DUBLIN CITY UNIVERSITY

Raspberry CCTV

Functional Specification

BY

Ian Dowling 12714619

Fourth Year Project

Computer Applications

School of Computing

27/11/2015

Student Blog*:* *http://blogs.computing.dcu.ie/wordpress/iandowling2/*

Table of Contents

1. Introduction

1.1 Overview

Raspberry CCTV is primarily a security device for entrance or exit passage ways. However, this technology could easily be deployed in many different environments and scenarios such as a baby monitoring system. Users of the system will be able to set up the Raspberry CCTV Camera over a doorway and monitor the activity of the doorway remotely on a mobile device, through the Raspberry CCTV application. The Raspberry CCTV Camera will have motion detection build in so when the system detects movement in front of the camera it will start recording video. It will also send a push notification to the mobile device so the user will be able to check who is at the doorway. Through the mobile application interface the user will be able to record images and video, along with text-to-speech functionality. This feature will allow the user to write a warning to the would be intruder or friend in the doorway and through a speaker connected to the Raspberry CCTV Camera the text will be converted to speech.

1.2 Business Context

Raspberry CCTV has not been sponsored by or is it being developed for a specific company. However, it is a product designed to improve the security and well-being of it's users. Home security is an increasing market growing each year with the increase of available technology products designed for this purpose. However, these products are generally still quite expensive with many people not being able to afford this kind of technology. The Raspberry CCTV Camera is being developed to make this technology more widely available to the average person. It is being developed as a cheaper alternative to the likes of Belkin's Netcam, Nest Cam and the latest version of Eircom's Phone Watch. All of which are very expensive to buy and install, with Eircom and Nest Cam having recurring monthly charges. Whilst, these are very good products I still believe that there is room for a more affordable option which has the ability to have a greater market share than these three other options.

1.3 Glossary

PHP

PHP - Hypertext Pre-processor is an open source server-side scripting language, that is embedded into HTML and used to create dynamic webpages. It is also used to relay information to a database which operates in the background of a website, e.g. when a user logs in to a site or buys something online.

Cross - Site Scripting

Cross-site scripting is a type of computer security vulnerability typically found in Web applications. XSS enables attackers to inject code into web pages viewed by others in order to access sensitive data stored by the website.

SSADM

Structured **S**ystems **A**nalysis and **D**esign **M**ethod is a systems approach to the analysis and design of information systems.

OpenCV

OpenCV is a library of programming functions used for real-time computer vision projects.

HoG Algorithm

The **H**istogram of **O**riented **G**radients is a technique used in computer vision and image processing for the purpose of object detection.

AWS EC2 – Amazon Web Services

Amazon **W**eb **S**ervices **E**lastic **C**ompute **Cl**oud is a web service that provides resizable compute capacity in the cloud.

2. General Description

2.1 Product/System Functions

The Raspberry CCTV system consists of multiple parts, a Raspberry Pi, a USB webcam, a USB Wi-Fi receiver, a wireless Bluetooth speaker, a 32GB MicroSD card and an Android application. The USB webcam, Wi-Fi receiver and speaker will be connected physically to the Raspberry Pi. The Raspberry Pi will run a version of the Raspbian OS which will be mounted on the MicroSD card. The system can be powered using a 5V Mini-USB power supply or battery power pack used for large tablets.

I plan on having the Raspberry Pi itself contain the server software to host the software needed to for the system to work as it would be a more enclosed system without other external entities. However, this may not be possible due to the Raspberry Pi's limited hardware resources. If this is the case, I hope to use a cloud server instead to help provide the processing power needed. Once the server is up and running, the user will first be greeted to a log in page. After the correct user details are submitted and the user is authenticated they will then be able to control the webcam. The image and video files will be stored on the device with the ability to add additional storage in the form of a USB memory stick.

The user will be able to connect to the webcam from their mobile device running a version of Android, Ice Cream Sandwich or later. Through the application the user will be able to take images, video content of an intruder and use text-to-speech software to warn off intruders. The application will be downloadable from the Android Play Store free of charge.

The Raspberry Pi along with the multiple hardware connections will be enclosed in a dummy CCTV camera shell in order to keep all the parts protected from the environment and other third parties. The outer casing can be installed by a bracket to a wall or ceiling.

2.2 User characteristics and Objectives

There is not much requirements in the way of technical knowledge, for the user to be able to use the system. The system will require the basic knowledge of Android, i.e. to be able to download the Raspberry CCTV application. The application itself will be very user friendly, increasing it's usability for all users. The user will be able to do all the setup from the application and therefore will not need the use of a laptop. All the user will need to know in terms of technical knowledge is how to mount the device to the purposed surface and be able to turn it on. Once the Raspberry CCTV Camera is turned on the server will automatically start up.

The objectives of the device will be to provide the user with multiple functionality when it comes to streaming video content to their mobile device. The application will have the ability to take and record images as well as being able to change different viewing options such as changing recording resolution and frame size. The Raspberry CCTV application will also cater for additional cameras. If the user adds on additional cameras the application will be able to display the extra cameras in the appropriate manner.

2.3 Operational Scenarios

2.3.1 User Login

In order to gain access to the Raspberry CCTV Camera, the user needs to enter login details. When the application is launched the user will be directed to a screen which contains two input fields, one for a username, the other for a password. If the user's details are authenticated correctly, the user will be granted access to the system. If the user is not authenticated, an on-screen message will be displayed.

2.3.2 Recording an Image/Video

Once authenticated, the user will be able to view a live stream from the Raspberry CCTV Camera. This page will also contain several buttons. Two of the buttons will allow the user to take a picture or video content from the camera. Once the camera button is pressed, a jpeg image file will be saved to a storage device connected to the Raspberry Pi. When the video button is pressed, an avi file will be also saved to the storage device connected to the Raspberry Pi. If the storage device has reached it's capacity, a warning message will be displayed to the user. It will inform the user that they must delete some files in order for more files to be recorded.

2.3.3 Changing User Password

Users will be able to change the default password of their Raspberry CCTV system. To do this, the user will navigate to the settings menu, denoted by the settings icon in the top right corner of the screen. This will activate a dropdown menu which will contain the page user accounts. On opening of the user accounts page they will see two input fields, new password and confirm password. The user will then need to enter their new password into both fields and press the confirm button. If the passwords both match the password will be changed. If the passwords don't match, then an on-screen error message will be shown.

2.3.4 Text-To-Speech

Users will be able to warn off would be intruders using text-to-speech software. To use this functionality the user will enter text into an input field at the bottom of the main page. On pressing the send button, the Raspberry CCTV system will speak the words through a speaker connected to the system. This will only be a one-way form of communication as there will not be a microphone to pick up any comments made by the possible intruder.

2.3.5 Detection of Movement

The system will use the HoG Descriptor Library within OpenCV to detect movement within the video frame. It compares small connected regions called cells and then creates a histogram of gradient directions from the pixels in each cell.

2.3.6 Notification of Movement

When movement is detected by the camera, an Android Push Notification will be sent via the application to the users phone. This will tell the user that an object has been detected by the system and to view the real time video stream. If the user is happy that it is not an intruder, they will be able to mute notifications for a set period of time to avoid constant spamming of notifications.

2.4 User Constraints

The following are constraints with which I will have to deal with during the construction of this product:

- Time Constraints – I will have to meet deadlines throughout the entire build including the final deadline of Monday 23 May, 2016.
- Hardware performance – By using a Raspberry Pi I will have only got so much resources to work with. The Raspberry Pi is not as powerful as a regular computer and therefore will perform slower than what some people may be used to.
- Budget constraints – because I'm a student and do not have an unlimited budget, the webcam being used is a basic webcam. It performs at a lower resolution than some top of the range webcams and it also does not have a night mode.

3. Functional Requirements

3.1 User Login

Description

The system will require the user to login in order for them to use the system. The login will be a username and password. On first use the user will be supplied with a default username and password but this should be changed immediately.

Criticality

The user will click on the login fields and fill in their login details. There will be an two input fields in the centre of the screen, which will represent the username and password text fields. On inputting the correct login details, the user will be directed to the main system page which will consist of a video stream from the Raspberry CCTV Camera and the input controls. There will also be a header at the top of the screen. This will include the system title, a gallery icon indicting access to the file system and a settings icon.

Technical Issues

The main issue is to make sure that every user has a different username and password and that the login system will only work when the correct login information is inputted. Another issue would be that we must make sure that the input fields protect against cross-site scripting attacks and that all passwords are correctly encrypted to prevent third party access.

Dependencies with Other Requirements

Every other requirement will have a high dependency on the User Login requirement. If the user is unable to log into the system, they will not be able to use any of the other requirements.

3.2 Recording an Image/Video

Description

When a user receives a notification informing them of activity in their security zone, they can choose to take additional images and video content from inside the application. The content recorded will be saved on a external USB memory stick connected to the Raspberry Pi for later viewing.

Criticality

Users will be able to do this by two simple buttons underneath the video stream that will display what the webcam is showing. The buttons will be identified by an image icon and a video icon. The same images a user would find on their Android smartphone.

Technical Issues

A technical issue which might arise here, is that the storage space may run out of the system will have to be able to tell if the storage device is full and to give an option on what to do in this scenario.

Dependencies with Other Requirements

This requirement depends on the User Login requirement being met in order for the user to be able to use this functionality.

3.3 Accessing Media Archive

Description

Users will need to be able to view the content recorded by the Raspberry CCTV Camera through the application. They will also need the ability to delete files from this location in case the storage device is full. A user will be able to access the media archive through the application by the use of the gallery icon at the top header of the screen. This icon will be located beside the settings icon.

Criticality

When a user presses the gallery icon at the top of the screen, they will be directed to the media archive of the device. From here the user will be able to delete files in order to free up some space on the storage device.

Technical Issues

The main issue here is figuring out how to tell the device that the external storage device is full or near full and to send a notification to the user. The user then must decide whether to delete files or not to record anymore content until another storage device with free space is inserted into the Raspberry Pi.

Dependencies with Other Requirements

This requirement depends on the User Login requirement being met in order for the user to be able to use this functionality.

3.4 Text-To-Speech

Description

Users will have the ability to enter text into a text field. The text entered will then be read out by the speaker connected to the Raspberry Pi to warn off any would be intruders.

Criticality

There will be a text field at the bottom of the screen to handle this functionality. Using the IBM Bluemix text-to-speech API the text entered into the text field will be converted into audio and played through the speaker connected to the Raspberry Pi.

Technical Issues

The main issue here is if the Bluetooth connection which connects the speaker to the Raspberry Pi drops out then the text will not be read out and the user will not be able to reconfigure the speaker remotely.

Dependencies with Other Requirements

This requirement depends on the User Login requirement being met in order for the user to be able to use this functionality.

3.5 Detection of Movement

Description

The system will detect movement of people in real time. The system will use OpenCV and the HoG Algorithm to detected movement of people within the video frame.

Criticality

A green box will appear around the subject showing that movement has been detected. This feature will be done automatically and there will be no on/off button for this functionality.

Technical Issues

The main issue with this will be the limitations of the Raspberry Pi hardware itself to power this functionality. I may need to move the server to a cloud server in order for this functionality to work.

Dependencies with Other Requirements

This requirement depends on the User Login requirement being met in order for the user to be able to use this functionality and the hardware capabilities of the Raspberry Pi itself.

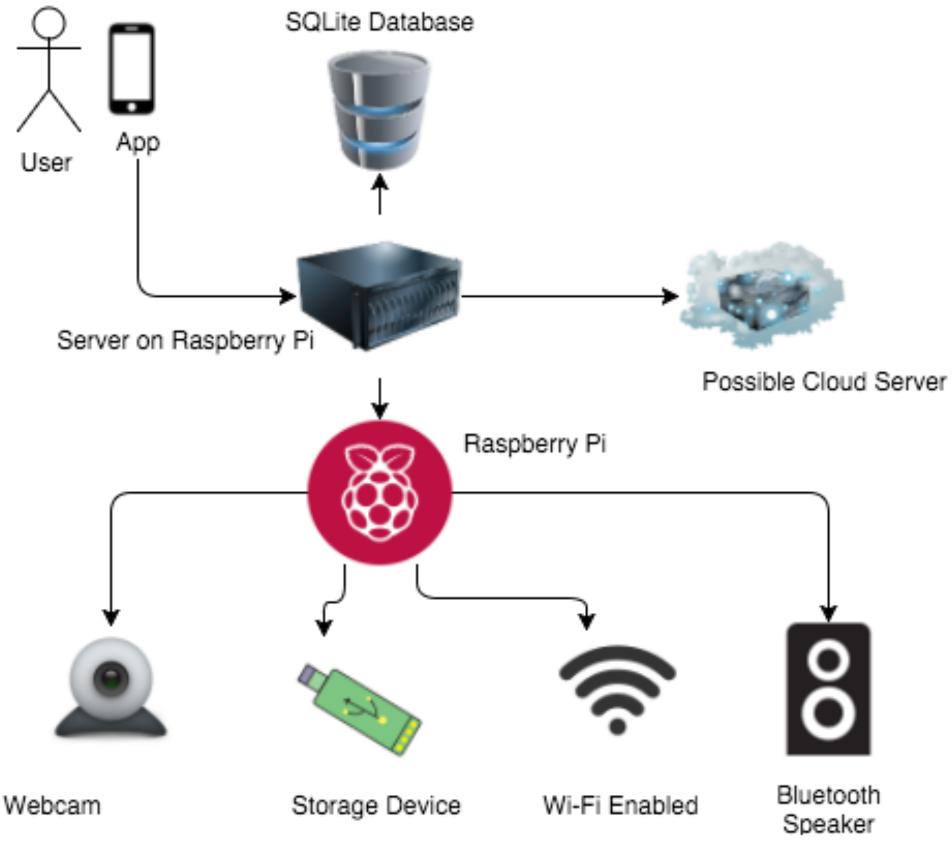
4. System Architecture

I plan to host the software along with my database on the Raspberry Pi itself using Apache Tomcat. This feature will allow the Raspberry CCTV system to be self-dependent, removing the need for more interactions and ultimately improving the user experience.

Third party tools which will be used are:

- Raspberry Pi 2 Model B 1GB RAM
- Logitech Webcam c170
- Bluetooth Speaker
- D-Link N300 Wi-Fi Adapter
- Apache Tomcat
- OpenCV
- AWS EC2
- Android Studio

4.1 System Architecture Model

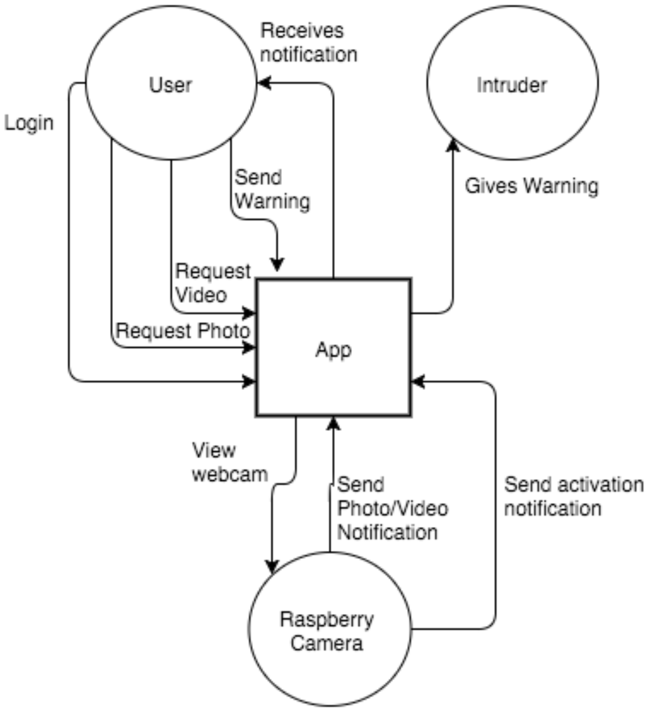


5. High Level Analysis Models

We use SSADM techniques to identify how the Raspberry CCTV system will flow and the interaction between the different entities and the system. This will give us a better understanding of how the different users will interact with the system.

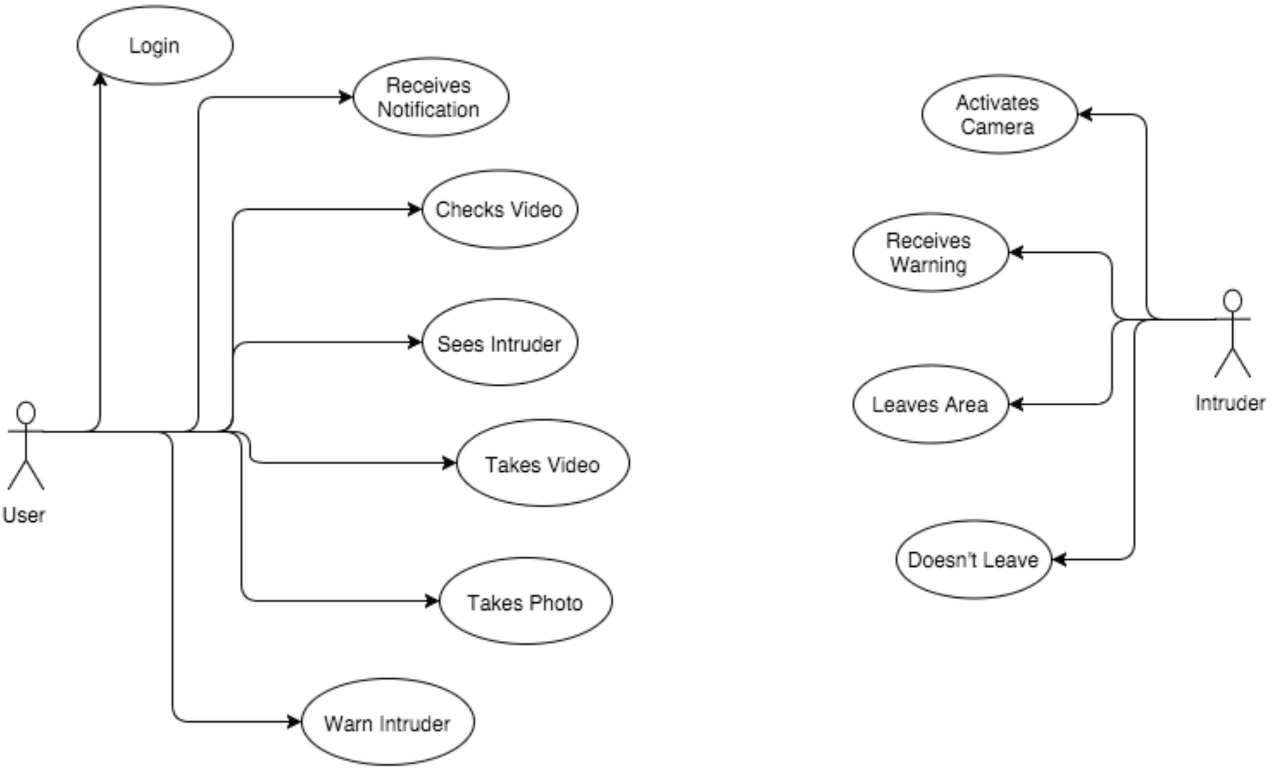
5.1 Context Diagram

This diagram shows how the system interacts with its external entities and the Raspberry CCTV system, comprising of a user(s), intruder(s) and the application. It shows the flow of information between the system and these entities.



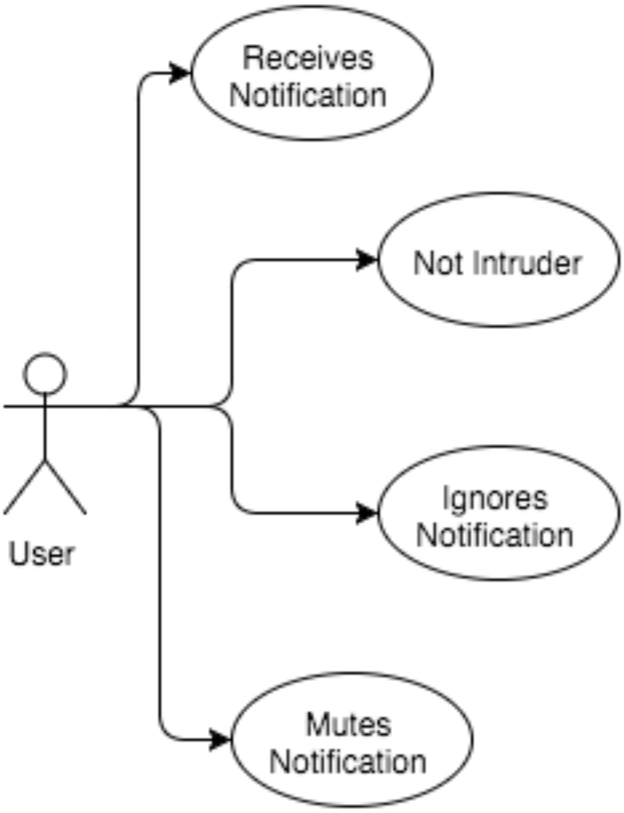
5.2 Use Case Diagram I

This model identifies the different interactions between the users of the system and the system itself. The user's are referred to as actors. In this diagram, we can see the different roles being played out. The scenario below describes if an intruder is indeed present.



5.3 Use Case Diagram II

This model identifies an alternate scenario which may play out for the user of the system, whereby an intruder is not present.



6. Preliminary Schedule

6.1 Schedule Overview

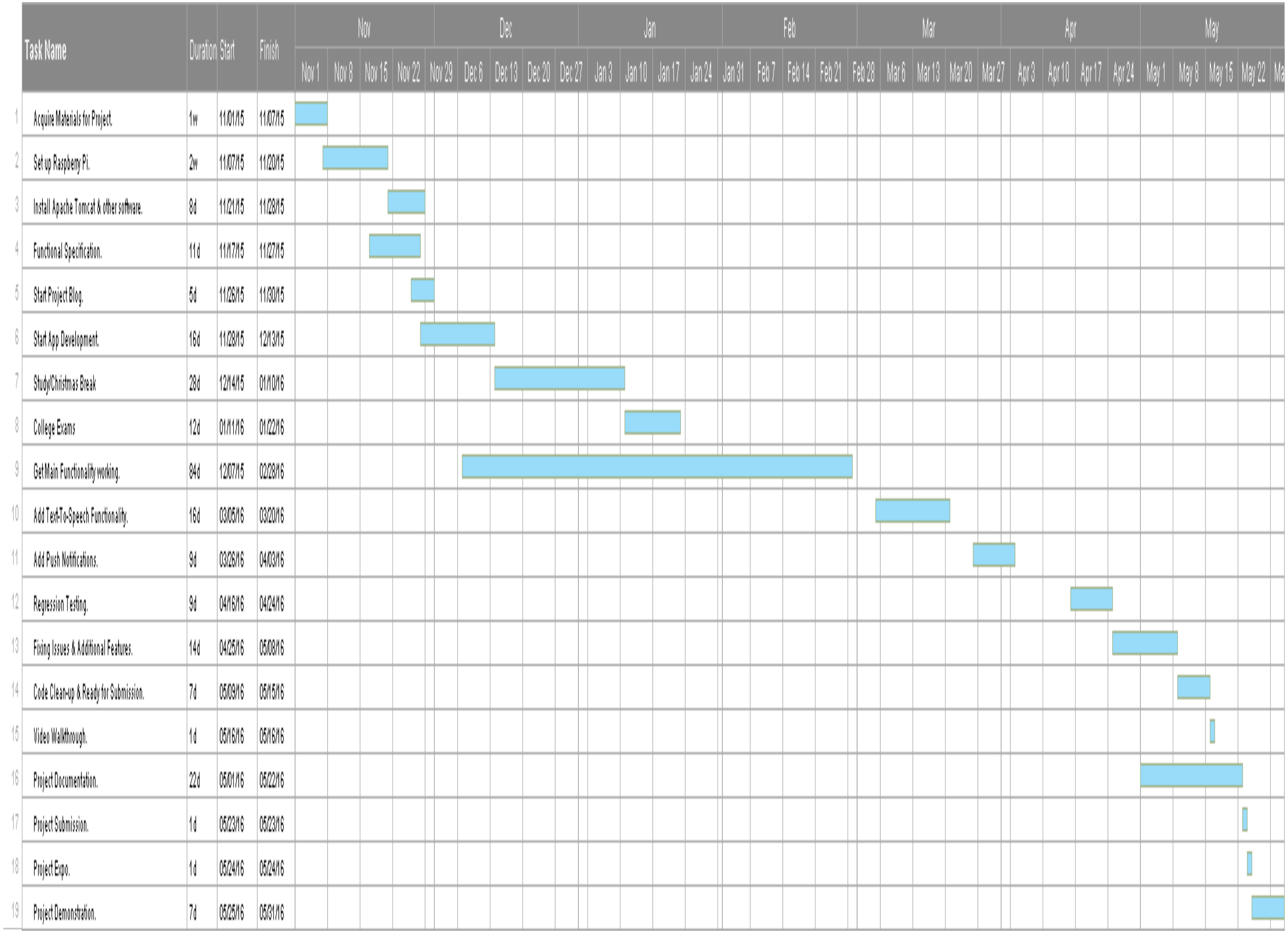
The following is my schedule to completion. It specifies exactly what work needs to be done and when in order to meet all my targets for a successful project. The major tasks are as follows:

- Set up the Raspberry Pi with Webcam server and Wi-Fi adapter all working. The user should be able to view the webcam using the Apache Tomcat on the Raspberry Pi. Start November 7, 2015. End November 20, 2015.
- Install Apache Tomcat and SQLite database of the Raspberry Pi. Start November 21, 2015. End November 28, 2015.
- Complete Functional Specification. Start November 17, 2015. End November 27, 2015.
- Create and start blog to track my progress of this project. Start November 26, 2016. End November 30, 2016.
- Start development of the application. Create the login page and framework for the main page. Start November 28, 2015. End December 7, 2015.
- Study break for semester 1 college exams. Start December 14, 2015. End January 10, 2016.
- Semester 1 college exams. Start January 11, 2016. End January 22, 2016.
- Connect the application to the Raspberry Pi server. End February 14, 2016.
- Get the main functionality all working. This includes connecting the server with the client, the video stream, the photo and video buttons. Start December 7, 2015. End February 28, 2016.
- Add Text-To-Speech functionality to the application using Bluemix. Start March 5, 2016. End March 20, 2016.
- Add push notifications to the application. Start March 26, 2016. End April 3, 2016.
- Start regression testing main functionality of the system. Start April 16, 2016. End April 24, 2016.

- Fix any issue resulting from testing. Complete any additional features implemented in the system. Start April 25, 2016. End May 8, 2016.
- Code clean-up, comment and javadocs. Code ready to be submitted. Start May 9, 2016. End May 15, 2016.
- Create the video walkthrough needed for final submission. Start May 16, 2016. End May 16, 2016.
- Create project documentation needed for final submission. Start May 1, 2016. End May 22, 2016.
- Project submission. May 23, 2016.
- Final Year Project Expo. May 24, 2016.
- Project Demonstrations. Start May 25, 2016. End May 31, 2016.

6.2 Gantt Diagram

This diagram shows the perspective timeline of the project. The project is broken down into sub-tasks with each sub-task being given a specific deadline. This gives me a specific breakdown of what I need to achieve by what time frame in order to get the project done successfully. Diagram on the next page.



	Task Name	Duration	Start	Finish
1	Acquire Materials for Project.	1w	11/01/15	11/07/15
2	Set up Raspberry Pi.	2w	11/07/15	11/20/15
3	Install Apache Tomcat & other software.	8d	11/21/15	11/28/15
4	Functional Specification.	11d	11/17/15	11/27/15
5	Start Project Blog.	5d	11/26/15	11/30/15
6	Start App Development.	16d	11/28/15	12/13/15
7	Study/Christmas Break	28d	12/14/15	01/10/16
8	College Exams	12d	01/11/16	01/22/16
9	Get Main Functionality working.	84d	12/07/15	02/28/16
10	Add Text-To-Speech Functionality.	16d	03/05/16	03/20/16
11	Add Push Notifications.	9d	03/26/16	04/03/16
12	Regression Testing.	9d	04/16/16	04/24/16
13	Fixing Issues & Additional Features.	14d	04/25/16	05/08/16
14	Code Clean-up & Ready for Submission.	7d	05/09/16	05/15/16
15	Video Walkthrough.	1d	05/16/16	05/16/16
16	Project Documentation.	22d	05/01/16	05/22/16
17	Project Submission.	1d	05/23/16	05/23/16
18	Project Expo.	1d	05/24/16	05/24/16
19	Project Demonstration.	7d	05/25/16	05/31/16