

Design Specification for Smart Home Guard System

Instructor: Prof. Seok-won Lee

Prepared by :

<Team 3>

Byungwook Lee

Hyesun Lim

Sunghoon Byun

Nov 11, 2015

Contents

Contents	2
Figure & Table	3
Revision History	3
1. Introduction	4
2. System Overview	4
2.1. Package Diagram.....	4
2.1.1. Introduction	4
2.1.2. TIZEN	5
2.1.3. Arduino	6
2.1.4. Raspberry Pi.....	7
2.2. Deployment Diagram	8
3. Domain Analysis and Design.....	9
3.1. Introduction	9
3.2. Design.....	10
4. Layered Architecture	16
5. User Interface Design.....	17
5.1. TIZEN UI.....	17
6. Circuit diagram.....	20
6.1 Arduino Circuit Diagram	20
6.2 Arduino and Raspberry Pi2 Connection Diagram	21
6.3 Camera Module and Raspberry Pi2 Connection Diagram	22
7. Reference	23

Figure & Table

Figure 1. Package diagrams of TIZEN system	5
Figure 2. Package diagrams of Arduino system	6
Figure 3. Package diagrams of Raspberry Pi system	7
Figure 4. Deployment diagram of Smart Home Guard System	8
Figure 5. Layered Architecture of Smart Home Guard System	16
Figure 6. Arduino Circuit Diagram-Lab View	20
Figure 7. Arduino Circuit Diagram-Schematic View	20
Figure 8. Arduino and Raspberry Pi2 Connection Diagram	21
Figure 9. Connection between Raspberry Pi and Camera module	22
Table 1. Revision History	3

Revision History

Doc. Name	Date	Reason For Changes	Version
Design	30 th of Oct	Initial version of Design Document	V 0.1
Design	4 th of Nov	Added UI diagram, Package diagram	V 0.2
Design	6 th of Nov	Added Domain analysis & design, Layered architecture	V 0.3
Design	10 th of Nov	Added Circuit diagram, Complete version of proposal	V 1.0

Table 1. Revision History

1. Introduction

This document is written for developers who implement the system to provide more detailed technical help. This document contains the design diagrams of each requirement and the whole architecture of the system.

In Section 2, whole system overview is provided. Three package diagrams (for TIZEN, Arduino, Raspberry Pi) and the deploy diagram are provided for helping developers to understand the overall architecture.

Section 3 has detailed design diagrams matching with each requirement. Design Class Diagrams and Design Sequence Diagrams are given. Using those diagrams, developers can understand the static view and the dynamic behavior of each requirement.

In section 4, a layered architecture is provided. In section 5, UI design is provided. In section 6, with the perspective of hardware, we present circuit diagram for overall system.

2. System Overview

2.1. Package Diagram

In this document we consider three packages to create our SHGS. There are TIZEN system, Arduino system and Raspberry Pi system. TIZEN system control the SHGS to monitor the situation with which user is confronted. Arduino system has two factors which are sensing environment and actuating system. Lastly, Raspberry Pi system can process the control of TIZEN system or Arduino system. Additionally, we draw package diagram using StarUML [1].

2.1.1. Introduction

The SHGS is the core of the Home-IoT technologies which are sensing an activity of household, real-time streaming for sending a situation or video to user who has TIZEN phone and equipped the SHGS. In order to give these function, the system consists of three components; TIZEN, Arduino and Raspberry Pi.

2.1.2. TIZEN

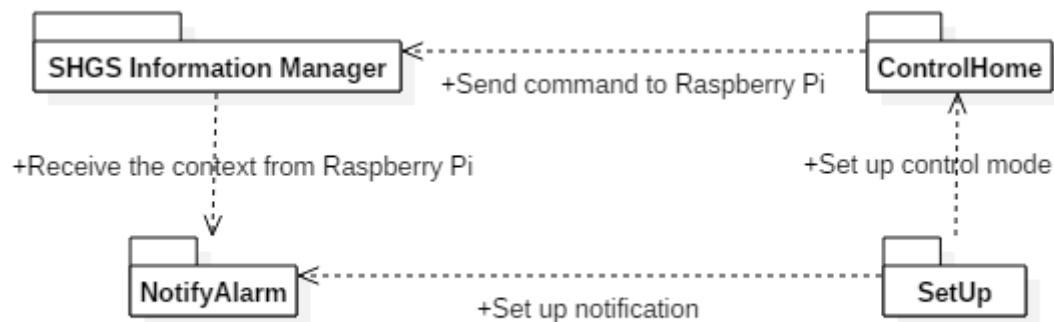


Figure 1. Package diagrams of TIZEN system

TIZEN system have 4 packages; each package performs below roles respectively.

- **SHGS Information Manager:** SHGS information manager sends processing information from Raspberry Pi system to TIZEN system for notification to user. Also, SHGS information manager receives user command for delivering to Raspberry Pi system.
- **Control Home:** Control home sends command information for manipulating the SHGS controlled by Raspberry Pi system. Also, Control home is set up by SetUp package for setting up control mode.
- **Notify Alarm:** Notify alarm receives the context from SHGS information manager, which is delivered from Raspberry Pi. Also, Notify alarm is set up by SetUp package for setting up notification information.

Set Up: Set up controls the set up information for control home package and notify alarm package. In control set up, there are Lock the door, Turn on TV, Turn on light, and Sound Up. Also in notify alarm set up, there are safe mode window and danger mode window setting.

2.1.3. Arduino

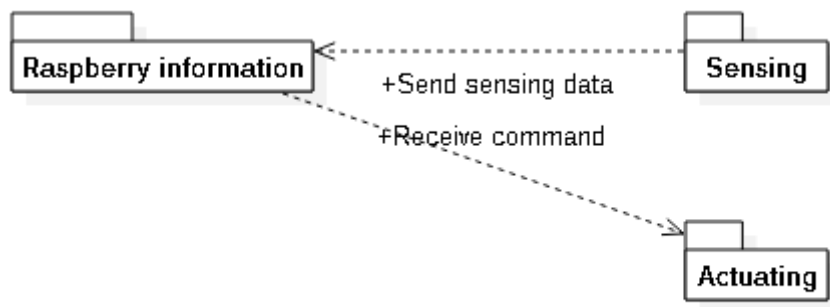


Figure 2. Package diagrams of Arduino system

Arduino system has 3 packages; each package performs below roles respectively.

- **Raspberry information:** Raspberry information receive the sensing data from sensors which are infrared sensor and switch sensor, and they do perceive inside motion and notify doorbell information, respectively. Also, Raspberry information sends the command from TIZEN to actuators.
- **Sensing:** Sensing sends the sensing data from sensors which are infrared sensor and switch sensor, and they do perceive inside motion and notify doorbell information, respectively.
- **Actuating:** Actuating receives command from Raspberry information in order to act right performance. There are LED light, Door lock (servo motor), and Piezo Buzzer.

2.1.4. Raspberry Pi

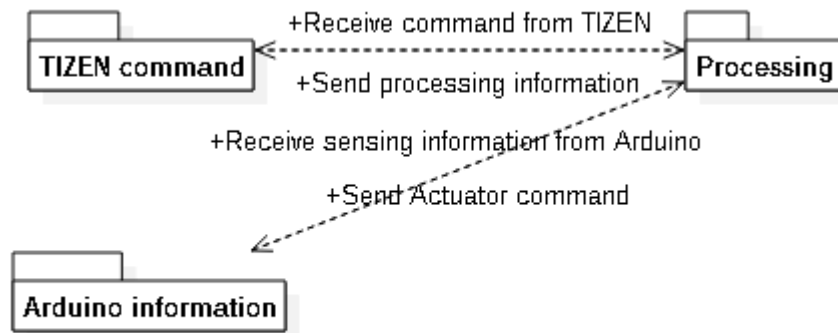


Figure 3. Package diagrams of Raspberry Pi system

Raspberry Pi system has 3 packages; each package performs below roles respectively.

- **Processing:** Processing receives command mode from TIZEN system in order to set the SHGS what the user want, and sends the command to Arduino information for actuating the actuators.
- **TIZEN command:** TIZEN command receives processing information from processing package, it means that TIZEN system will be notified by Raspberry Pi system. Also, TIZEN command sends the command from TIZEN to processing package, it is about the user's command, such as turning on TV, turning on the light, and so on.
- **Arduino information:** Arduino information receives actuator commands from processing package in order to perform the actuators. Also, Arduino information sends the sensing data to processing package, it will be processing in processing package and send the information to TIZEN system.

2.2. Deployment Diagram

Deployment diagram gives information that environment of each devices and packages and how these devices communication works through interfaces. Additionally, we draw package diagram using draw.io [2].

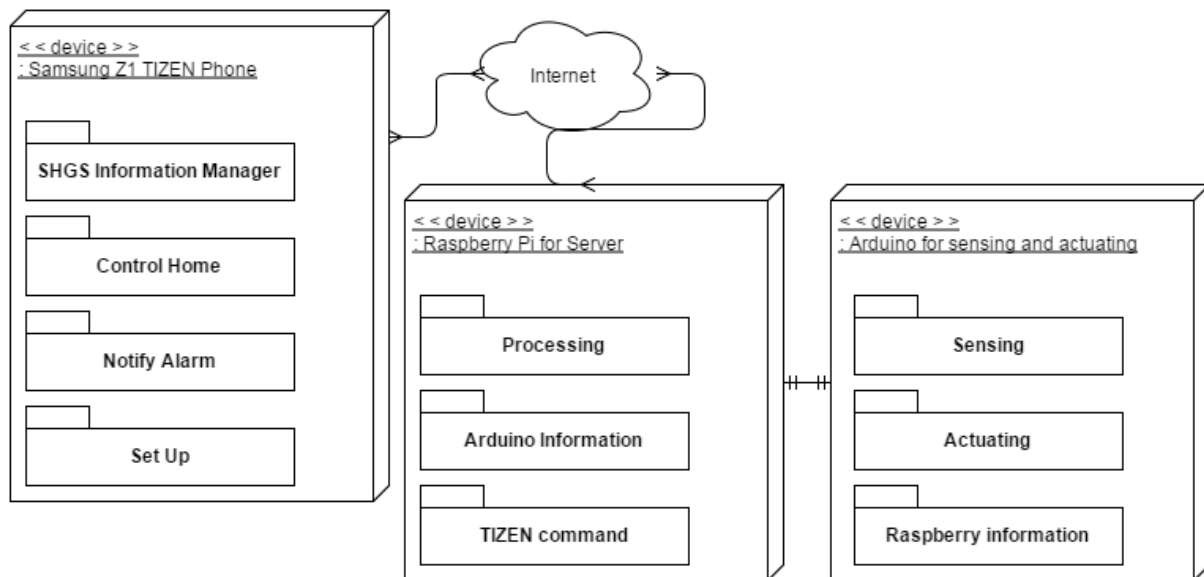


Figure 4. Deployment diagram of Smart Home Guard System

3. Domain Analysis and Design

3.1. Introduction

In the last section, we gave a very high-level view over our system. This section's objective is to provide the reader with a good understanding of the details of the logical architecture of our system. We want to explain the static structure of our system, using object-oriented design. We will explore each of our use cases, and we will elicit which classes are necessary. We will also provide a dynamic view over the requirement, so that to explain in what order each function should be called. We use UML class diagrams and UML sequence diagrams respectively to model the static and dynamic aspects of our design. We model using web tool, draw.io [2] and websequencediagrams.com [3].

3.2. Design

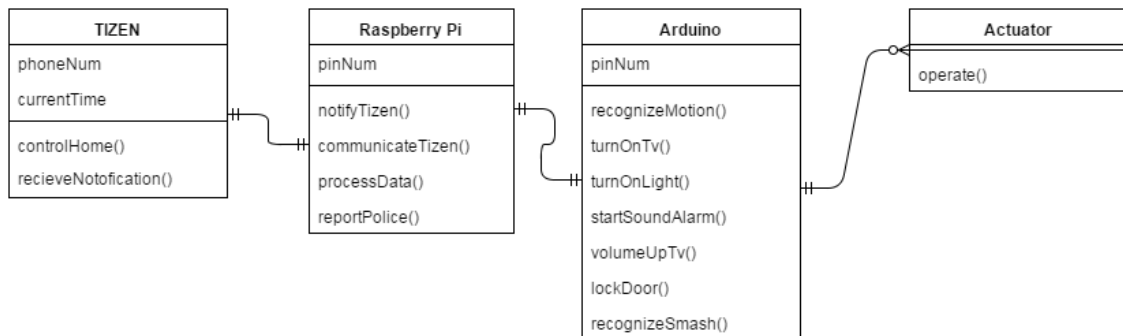
We draw the Class Diagram, Sequence Diagram based on use case which derived from Goal graph.

UC001: Controlling home in ordinary day

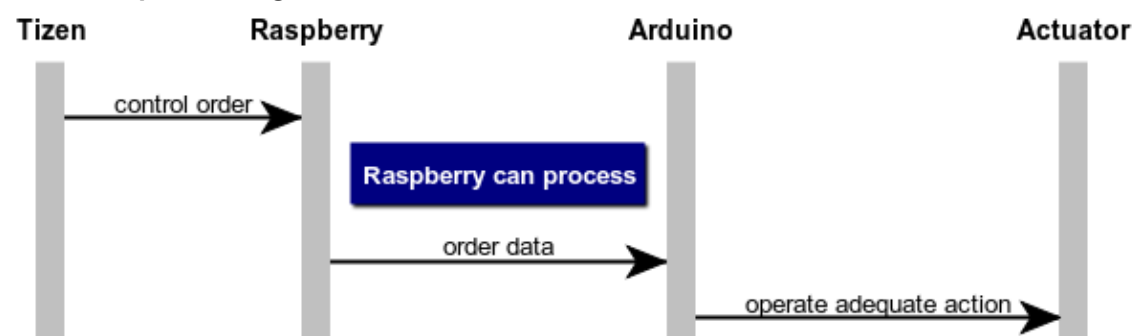
UC001 Class Description

1. User wants to pretend that there is someone at home.
2. User executes mobile application installed on TIZEN.
3. User turn on TV and volume up TV
4. User turn the light on.
5. If user want to change the status of the above, he can control them with TIZEN.

UC001 Class Diagram



UC001 Sequence Diagram

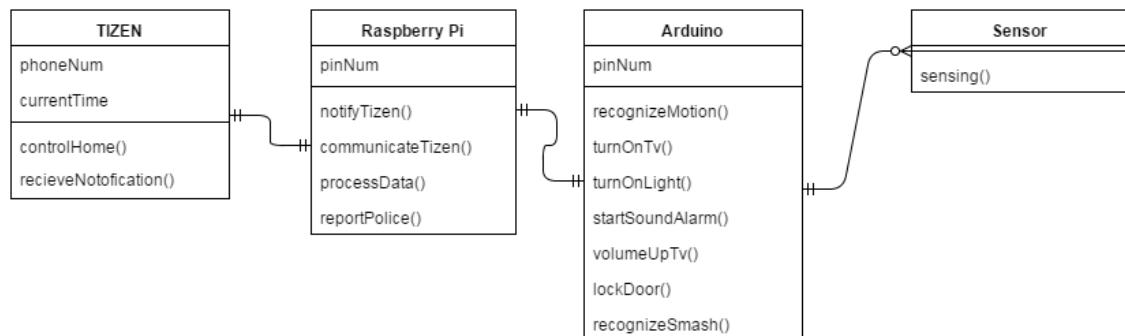


UC002: Communicating with visitors

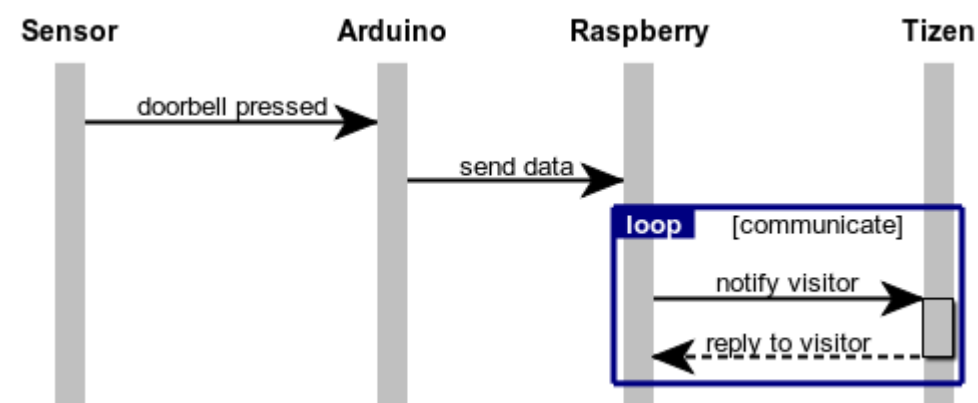
UC002 Description

1. User is carrying his phone.
2. Normal visitor visits user's home.
3. Visitor pressed a doorbell.
4. A notification alarm arrives on the user's phone.
5. User presses 'confirm' button on the phone's screen.
6. User confirm who the visitor is and converse with him.
7. And then, user can deal with the reason why the visitor has visited.

UC002 Class Diagram



UC002 Sequence Diagram

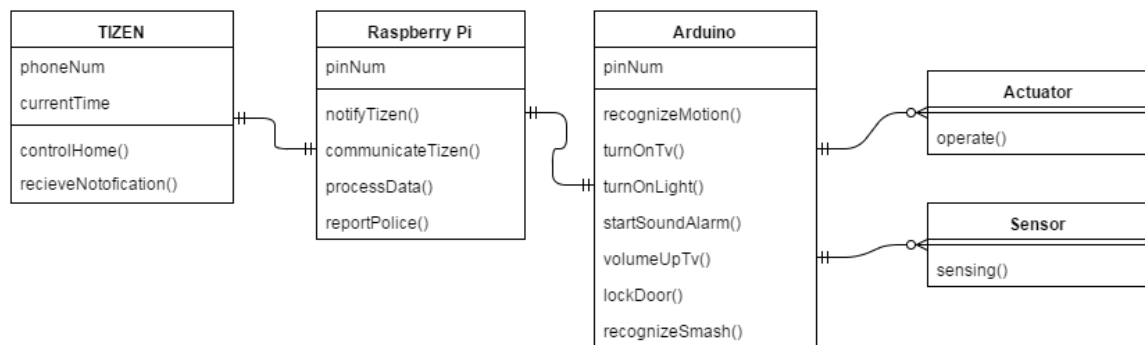


UC003: Replying to malicious visitor

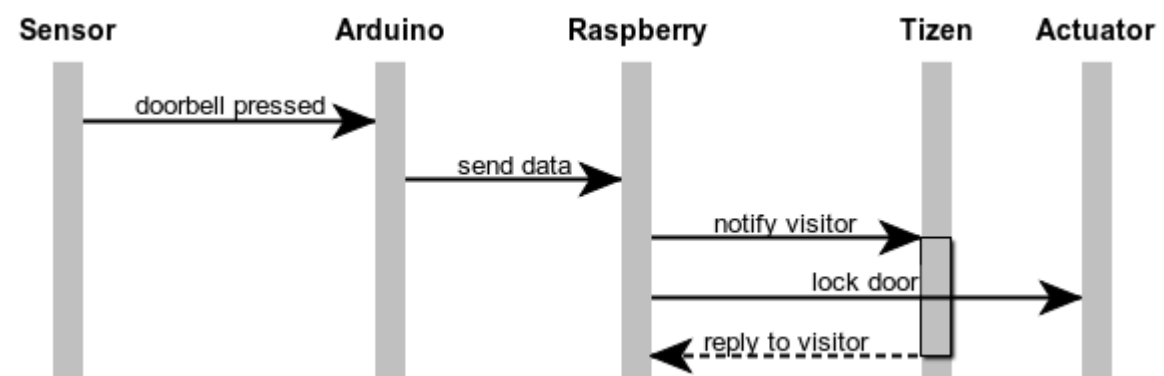
UC003 Description

1. User went out of the home at night.
2. User is carrying his phone.
3. A malicious visitor visits user's home.
4. Visitor pressed a doorbell.
5. A notification alarm arrives on the user's phone.
6. User presses 'confirm' button on the phone's screen.
7. User confirm the person who the visit home and notice that he is suspicious.
8. User locks the door and windows.

UC003 Class Diagram



UC003 Sequence Diagram

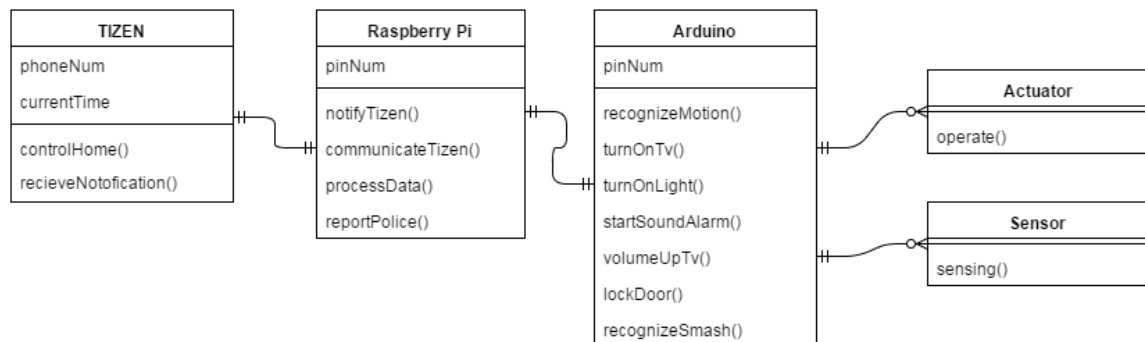


UC004: Automatically protect home

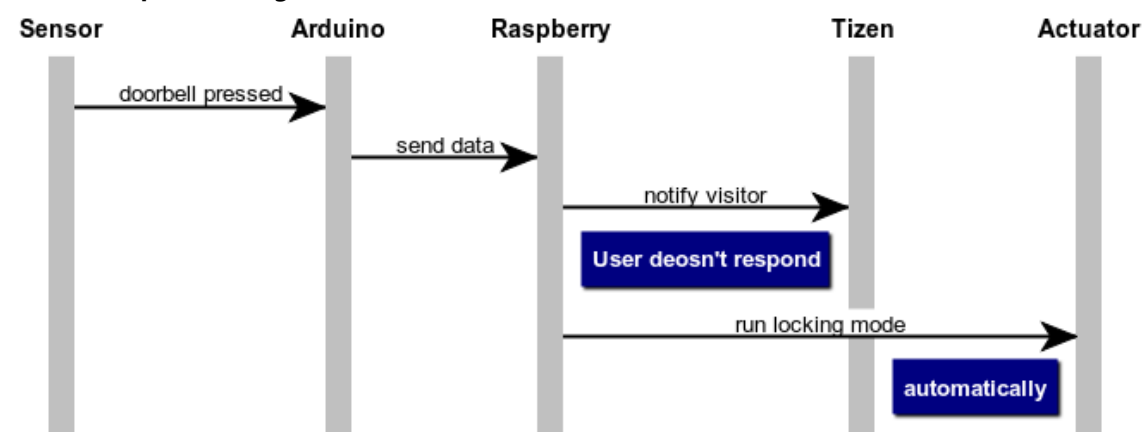
UC004 Description

1. User is not able to check his phone's notification alarm.
2. A malicious visitor visits user's home.
3. Visitor pressed a doorbell.
4. A notification alarm arrives on the user's phone.
5. User cannot confirm the notification.
6. The pre-configured functions are automatically executed.
7. The malicious visitor cannot enter into user's home.

UC004 Class Diagram



UC004 Sequence Diagram

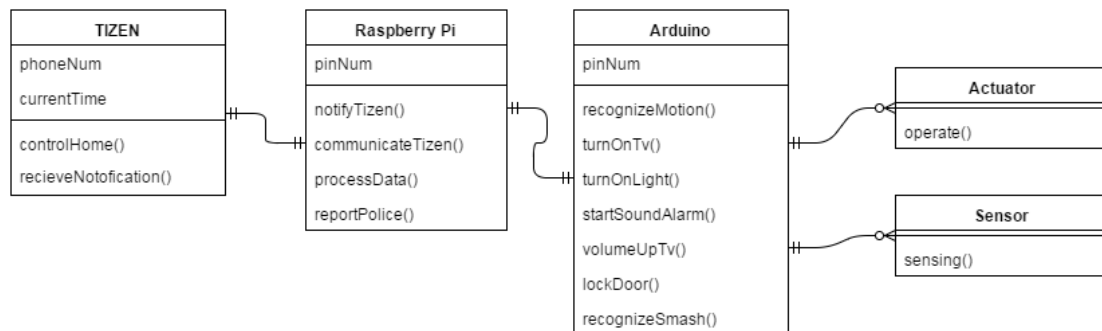


UC005: Automatically protect home

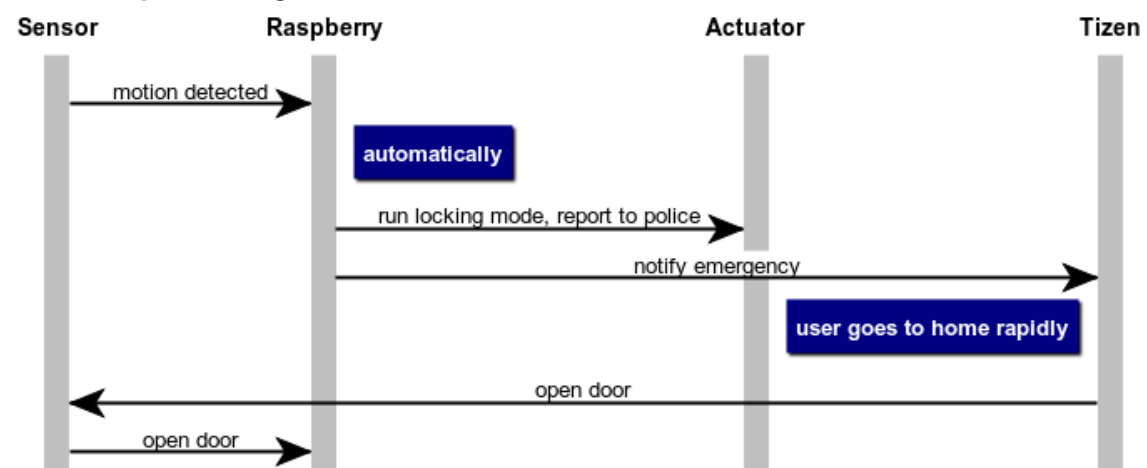
UC005 Description

1. While user is out of home, the window is opened.
2. User's phone receive an emergency alarm.
3. Alarm say that the motion is detected at home.
4. The police arrive at user's home.
5. The user arrive at home.
6. The police open the door with user's phone and enter into home.
7. The police arrests the thief.

UC005 Class Diagram



UC005 Sequence Diagram

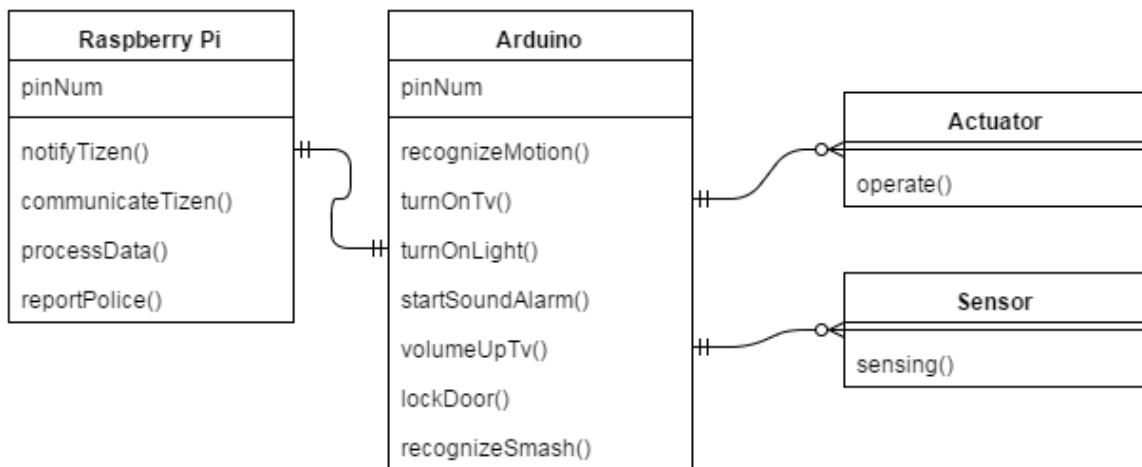


UC006: protecting home during sleep

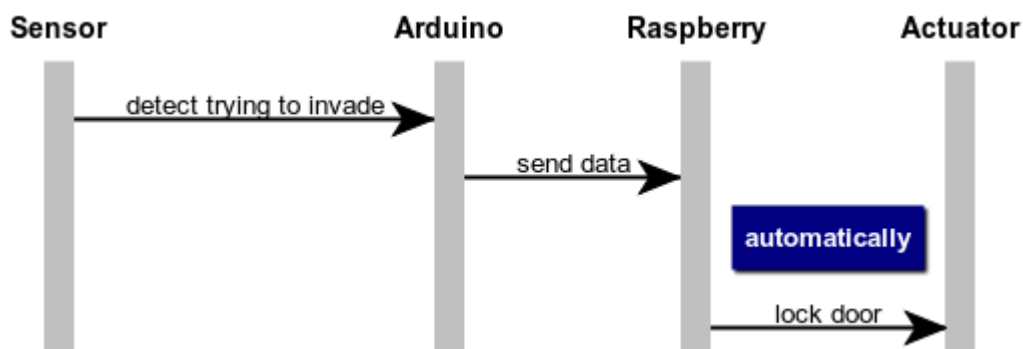
UC006 Description

1. User is sleeping.
2. A thief try to open the door forcibly.
3. The pre-configured functions are automatically executed.
4. Because of the system's functions, the thief cannot enter into use's home.

UC006 Class Diagram



UC006 Sequence Diagram



4. Layered Architecture

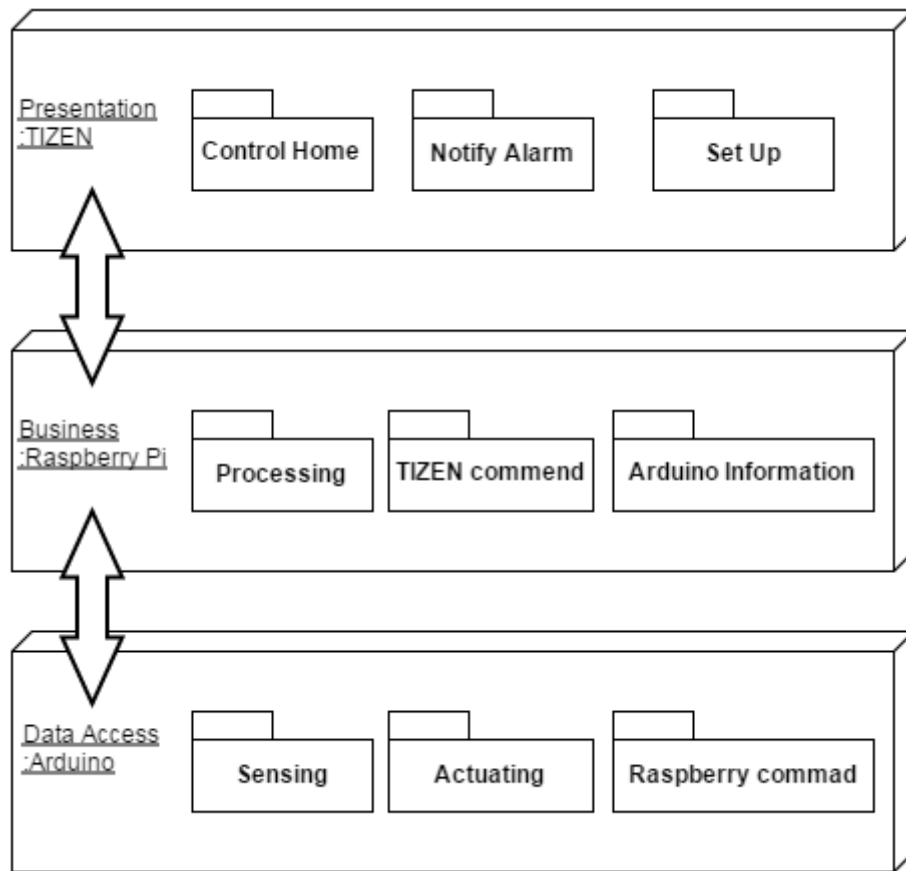


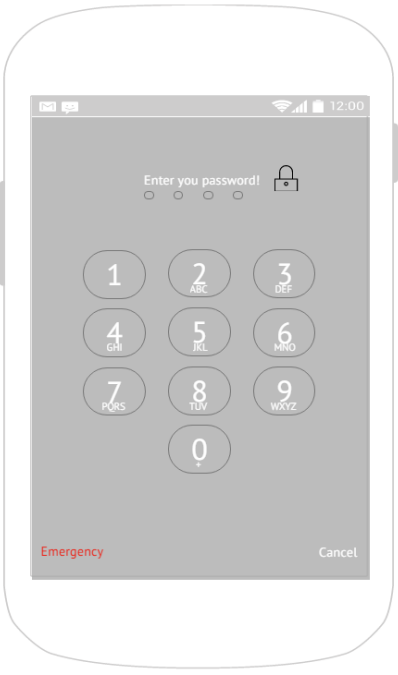
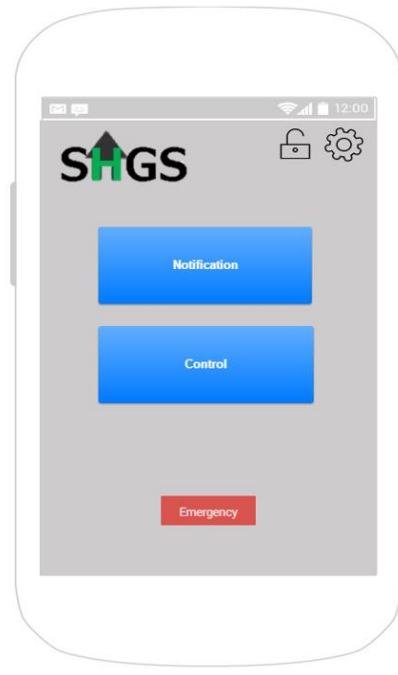
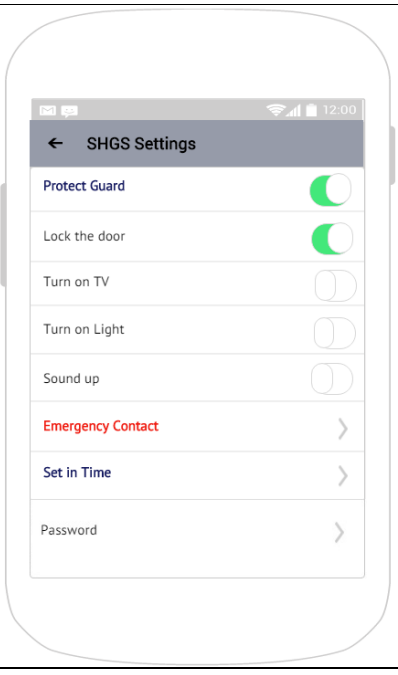
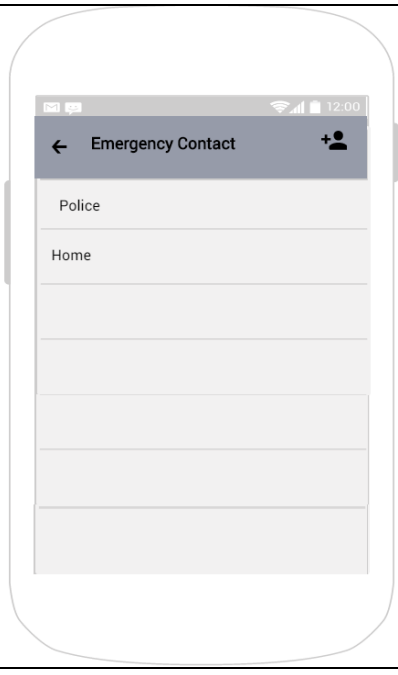
Figure 5. Layered Architecture of Smart Home Guard System

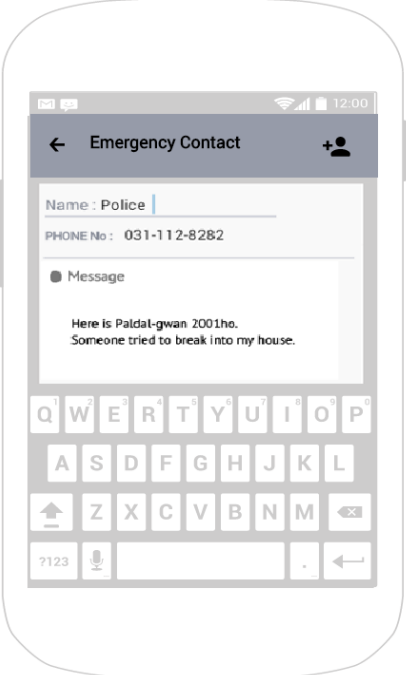
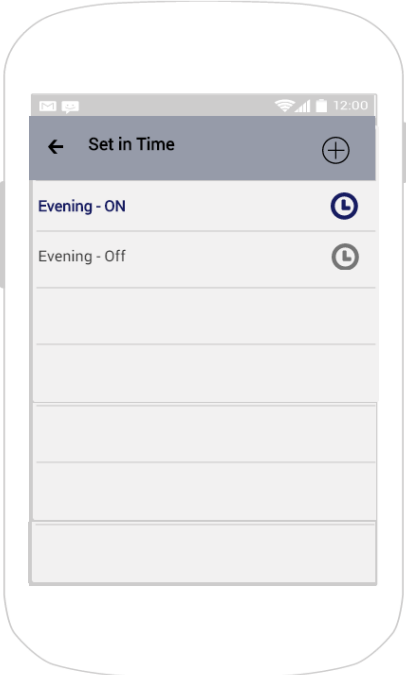
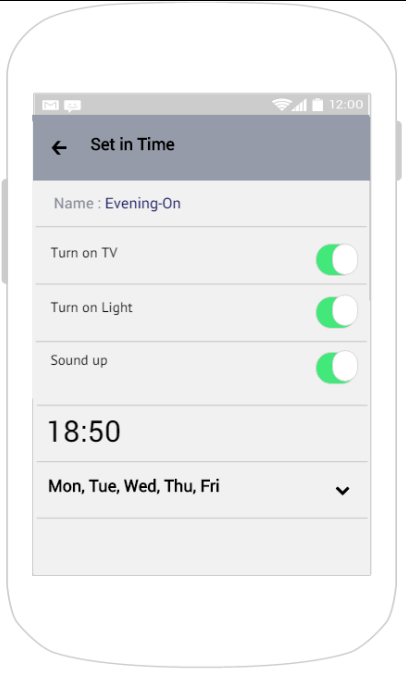
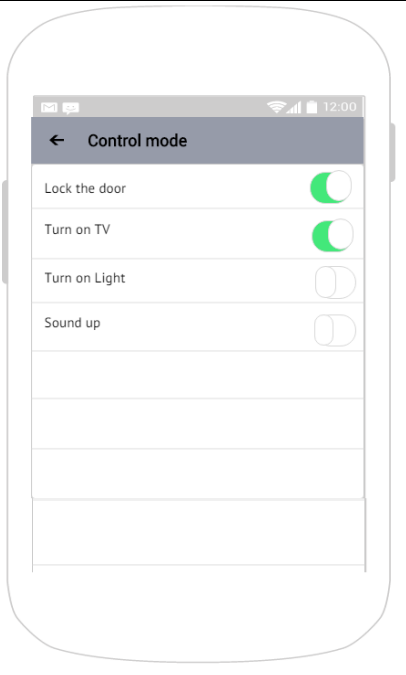
Figure 5 shows the layered architecture of SHGS. Each layer represents presentation, business and data access which are represented for TIZEN, Raspberry Pi, and Arduino, respectively. In each layer, packages are already explained before sections. Additionally, we draw package diagram using draw.io [2].

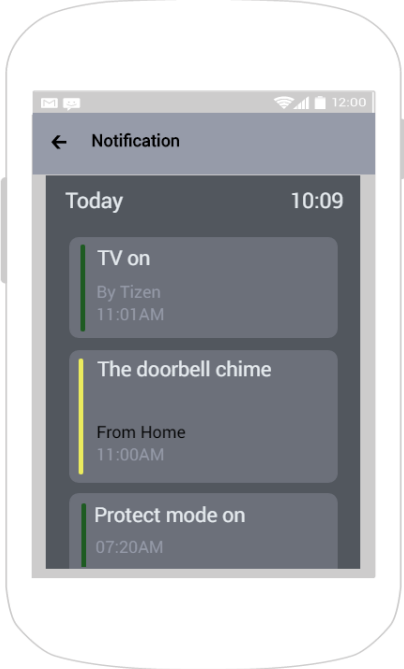
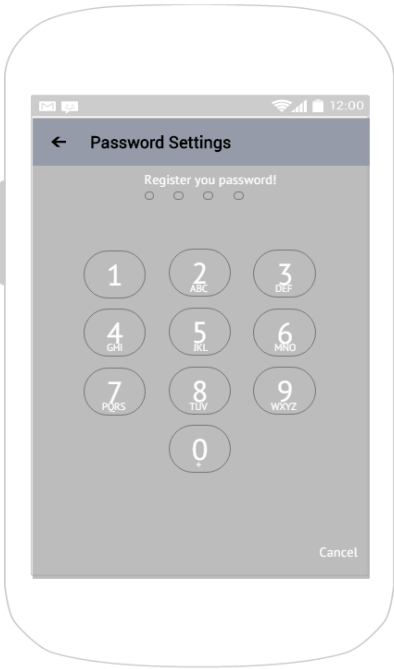
5. User Interface Design

We were used UI mockup tool [4] for UI design. It divided the Main Menu, SHGS settings and sub-menu.

5.1. TIZEN UI

Lock release by password input	Main Menu
	
SHGS Settings	Emergency Contact List
	

Emergency contact - Input	Set in Time List
	
Set in Time - Input	Control
	

Notification	Password Setting
 A mockup of a Tizen notification screen. At the top, there's a status bar with icons for mail, messages, signal, battery, and the time 12:00. Below is a header bar with a back arrow and the title 'Notification'. The main content area shows a group of notifications under the heading 'Today' with a timestamp of 10:09. The notifications are: 'TV on' by Tizen at 11:01 AM, 'The doorbell chime' from Home at 11:00 AM, and 'Protect mode on' at 07:20 AM. Each notification has a vertical bar on its left side.	 A mockup of a Tizen password setting screen. At the top, there's a status bar with icons for mail, messages, signal, battery, and the time 12:00. Below is a header bar with a back arrow and the title 'Password Settings'. The main content area has the text 'Register your password!' followed by four empty circles for a visual password. Below this is a numeric keypad with buttons for digits 1-9 and 0, each with its corresponding letters (e.g., 2 has ABC, 3 has DEF). A 'Cancel' button is located at the bottom right.

We will develop the Tizen app to using UI shown here.

6. Circuit diagram

In SHGS, we use various sensors and actuators. For smooth development at implementation phase, we made a circuit diagram that shows board and each of connected sensor and actuator. Additionally, we draw package diagram using 123d Circuits [5].

6.1 Arduino Circuit Diagram

Arduino use RGB LED(2,3,4 pin), Buzzer(5 pin) as TV function, LED(6 pin) as light function, Servo Motor(12 pin) as locking door function, Buzzer(10 pin) as warning function, pressure sensor(A0 pin) as detecting trespassing.

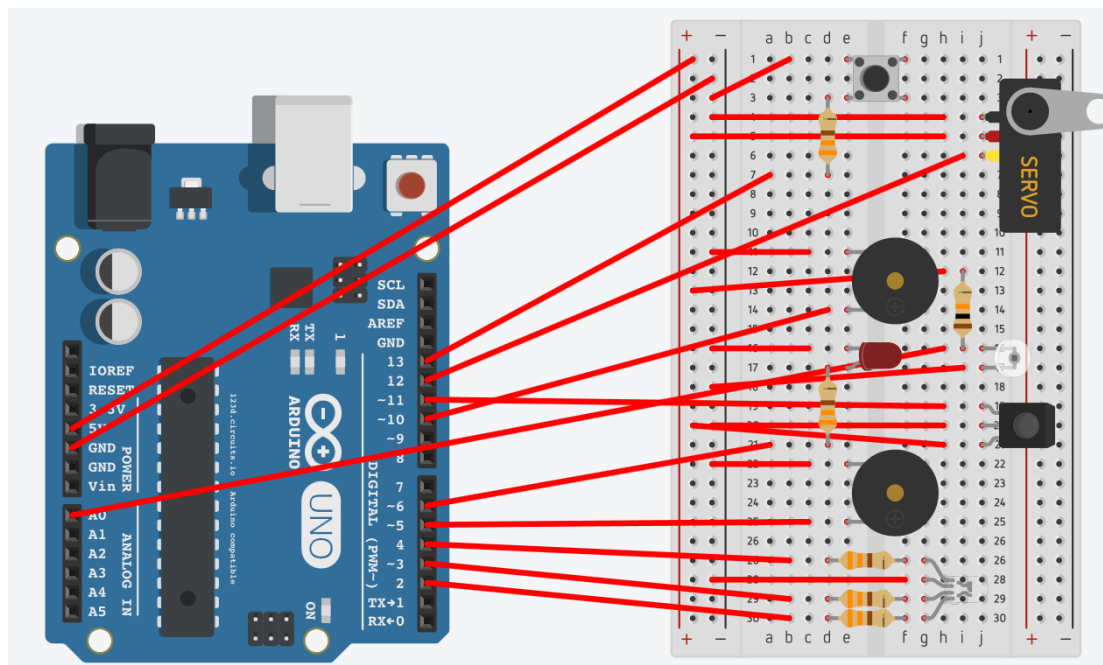


Figure 6. Arduino Circuit Diagram-Lab View

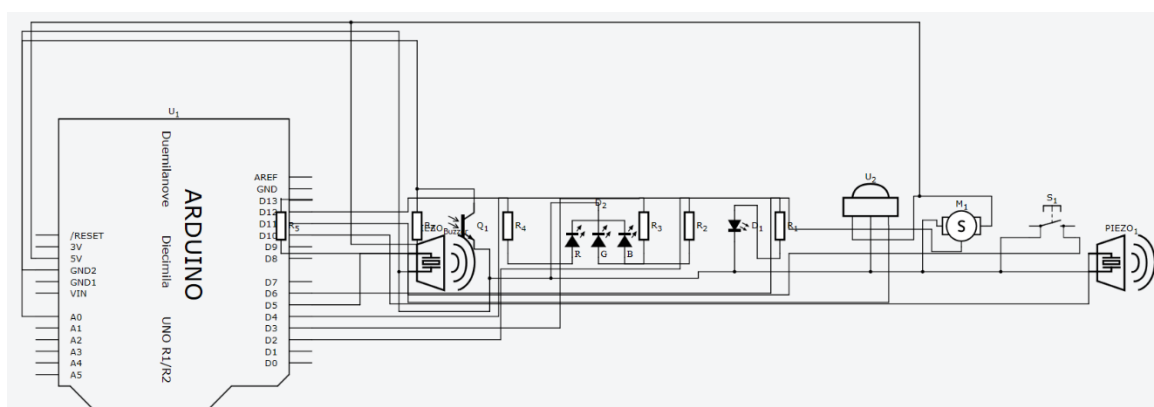


Figure 7. Arduino Circuit Diagram-Schematic View

6.2 Arduino and Raspberry Pi2 Connection Diagram

Board uses RX and TX to process data between Raspberry Pi and Arduino.

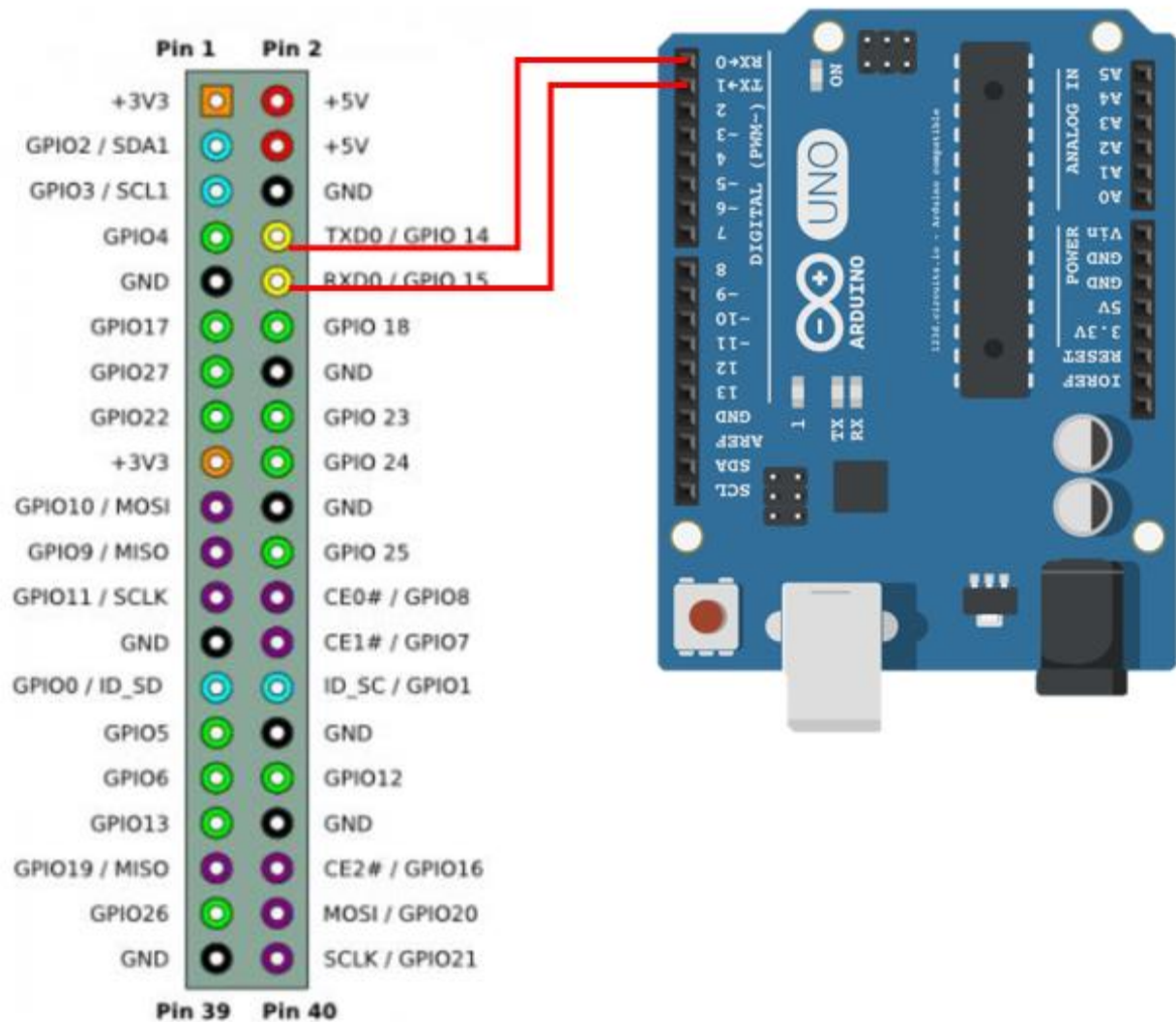


Figure 8. Arduino and Raspberry Pi2 Connection Diagram

6.3 Camera Module and Raspberry Pi2 Connection Diagram

When a visitor press the doorbell, we equip Raspberry Pi with camera module. Camera module will be connected to the red box area in Fig.9.

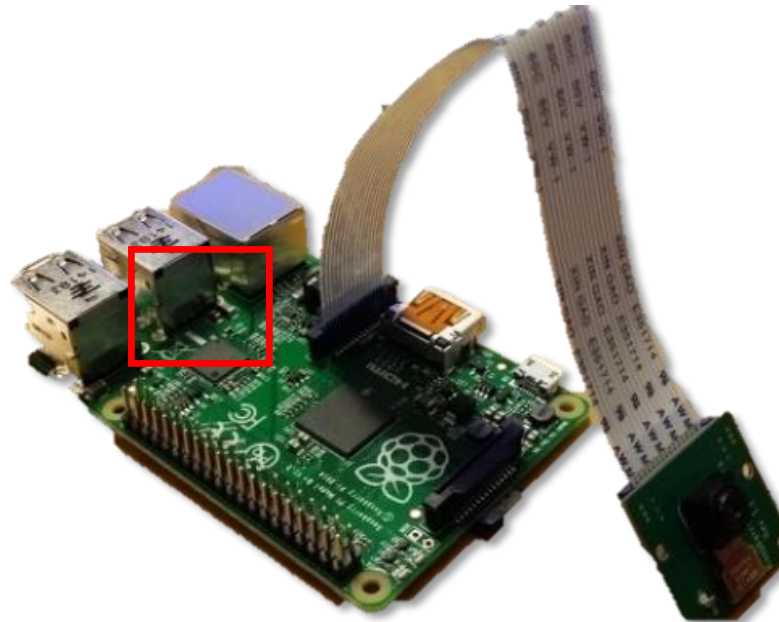


Figure 9. Connection between Raspberry Pi and Camera module

7. Reference

[1] MKLab, Co., *StarUML*, MKLab, Co., 2015.

[2] JGraph Ltd, *draw.io*, JGraph Ltd.

[3] Hanov Solutions Inc., "Web sequence diagrams," Available:
<https://www.websequencediagrams.com/#>.

[4] UXPin Sp., *UXPin*, UXPin Sp..

[5] 123d Circuits, "123d Circuits," Available: <https://123d.circuits.io/>.