Home Security System

**Software Requirements Specification**

Version 1.0

Nhóm 2 :

Nguyễn Văn Tấn Đạt

Ngô Văn Tiến

Lưu Đình Hải Châu

Nguyễn Hoài Vinh

Trương Văn Quốc Hoàng

Danang, Sep 2016

**TABLE OF CONTENTS**

[**TABLE OF CONTENTS**](#_gjdgxs)

[1. DOCUMENT HISTORY](#_30j0zll)

[2. REFERENCE DOCUMENTS](#_1fob9te)

[3. DISTRIBUTION LIST AND APPROVALS](#_3znysh7)

[4. INTRODUCTION](#_2et92p0)

[4.1 Purpose](#_tyjcwt)

[4.2 In Scope](#_3dy6vkm)

[5. ACTORS](#_1t3h5sf)

[6. OVERVIEW](#_4d34og8)

[6.1 System overview](#_2s8eyo1)

[6.2 Requirements and Requirement Use cases](#_17dp8vu)

[6.3 System Pages Flow](#_3rdcrjn)

[7. APPENDIX](#_26in1rg)

# DOCUMENT HISTORY

|  |  |  |
| --- | --- | --- |
| **Date** | **Summary of Changes** | **Version** |
| 2/9/2016 | Create SRS document | 1.0 |
| 15/9/2016 | Fix use cases and screenshot | 1.1 |
| 19/09/2016 | Fix use cases and update 2.5.6 | 1.2 |

# REFERENCE DOCUMENTS

|  |  |
| --- | --- |
| **Document Name** | **Description** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# DISTRIBUTION LIST AND APPROVALS

|  |  |
| --- | --- |
| **Name** | **Title** |
| Nguyễn Văn Tấn Đạt | * Design use case * Design screen |
| Ngô Văn Tiến | * Specify use case * Support use case |
| Lưu Đình Hải Châu | * Fix typos * Update technology and data variations list |
| Nguyễn Hoài Vinh |  |
| Trương Văn Quốc Hoàng | * Fix typos * Update use cases and reference between use cases |
|  |  |
|  |  |
|  |  |
|  |  |

# INTRODUCTION

## Purpose

Build an app to monitor a room or a small house with only one camera. Camera can swing, detect object, and capture object.

## In Scope

# ACTORS

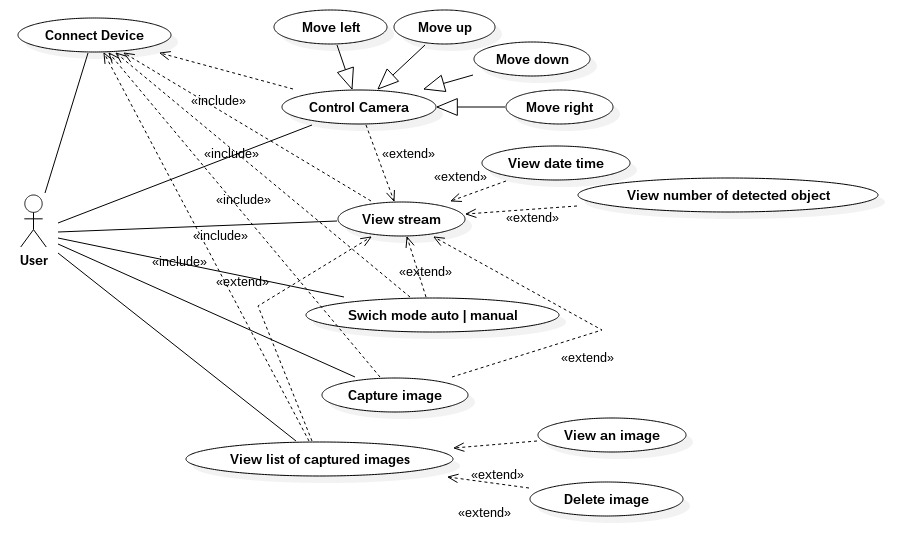
|  |  |
| --- | --- |
| **Actor** | **Description** |
| Normal User | Has the right to Control camera, View records, View images |

# OVERVIEW

## System overview

* View stream of a camera
* Camera can automatically swing, detect object and capture images in Auto mode
* User can switch mode to Manual mode, to control Camera, capture images
* User can also view captured images and delete images

## Requirements and Requirement Use cases



**6.2.1. Choose Device**

|  |  |
| --- | --- |
| **Use Case** | Choose Device |
| **Actors** | User |
| **Priority** | High |
| **Pre-condition** | * App is opened. * Wifi is turned on |
| **Post-condition** | Successfully connect to a camera |
| **Main success scenario** | Camera is connected |
| **Special requirements** | * Click on the icon on the screen of the app. * App show list of devices (camera) * User choose one camera in list to connect. |
| **Technology and Data Variations List** | Wifi |
| **Screenshot** |  |
| **Frequency of Occurrence** |  |
| **Open issues** |  |

**6.2.2. View stream**

|  |  |
| --- | --- |
| **Use Case** | View stream |
| **Actors** | User |
| **Priority** | High |
| **Pre-condition** | * App is opened * Camera is connected |
| **Post-condition** | Stream will be playing |
| **Main success scenario** | Stream video is connected |
| **Special requirements** | * App will display current footage of the camera. * There is a hidden menu bar to the right to select the optional tasks |
| **Technology and Data Variations List** | - Flask framework for streaming server |
| **Screenshot** |  |
| **Frequency of Occurrence** |  |
| **Open issues** |  |

**6.2.3. Control Camera**

|  |  |
| --- | --- |
| **Use Case** | Control Camera |
| **Actors** | User |
| **Priority** | High |
| **Pre-condition** | * App is opened * Camera is connected * User clicked on the button on top right of the application |
| **Post-condition** | User a control camera, application through the buttons |
| **Main success scenario** | User can control the camera or capture or switch mode, and also view date time, and number detected objects |
| **Special requirements** | * Press the menu top right corner to move to the main screen work. * The screen includes navigation bar, select mode, the camera controls and buttons, switch tasks and current parameters * Auto mode: see **6.2.5** * Joystick can control the camera with 4 directions: up-down-left-right * Capture button: see **6.2.6** * View image: see **6.2.7** * The biggest rectangle show video stream from camera * The smaller rectangle below display information of object inside the image being captured, more detail in **6.2.4** |
| **Technology and Data Variations List** | * Use 2 servos to control camera move up/down, left/right: 1 for X-axis, 1 for Y-axis * Use raspberry pi to control servos, raspberry act as middleware role between server and camera controller * Control camera actions (Auto mode, Capture image actions) by sending request to server |
| **Screenshot** |  |
| **Frequency of Occurrence** |  |
| **Open issues** |  |

**6.2.4. View date time and number of detected objects**

|  |  |
| --- | --- |
| **Use Case** | View date time and number of detected objects |
| **Actors** | User |
| **Priority** | High |
| **Pre-condition** | * App is opened * Camera is connected * User click on the button on top right of application |
| **Post-condition** | User can see information of the captured images |
| **Main success scenario** | User can view time and number detected objects |
| **Special requirements** | * When camera automatically capture image, or user press capture button, it will trigger this functionality * Application will try to detect object from the captured image and display. * Display format: each line for each image including: date and time and number of object detected. * Can be scroll if there are many lines |
| **Technology and Data Variations List** | - OpenCV for video capture, image capture and face detection |
| **Screenshot** |  |
| **Frequency of Occurrence** |  |
| **Open issues** |  |

**6.2.5. Switch mode Auto | Manual**

|  |  |
| --- | --- |
| **Use Case** | Switch mode Auto | Manual |
| **Actors** | User |
| **Priority** | High |
| **Pre-condition** | * App is opened * Camera is connected * Users click on the button on top right of stream to show controls |
| **Post-condition** | Auto mode: Camera automatically move to monitor the house and capture image if objects are detected.  Manual mode: User move the camera through application |
| **Main success scenario** | User can switch between Auto mode and Manual mode |
| **Special requirements** | * User click on toggle button with label Auto mode. * If auto mode is on: camera will automatically swing and capture image depend on the object * Maximum 3 pictures is captured within 5 minnutes * If auto mode is off: camera will receive the signal from application to swing or capture images |
| **Technology and Data Variations List** | * See **6.2.3** |
| **Screenshot** |  |
| **Frequency of Occurrence** |  |
| **Open issues** |  |

**6.2.6. Capture images**

|  |  |
| --- | --- |
| **Use Case** | Capture images |
| **Actors** | User |
| **Priority** | High |
| **Pre-condition** | * App is opened * Camera is connected * User click on the button on top right of stream to show controls |
| **Post-condition** |  |
| **Main success scenario** | User press button Capture → image will be saved. |
| **Special requirements** | Watch the entire video function was over and saved in memory. It helps to better manage and can review the video effect. |
| **Technology and Data Variations List** | * See **6.2.3** |
| **Screenshot** |  |
| **Frequency of Occurrence** |  |
| **Open issues** |  |

**6.2.7. View list of captured images**

|  |  |
| --- | --- |
| **Use Case** | View list of captured images |
| **Actors** | User |
| **Priority** | High |
| **Pre-condition** | * App is opened * Camera is connected * User click on the button on top right of stream to show controls * User click on View Images button |
| **Post-condition** | The list of images will be shown sorted by date |
| **Main success scenario** | Show the list of saved images |
| **Special requirements** | Show both videos and images that have been captured |
| **Technology and Data Variations List** |  |
| **Screenshot** |  |
| **Frequency of Occurrence** |  |
| **Open issues** |  |

**6.2.8. View an image**

|  |  |
| --- | --- |
| **Use Case** | View an image |
| **Actors** | User |
| **Priority** | High |
| **Pre-condition** | * App is opened * Camera is connected * User click on the button on top right of stream to show controls * User click on View Images button * User click on a specific image |
| **Post-condition** |  |
| **Main success scenario** | The images will be shown at full screen |
| **Special requirements** | * Show image with a small rectangle surround the face of object * There are 2 arrow button to go next, or previous image * Bottom of screen: show date time, and number of object detected in the current image |
| **Technology and Data Variations List** |  |
| **Screenshot** |  |
| **Frequency of Occurrence** |  |
| **Open issues** |  |

**6.2.9. Delete images**

|  |  |
| --- | --- |
| **Use Case** | Delete images |
| **Actors** | User |
| **Priority** | High |
| **Pre-condition** | 1. View list of captured images  2. Press and hold an image |
| **Post-condition** |  |
| **Main success scenario** | The button delete will show up on top left of image  User can click to delete an image |
| **Special requirements** | When click on x button, that image will be deleted |
| **Technology and Data Variations List** |  |
| **Screenshot** |  |
| **Frequency of Occurrence** |  |
| **Open issues** |  |

## System Pages Flow

# APPENDIX