easier to code. This section provides the fundamental design for an assistive medical application. Program design guides the programming process eventually manifesting in the final product - allot sufficient time in designing the flowchart.

A. Application Structure

(Montero, Baldonado, Cruz)

Before the application could be programmed, its structure should be designed. The application structure should be designed based on the flow of events when using the application, as depicted in Figure 29.

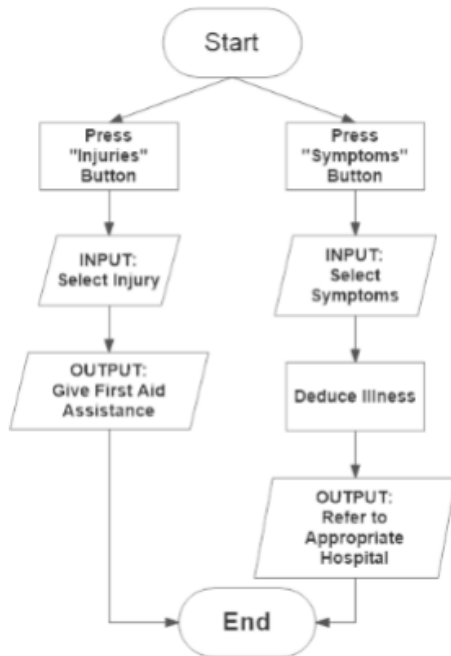


Figure 29. Flow of events in the mobile application.



IMAGE ACQUISITION AND ANALYSIS

OVERVIEW

Various functions such as acquisition, analysis, and comparison of data can be automated through the use of modern technology. This section provides the parameters in the acquisition of fish-eye images as well as introduces the software, tools, and processes to analyze them, particularly as they relate to color, value, and saturation. This may be applicable in similar studies involving image acquisition and analysis. A controlled environment, i.e. lighting, angle, etc., is crucial for research studies using images as it establishes the validity of the analysis and/or comparison. In image analyses, each software has specifications that make them ideal for certain uses, i.e. in the conduct of research; hence, the pros and cons of each software must be considered prior to its inclusion in the method.

A. Fish Images Acquisition

(Cadorna, Chan, Salmon, Salazar)

A color camera and an illumination chamber or lightbox as described by Dowlati et al. (2003) were used in an image acquisition system to capture the images of fish. Two LED lamps (Natural Daylight, 240 V / 4W) with a color temperature of 6500 K were used to capture high quality images under reproducible lighting conditions. The lamps with lengths of 30 cm were installed opposite of each other on the left and right sides of a wooden lightbox of dimensions of 50 x 50 x 30 cm. The interior walls were painted matte black to minimize background light and light reflectance. The sample tray was green in color for adequate contrast with the fish sample (Dowlati et al. 2013). The images of fish eyes were captured using a digital color camera (Nikon AF-S DX NIKKOR 18-55mm 1:3.5-5.6G VR II). Top-view images were taken at a vertical distance of 30 cm (Lehnert et al. 2011). All images were taken using constant camera settings (see Table 6). The LED lamps were turned on an hour before image capturing to ensure stable lighting (Dowlati et al. 2013).

Table 6. Camera control settings.

Variable	Settings
Image size	6000 x 4000 pixels
Zoom	No zoom
Flash mode	No flash
Sensitivity	ISO-200
Operation mode	Manual
Aperture Av.	f/6.3
Exposure time Av.	1/15 s
Image type	JPEG
Macro	On
Focal Length	18mm

B. Fish Image Analysis

(Cadorna, Chan, Salmon, Salazar)

Segmentation is the process of partitioning an image into regions that correlate strongly to features of interest in an image. The regions of interest, the eye and the pupil, were manually segmented using GNU Image Manipulation Program 2.10.8 to isolate them for color analysis (Lehnert et al. 2011). The elliptical marquee tool was used to select and segment images of the eye which includes the pupil and iris, and the pupil which appears as a black circle in the middle of the eye (Kanamori et al. 2017). The RGB values of the processed images were converted to their respective HSV equivalents. From the HSV matrices, only the saturation and value layers were used for analysis. All empty pixels of the image were removed before calculating for the mean saturation and value of the respective layers. Image analysis was done using Scilab 6.0.2 (Sengar et al. 2017).



IMAGE PROCESSING

OVERVIEW

Analyzing images can be made easier through the use of softwares and applications. The methods in this section provide a means of automating tasks using the software ImageJ. Each software has specifications that make them ideal for certain uses, i.e. in the conduct of research; hence, the pros and cons of each software must be considered prior to its inclusion in the method.

A. Fractal Dimension

(Aguirre, Sombiro, Valdestamon)

All 300 x 210 px photographs of the chorioallantoic membrane were analyzed using the software ImageJ. The blood vessels on the image were first reinforced by being manually overlaid with black using any image-editing software to have a strong contrast against the background.

These reinforced images were then converted to 8-bit grayscale in ImageJ and the range of dark colors that represent the blood vessels were isolated from the background. The blood vessels were skeletonized and the fractal dimensions were retrieved using the box-counting method set to a maximum of 64 boxes on ImageJ.



PROCUREMENT AND PROGRAM IMPLEMENTATION

OVERVIEW

Each hardware and database has different functions and efficiency in terms of their performance; thus, before any implementation is done, it is necessary to have the appropriate hardware and database for the conduct of the study. Making or modifying a program is also needed to determine a particular thing depending on the objective of the study. This section focuses on the procurement of hardware and database and the use of an Apriori algorithm that has been modified to provide the necessary details needed for the study. In this case, it aimed to output the execution time of the algorithm and the frequency of the most frequent itemsets.

A. [Hardware and Database Procurement](#)

(Celestial, Ibarreta, Tiron, Nulla, Oberio)

An ASUS Vivobook X442U laptop which had an Intel Core i5 3.4GHz Processor with 4GB RAM was used in the testing process. A grocery database having 9,835 transaction records in total was acquired from Salem Marafi (Salem 2014). It contained a collection of receipts with each line representing one (1) receipt and the items purchased. The database was then converted from a comma-separated values (.csv) file to a text (.txt) file by replacing the commas with spaces, since the source code used required plain text for its input.

B. [Algorithm Implementation](#)

(Celestial, Ibarreta, Tiron, Nulla, Oberio)

The Java source code of Apriori was acquired from Github (Umanghome 2016) and was modified to output the execution time of the algorithm and the frequency of the most frequent itemsets. The modified source code was then checked by two (2) consultants, Mr. Christian Chiu, the owner of SoftArtifact Incorporated, a product development company based in Koronadal City, South Cotabato, and Mr. Marc San Pedro, an engineer at Samsung Electronics.

