Software Requirements Specification

for

Motion Detector

**Version 1.0**

**Prepared by**

**Group Name:**

|  |  |  |
| --- | --- | --- |
| **Archit Puri** | **101603050** | **apuri\_be16@thapar.edu** |
| **Arham Bansal** | **101603051** | **abansal1\_be16@thapar.edu** |
| **Arpit Agarwal** | **101603054** | **aagarwal\_be16@thapar.edu** |
| **Arjun Gupta** | **101611008** | **agupta14\_be16@thapar.edu** |
|  |  |  |

|  |  |
| --- | --- |
| **Instructor:** | **Dr. Ashima singh** |
| **Course:** | **Software Engineering** |
| **Lab Section:** | **COE4** |
| **Teaching Assistant:** |  |
| **Date:** | **11th October, 2018** |

**Contents**

**Contents**

**Revisions 3**

**1** **Introduction 4**

1.1 Document Purpose 4

1.2 Product Scope 4

1.3 Intended Audience and Document Overview 5

1.4 Definitions, Acronyms and Abbreviations 5

1.5 Document Conventions 6

1.6 References and Acknowledgments 6

**2** **Overall Description 6**

2.1 Product Overview 6

2.2 Product Functionality 7

2.3 Design and Implementation Constraints 8

2.4 Assumptions and Dependencies 8

**3** **Specific Requirements 9**

3.1 External Interface Requirements 9

3.2 Functional Requirements 13

3.3 Use Case Model 16

**4** **Other Non-functional Requirements 19**

4.1 Performance Requirements 19

4.2 Safety and Security Requirements 19

4.3 Software Quality Attributes 20

**5** **Other Requirements 20**

**Revisions**

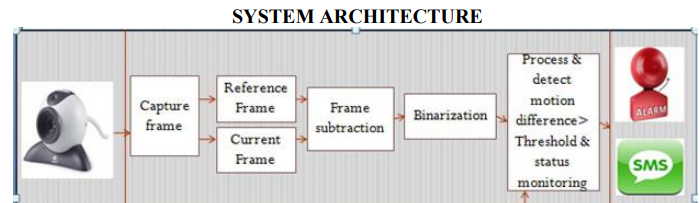
|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| Draft Type and Number | Full Name | Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being upgraded. | 00/00/00 |

# Introduction

The current security system, specifically, the well-known CCTV, consumes a lot of resources such as memory, due to nonstop recording. Verily, they are efficient but it takes a while before one gets back to locate the precise time where an event happened in the area under surveillance. One has to rewind and fast forward, going back and forth to search a particular scene and that takes a lot of time and effort.Furthermore, time is needed to keep watch on the activities going on via the screen. Something may be happening but due to negligence and human errors it may pass by without been noticed, until something happens. Then the search will begin without any idea of where to start searching with lots of videos to go through. As such, much attention and concentration is required to avoid missing important and significant activities. Our idea is to develop a System to detect the human motion and give sms alert at the same time. We are developing this idea because earlier methodologies are not so accurate and expensive also. As well as previous techniques only give alert but do not send the sms to the authorized person.

## Document Purpose

## Security is one of the most important problems all over the world. This project basically concern on the use of automatic motion detection application using webcams for security purpose. Today in our society security is one of the major issues and having a 24\*7 human eye is just impossible. Our project Motion Detection Application Using Web Camera is just a one of the applications which help us to achieve this goal. The prime motivation for developing this project is that, earlier methodologies only give alert but do not send the sms to the authorized person.



## Product Scope

In order to be secured of safety, it is useful to realize and manage smart surveillance system combined with image processing techniques. These functions are necessary for autonomic monitoring, which is provided by our surveillance system. This paper presents related works and problems of our previous surveillance system, at first. An overview of the new version of our surveillance system and its functions are described in the next section.

## Intended Audience and Document Overview

The goal is to design a software for a Image Processing based Multi-Utility System which will cater anyone who want to ensure trespassing protection in his/her absence.Target audience is Academicians which can be aided by Security Guards and Security Officers.

As our users will have different educational background and expertise level in using the system. Our goal is to develop software that should be easy to use for all types of users, including the Security Guards. Thus while designing the software one can assume that each user type is mobile-literate and has little or no difficulty in using android application to access information such as room status.

## Definitions, Acronyms and Abbreviations

|  |  |
| --- | --- |
| Admin | Person who is responsible for the upkeep,configuration and reliable operation of the system |
| CPU | Central Processing Unit |
| GPIO | General purpose input/output ports. |
| GPU | Graphics Processing Unit |
| JPEG | Joint Photographic Experts Group |
| METU | Middle East Technical University |
| MJPEG | Motion Joint Photographic Experts Group |
| Mah | Milliampere per hour |
| OS | Operating System |
| OpenCV | A real time computer vision library that is free for academic and commercial purposes. |
| Raspberry Pi | A low-cost, basic computer |
| SRS | Software Requirements Specification |
| TCP | Transmission Control Protocol |
| TTS | Text to Speech |
| Tesseract Engine | An OCR tool to read text |
| UDP | User Datagram Protocol |
| USB | Universal Serial Bus |
| Visiondary | The name of the team which develops the System |

## References and Acknowledgments

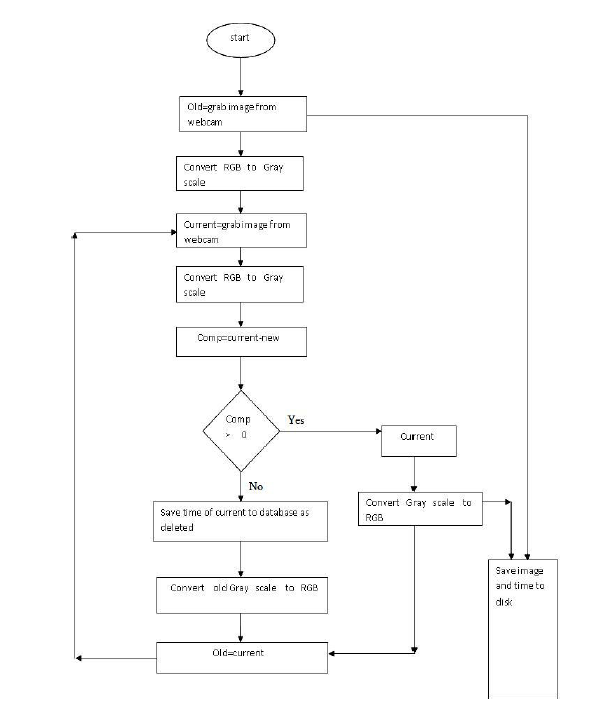
1. <http://www.ijmer.com/papers/(NCASG)%20-%202013/16>
2. <https://www.ijcsmc.com/docs/papers/November2015/V4I11201554>
3. <https://fenix.tecnico.ulisboa.pt/downloadFile/1689244997255576/Thesis>
4. <https://pdfs.semanticscholar.org/bf34/e342579ad79ac811dd3f7487a6c0ab8508d2>
5. <https://github.com/CankayaUniversity/ceng-407-408-2017-2018-project-face-and-iris-recognition/wiki/Software-Requirements-Specification>
6. <https://www.scribd.com/doc/76026766/HomeSecurity-SRS-documentation>
7. <http://www.cs.uwc.ac.za/~dwarren/Dmitri%20Warren%20De%20Klerk%20-%202653786%20-%20Thesis>

# Overall Description

## Product Overview

Web cams have been long used for motion detection but we thought of upscaling the project and using it as a low cost surveillance tool saving resources and improving security.

For comparison process there are many approaches but here frames are converted from RGB format to GRAY scale format and then compared. Again the frames are converted back to RBG format before storing. The gray scale conversion is done to decrease the pixel values. Whereas the frames are compared pixel by pixel. The difference in the frame gives only the moving objects on the frame. The main objective of the proposed approach is to reduce the storage size by storing only the frame having motion instead of the whole video.



In the web cam selection the user first of all initialize the camera then specify frame size. The frame size and frequency will be changeable because of frame saved in the specified location. In this location image stored as frame. It will then start to capture the video, it will set the first image as a background, the next image will be taken with the same frame size “x‟&‟y‟ i.e. width & height. Now we have to calculate the image mean position and mean velocity. For any given frame of video, we can subtract the background image from it to check the motion (the detection for motion is purely pixel to pixel detection). Those pixels with a result near zero are treated as background and those pixels with a larger result are treated as foreground. Thus, once we have the model of the background image, this is simple, efficient, and easy to implement.

## Product Functionality

1. Capturing Video
2. Detection Of Motions
3. Temporal Difference
4. Background Modeling
5. Synchronization With The Database

## Design and Implementation Constraints

1. Camera is very important constraint. If it does not work no image will be captured from the environment and thus both image processing part and web-stream module will fail.
2. In case of the usage of product in some extreme conditions such as cold, hot or wet, some precautions should be taken to keep device working.
3. Cost of the hardwares should be in a reasonable margin to be affordable.
4. Components that are used in hardware should not be threat to the human life or should be safe enough to use.
5. Battery life and recharging time are also important. Recharging should not be more than 10 hours and also battery life should not be less than 6 hours.
6. Can only be implemented in static background environments.

## Assumptions and Dependencies

1. The project was started with the assumption that we would be given the necessary support in the form of hardware and software resources. Our project depends a lot on the inputs.
2. User has the basic knowledge of the system environment.
3. The users have sufficient knowledge of security system.
4. The home security system should have the product installed correctly.
5. Whoever can interact with our home security systems such as the lightening system, phone system, alarm etc in enable mode
6. We are assuming that the space to which we are providing trespassing protection is not a very big area as larger area will require camera synchronization feature.
7. We are not protecting high valued objects
8. System running when required (Power Backup)
9. Area sufficient lighted as not using infrared
10. Static Environment
11. Area covered with cameras completely No Blind Spots

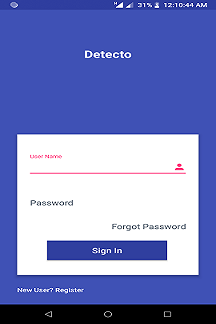
# Specific Requirements

## External Interface Requirements

The framework holder is anticipated to be able to utilize button, pull-down menus, and comparative devices. A window with graphical client interface will be utilized. To begin with, of all we have chosen to store the pictures of people/objects detected and utilize those for later inspection. The admin after logging in can search for activity of a particular date. In case the input date is invalid, at that point a message will be prompted, otherwise activity database would be shown. The layout of the required screen designs, report formats, and menu structures will be chosen on.

### Basic User Interfaces

**Login Screen**



Fields -

Username - User will enter his/her UserName here

Password - User will enter his/her Password here.

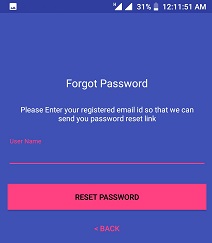
Button Clicks -

Sign In - If the username and password entered above is correct and if the internet connection is working, this action will take the user to activity detection screen which will show the history of activity detected in the given area and place.

Forgot Password - If user somehow forgot his/her password,he may click on this button and it will direct him to Forgot Password Screen.

New User?Register - If user is using his/her app for the first time, he should click on this button to get himself registered.

**Forgot Password Screen**



Fields-

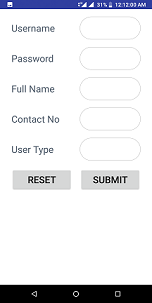
UserName - User will enter his/her UserName

Button Clicks-

Reset Password - If a user exists in database having UserName as entered in UserName field, this will send you a link which will further guide you in changing your password.

Back - This will take you back to Login Screen.

**User Registration Screen**



Fields-

UserName - User will enter his/her Email Id

Password - User will enter his/her Password (which he wants to set)

Full Name - User will enter his/her Full Name

Contact No - User will enter his/her Contact Number

UserType - User will choose his/her UserType

0 Professors

1 - Security Officers

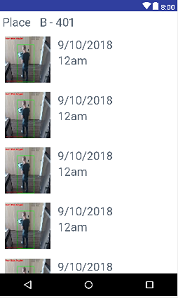
2 - Guards

Button Clicks-

Reset - Sets all fields to empty

Submit - Sends info of all fields to server.If the username is unique, user details will be added to system.

**Activity Detection Screen**



Fields-

Place - User may select the place to look for detection history.

The screen will show a list of motion detection along with image proof,date and time details.

### Hardware Interfaces

Hardware devices would primarily include a Camera, a PC (Admin Access) and Mobile Devices. The Camera will be used for capturing images. The Mobile Devices are used for receiving notifications.

|  |  |
| --- | --- |
| Logitech WebCam HD C310 | 1. HD Video calling: 1280 X 720 pixels 2. HD video capture: Up to 1280 X 720 pixels Logitech Fluid Cyrstal™ Technology 3. Photos: Up to 5 megapixels 4. Built-in mic with noise reduction 5. Hi-Speed USB 2.0 certified |

### Software Interfaces

Windows 10 operating system will be used during development process. The system will be implemented using in python language. For image processing part and to convert processed data into sound, OpenCV and Haar-Cascade libraries will be used respectively. Android Studio would be used for mobile application development.For mobile notifications, firebase messaging will be used

The used software tools, their versions and sources are given in the table below:

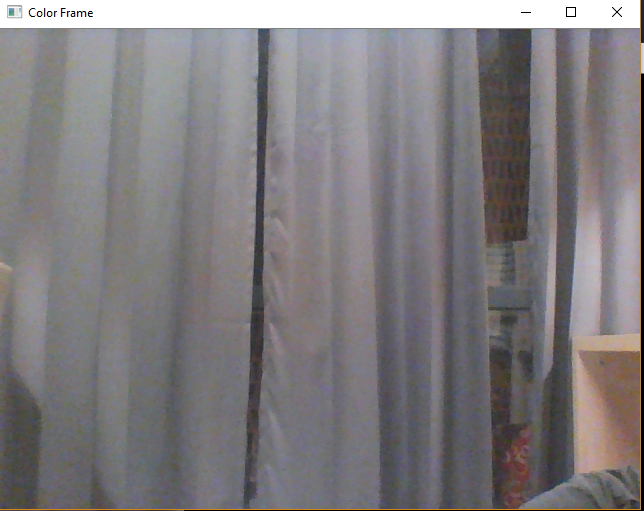
|  |  |
| --- | --- |
| **Software Product & Version** | **Source** |

|  |  |
| --- | --- |
| Windows 10.0.17134.285 | Default |
| OpenCV 3.0 | http://opencv.org/ |
| Python 3.7 | https://www.python.org/ |
| Android Studio 3.1.4 | https://developer.android.com/studio/ |

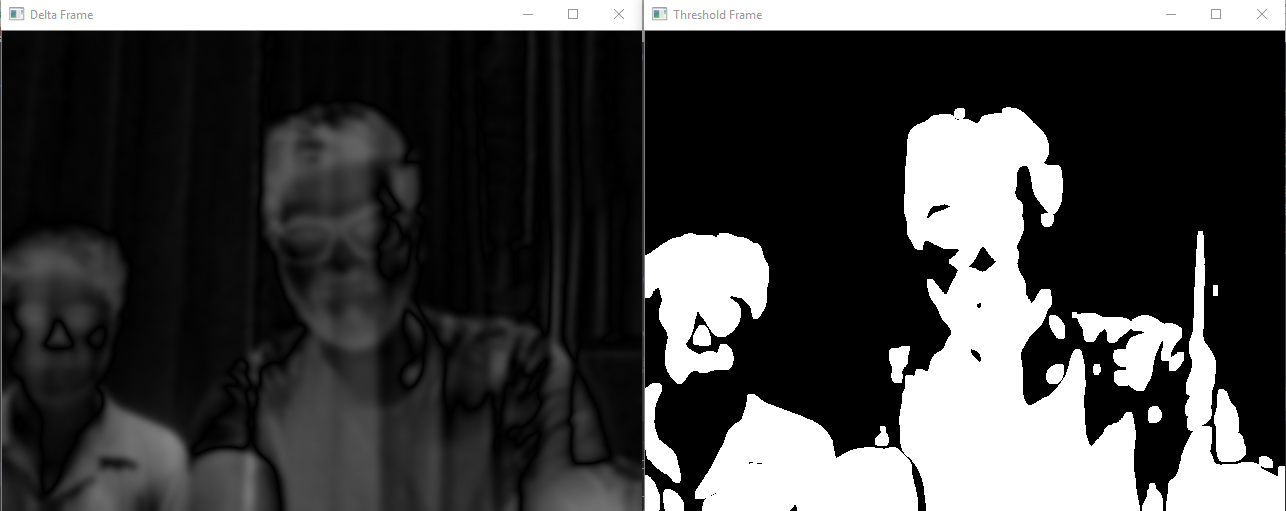
## Functional Requirements

Image processing has been done with the following procedures; two target images extracted from database are divided into fourth or ninth pieces of sub image, corresponding two pieces of sub image are compared with pixel-wise operations, and detection of dynamic changes can be performed based on the result whether compared two sub images are different or not.

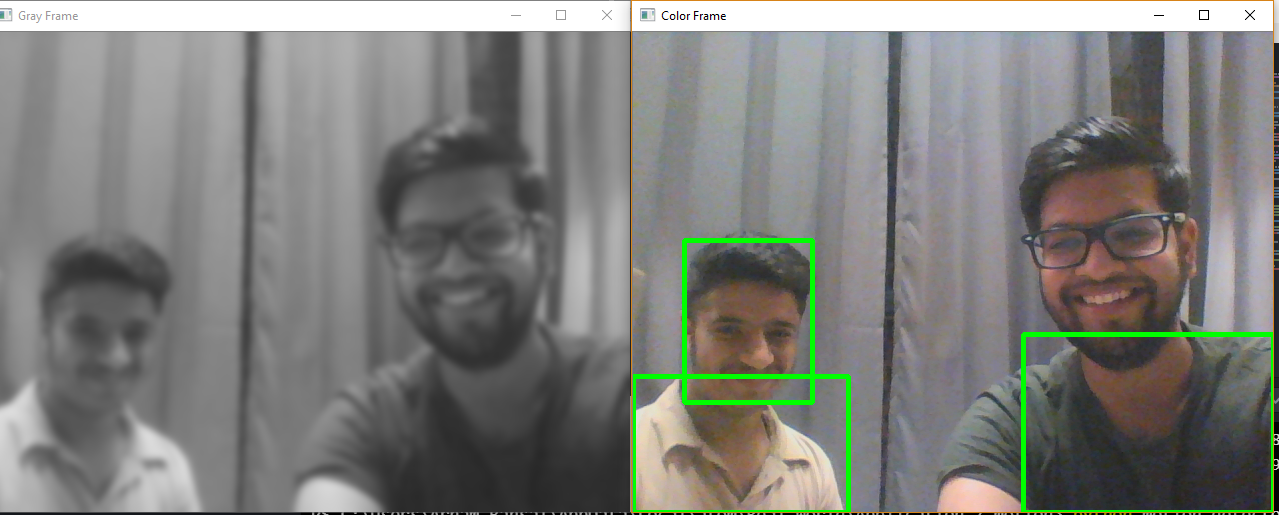
1. **CAPTURING VIDEO:** The capture video images must be specified the position x, y, width, height. This position calculates the images (frames). Position (x, y) indicating the mean position of the feature on the human body.
2. **DETECTION OF MOTIONS** : In detection of motion process, first of all the background image (initial frame) must be converted into gray scale and then the current image is converted into the gray scale image since in a gray scale conversion the range of the color is from 0-256 only. Reducing the number of point features in each model evaluation requires us to be more intelligent about choosing them in the first place. It would be impractical to simply find the point motion throughout the entire image, and feed all resulting features to the model in a single evaluation. First of all, the number of features would be very large, giving poor performance.Secondly, we would be supplying a large number of extraneous point features, such as those that are part of the background. In order to avoid wasting time on the background, we use image segmentation to separate foreground portions of the scene from the background by subtracting the converted current image to initial image. Methods for motion detection can be categorized into two main classes, i.e. pixel-based and region-based algorithms.



1. **TEMPORAL DIFFERENCE :** The comparing of consecutive frames on a pixel by pixel basis in a motion sequence is calculated and threshold is applied which classifies them as either stationary or in motion. However, as objects moves, their homogeneous interiors do not result in changing image intensities over short time periods, so motion can only be detected at the boundaries. Also, it does not show relationship of pixel with its neighborhood.



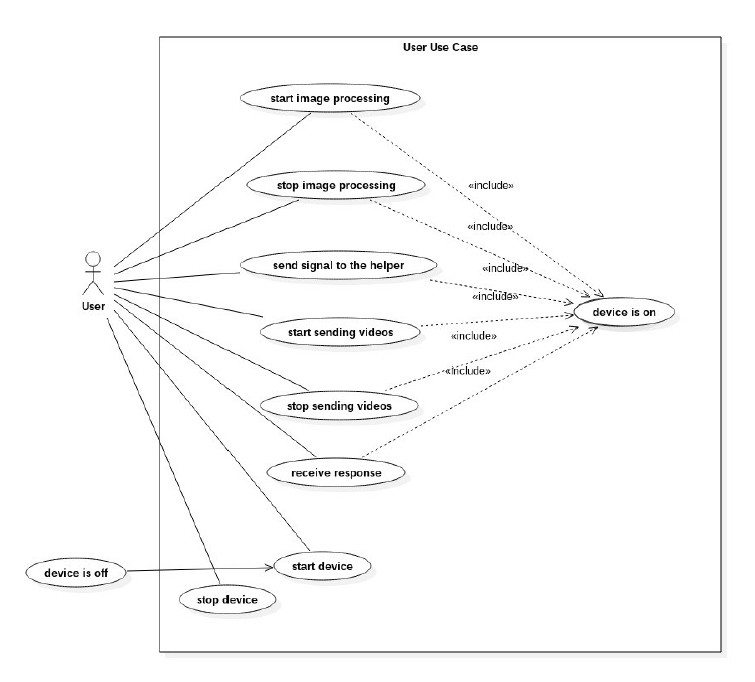
1. **BACKGROUND MODELING :** Background modeling methods can be classified as pixel-based and region-based models. For pixel-based background modeling, background subtraction is often used. An image of the stationary background is generated by averaging the image sequences over a period of time on a mixture of Gaussian distributions. Each incident pixel color is computed by comparing it with every Gaussian density. If two pixels do not match, they will be labeled as the foreground pixels and be detected as motions. We are using the co-occurrence algorithm for the comparison of frames. In this co-occurrence algorithm, each point pixel is compared i.e. (location based ratio of the pixels).



1. **SYNCHRONIZATION WITH THE DATABASE :** This Module is basically needed to check the continuous working of the application or we can say that to check the continuous working of webcams. Basically in this module a log file is created which keeps record of all the image compared and deleted with its time of deletion. Let’s take a scenario in which a camera in a supermarket stops working at 11:00 pm just half an hour after closing of the market and starts working at 9:00 am next morning before opening of the market. So it will be practically impossible to know that webcam was working all the time as no two same images are saved. So in this section all the images taken compared and deleted will have a entry in a log fill with reference to its time.

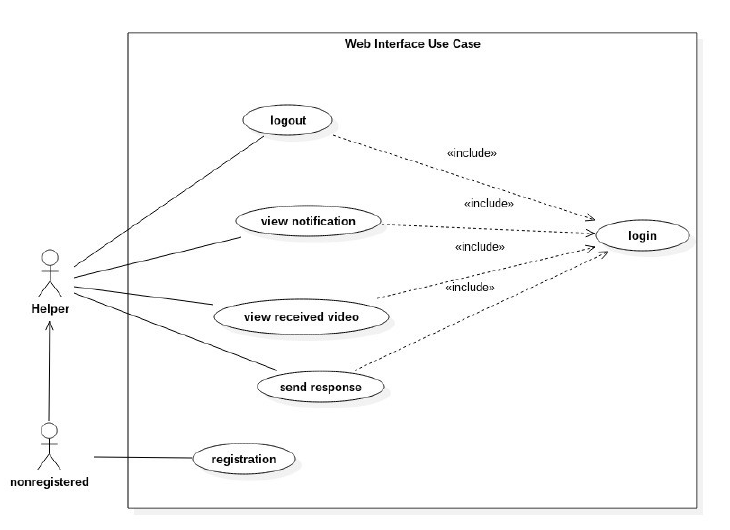
1. **LOGIN MODULE :** This module is used for user login.The user will enter his details and these details will be checked as specified in database and he will be given access if they match.This module will also help with new user registration process and user forgot password actions.

## Use Case Model

1. **User use case**

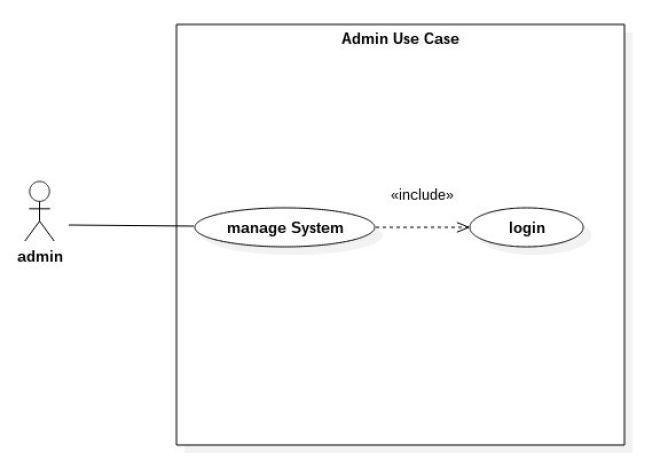
|  |  |
| --- | --- |
| Start image processing | User can start image processing by pushing a button |
| Stop image processing | User can stop image processing by pushing a button |
| Start sending videos | User can start sending videos by pushing a button |
| Stop sending videos | User can stop i sending videos by pushing a button |
| Receive response | User will get the response message as speech via headphones |
| Start device | User can start device by pushing a button |
| Stop device | User can stop device by pushing a button |

2. **Web interface use case**



|  |  |
| --- | --- |
| Login | The helper has to login to connect device. |
| View notification | The helper can view the user's alerts |
| View received video | The helper can view the videos sent by the user.  The helper can gives directives to the user by |
| Sending Response | Using Chat Screen |
| Registration  Logout | The helper should register if he is not registered to system.  The helper can logout when the interaction is over. |

3.  **Web interface use case definition**



|  |  |
| --- | --- |
| login | Admin has to login to manage the system |
| Manage the system | Admin can manage the system |

**User :**

User will wear the device and send a signal to the helper by pushing a button etc. to say that he needs help. In case of the user is unable to reach anybody, the image processing part will be step in so that the user takes care of himself. Then the user will get the required information from headphones.

**Helper :**

Helper will access the device remotely via a web application. The helper can view the user's environment from video screen and send response back to the user via chat screen.

**Admin :**

Admin is responsible for the upkeep, configuration and reliable operation of the system. The system administrator seeks to ensure that the uptime, performance, resources and security of the system in order to meet the needs of the user.

# Other Non-functional Requirements

## Performance Requirements

Image data transfer through internet connection and live streaming makes performance measures crucial.

For desired performance, image capturing, transfered data size, speed of connection, response time, processing speed must be considered.

System should work real-time which means there should be an acceptable time delay such as max 4-5 seconds between request and response.

should have wifi adapter which is fast enough to transfer live camera feed to the web server.

Image processing should be optimized so it should not take time more than 2 seconds. Web server should not process every frame and should determine whether process or not the frame.

## Safety and Security Requirements

The computer that runs the program will have its own security. Only the System Admin will log in to the system with his/her username and password.

Battery of the device should be covered by case or it should be placed on a belt or bag.

In addition to that voltage levels on device should be adjusted.

Noise or heat produced by the device should be minimized.

In case of malfunction, system should shutdown itself and reboot in order to prevent unpredicted results.

Streaming the device camera on web makes security measurements crucial. Accessing and interacting with the streaming web server should be controlled and any misusage should be prevented. User authorization and data encryption are important security requirements of the project. User stream should not be available to anyone who is not authorized by the user of the device.

System should store user data on database securely and set access permissions to the these datas carefully

## Software Quality Attributes

**Software Quality Attributes:**

**Interface is accessible from at least one or more convenient places.**

This includes all of the home's common entry ways, bedrooms, and frequently used rooms. If the system interface is not readily accessible, then it will not beas easy to control and will offer little convenience.

**Interface and system must be properly connected to the sensors.**

It is essential to have secure wiring, preferably inside the walls, for reliable sensor information to guard against false alarms and tampering.

**System must have a backup power supply.**

The system needs to have a 24 x 7 power supply in order to thwart attempts to bypass the security system.

# Other Requirements

**Usability**

The system must be easy to learn for both users of the wearable device and helpers who are the users of the web interface.The web interface elements (e.g. login, video display page) will be easy to understand. There will be a help page and complete user documentation which will explain how to achieve common tasks. Error messages must give the user specific instructions for recovery. The help system will explain all functions and how to achieve common tasks.

**Supportability**

The system shall allow the system administrator to add additional features. The system needs to be cost-effective to maintain.

**Reliability**

It is expected that, camera, sensor and other tools works perfectly until their life time expires. The reliability of the web interface has also a crucial importance, If the web interface crashes, it must be repaired in at most 30 minutes.