

**Software Requirements Specification**

**For**

**Baby Monitor**

Course: ECCE-336-Intro to Software Engineering

Semester: Fall 2022

Section No.: 1

Students Names: Khaled Alhefeiti, Ahmad Mohammad, Saeed Almarri, Falah Alfalahi

Student Ids: 100058849, 100053567, 100053463, 100059123

Instructor: Dr. Davor Svetinovic

Teaching Assistant: Ruba Nasser

Link: <https://github.com/ECCE336-100053463/ECCE-336-Project>

Table of Contents

[1. Introduction 3](#_Toc119883254)

[2. System Requirements and Functional Requirements 4](#_Toc119883255)

2.1. System Requirements………………………………………………………………… … 4

2.2. Functional Requirements…………………………...…………………………………….6

[3. External Interface Requirements 7](#_Toc119883256)

[4. Non-functional requirements (NFRs) and Constraints 7](#_Toc119883257)

[4.1 Non- Functional Requirements: 8](#_Toc119883258)

[4.2 Constraints 9](#_Toc119883264)

[5. Use Cases 10](#_Toc119883266)

5.1 Use Case Model……………...…………………………………………………………..10

5.2 Use Cases………………… /…………………………………………………………… 10

[6 Data Dictionary: 15](#_Toc119883267)

[7 Tools: 15](#_Toc119883268)

1. Introduction

**1.1**  **Introduction**

The purpose of this Software Requirements Specification (SRS) document is to clearly define and describe all the functional and non-functional requirements of the project, and its constraints. While also displaying the use case model and each use case in detail.

**1.2**  **Scope**

The system's goal is to set up a portable app in which the parents Mouza and Ahmed are able to view their child wherever they may be. Only the four group members and the clients will have access to this document. We will try to make the most out of the budget at hand by minimizing unnecessary costs. The final phase of the project will be delivered by Friday the 9th of December.

**1.3**  **Overview**

The product is a Baby monitoring application with a live camera that shows live footage of the child’s room 24/7. Moreover, the system will send notifications of the wellbeing of the child and send alerts in the case of emergencies. Furthermore, the application can even contact the local authority in the case that the parents do not reply fast. Multiple devices will be able to use this application at once.

**1.4**  **Intended Audience**

This application is targeted directly to working parents that have to leave their child at home with a caretaker all day long until they get back home from work. It can be very helpful and ease the parents worrying significantly and not have call back home multiple times a day.

**1.5**  **Intended Use**

The baby monitor is helpful in multiple ways. For instance, it can access a live feed of the baby’s room anywhere anytime. Furthermore, it can also access recorded footage up to a week prior. Most importantly, it sends alert in the case of any emergency that occurs in the child’s room

**1.6**  **Definitions and Acronyms**

|  |  |  |
| --- | --- | --- |
| **Terminology** | **Acronym** | **Definition** |
| **Virtual Private Server** | **VPS** | Hosting service that uses virtualization technology to provide you with dedicated (private) resources on a server with multiple users. |
| **Non-Functional Requirement** | **NFR** | A requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours. |
| **Graphical User Interface** | **GUI** | a form of user interface that allows users to interact with electronic devices through graphical icons and audio indicators. |

2. System Requirements and Functional Requirements

**2.1 System Requirements:**

The system must be accessible by downloading the app through Google Play store for Android devices, or Apple’s App Store for iPhone devices.

Upon opening the app, the user should be requested to either log-in or register for an account, with inputs for an e-mail and password to enter to their account, as well as a phone number for emergency notifications.

If the e-mail and password combination is incorrect, a response message should appear to ask the user to try again.

If the e-mail has already been registered, it cannot be used to register a new account and a response message informing the user shall appear.

The system must be responsive through-out each action taken by the user, and most crucially, should be private and secure without any compromise on the safety of the user’s data.

Once a user has successfully logged in, they should be presented with a comprehensive monitoring feed that shows the overall system of monitoring devices connected.

The application should have a list allowing the user to navigate to different parts of the application, such as “Connected Devices”, “Settings”, “Monitor history”, among other things.

The “Connected Devices” tab should show the user all the devices that are currently linked to their account, as well as the option to remove or add other devices. Also, tapping on a specific device should take you to a screen showing more details pertaining to that monitoring device.

The “Settings” tab should contain options to customize the application and monitoring devices to a certain extent, as well as make changes to any personal information such as the e-mail, password, or phone number.

The “Monitor History” tab should present the logged history of each monitoring device, to allow the user to view past recordings.

Previous recordings are to be stored in a private server that is to be retrieved upon the request by the user to view past recordings through the Monitor History tab.

The VPS must be thoroughly secured from any unauthorized or malicious access, as it poses a very big threat to the privacy of the user.

When viewing a camera feed, the user should be able to zoom into the display if needed.

The system should maintain continuous contact with the VPS to monitor for emergencies, while the system’s database should always be ready to issue a notification to the user’s personal device in case of any emergency.

**2.2 Functional Requirements:**

|  |  |
| --- | --- |
| **Requirement** | **Description** |
| The system should allow a user to register and login to the application. | Input for email, username, phone number, and password is required. |
| The system should allow user to link their account(s) to their monitoring devices | Communication between the monitoring devices, the private server, and the user’s device or devices |
| The system should be able to display monitoring data at any point in time, consistently. | It is crucial that the server, and software must not have any major issues that could disrupt monitoring. |
| The system should be able to send notifications to the user’s personal devices | The user must be notified in the case of any indication of disturbance recorded from the monitoring devices. |
| The system should log monitoring data and retrieve it upon user request. | The user should be able to retrieve past monitoring data if requested. |
| The system should be able to display all types of monitoring data in one output page. | For a comprehensive monitoring experience, the user should be able to view all monitoring data in one page. (e.g, live camera feed, microphone feed, etc.) |
| The system should provide contextual help to set-up and pair the monitoring devices. | For a better user experience, the system should provide help to set-up and pair the devices. |
| The system should allow digital zoom-in and zoom-out of the camera feed from the display. | The user should be able to zoom into the camera feed if necessary. |
| The system should display a list of monitoring types. | The user should be able to select the type of monitoring to be displayed. (e.g, comprehensive, a specific device, etc.) |
| The system should allow the user to view a list of paired devices and remove or add if needed. | Provide the user with information as to which devices are linked to their personal device. |

3. External Interface Requirements

The Application will have a Graphical User Interface (GIU) which will be designed in a way that makes it easy to use and simple to understand. The navigation should also be effortless for the user and there should be an assistance feature which helps the user learn how to use the Application. Additionally, it should support a variety of devices. The priority is to have support for smartphones such as android and apple phones because it’s compact and can be used anywhere. Other platforms such as Windows and OSX will be supported later.

The application will use a database in the VPS to store users' account information. For communication, there will be a local authority phone number available in case of an emergency as well as an email for user support. The product in function should depend on external hardware such as a camera system that is installed around the house. There will be a way to control the camera itself and use functions such as zoom in and zoom out.

4. Non-functional requirements (NFRs) and Constraints

4.1 Non- Functional Requirements:

**4.1.1 Security**

The system will be designed with a level of security that is appropriate for the sensitive information. Therefore, the virtual private server (VPS) will be more than enough to ensure that everything is secured, and the customers’ privacy desirable is achieved. Since the system does not handle high security level information such as credit card and credentials, password protecting the database. So only the users should have access to it should suffice.

**4.1.2 Reliability**

Regular back-ups will be made. So, there will be minimal data loss if the system stops working somehow. Thorough testing and truly challenging the system’s limits by all four team members will also ensure its reliability.

**4.1.3 Maintainability**

The system will be regularly maintained by a group member each month to ensure that the system is running smoothly. 

**4.1.4 Portability**

This system will be compatible with both android and IOS with time, and it will be designed with more than those two operating systems in mind. In the case that time does not allow the system will shift focus to whichever operating system the customer prioritizes

**4.1.5 Capacity**

The VPS can store high amounts of data and the system will only store surveillance camera footage up to a week back. That data is what will occupy most of the system capacity. Whereas its functionality and everything else in the system falls shy in comparison and would not need a lot of space to operate

**4.1.6 Usability**

The system will be designed in a way that makes it intuitive and very easy to use without filling it with extra and aimless functionality. So, it will be designed to be direct, clear, and straight to the point.

**4.1.7 Reusability**

The system should be designed in a way that allows the database to be re-used regularly for the various silent auctions that the organization shall hold.

4.2 Constraints

**4.2.1 Budget**

Since physical items must be purchased. Such as the surveillance cameras and the private virtual server. Furthermore, the wages per hour for each group member is also considered a constraint

**4.2.2 Lack of expert knowledge**

 is a limitation, for the reason that some functionalities of the system require skills and expertise in topic that have high learning curves

**4.3.3 Time**

Because urgency is required, and some parts of development have to be rushed. Which means some compromises are inevitable

5. Use Cases

5.1 Use Case Model:

**Diagram

Description automatically generated**

5.2 Use Cases:

**5.2.1 Register**

**Title: Register**

**Description:** The client registers a new user in the system.

**Actors:**

1. Client

2. Application Administrator

**Pre-conditions:**

1. The client must be unregistered.

**Post-conditions:**

1. Client is registered in the system.

**Basic flow:**

1. The system asks the client to enter his personal information.
2. The system asks the client to enter his email and password.
3. The system checks if the user is registered before.
4. The system registers the client in the system.

**Alternative flow:**

3a) The client is already registered in the system.

     4) The system prompts an error message.

**5.2.2 Login**

**Title: Login**

**Description:** The client logs in to the system.

**Actors:**

1. Client

2. Application Administrator

**Pre-conditions:**

1. If unregistered, the client must register.
2. If registered, the client must log in.

**Post-conditions:**

1. Client is logged in.

**Basic flow:**

1. The system validates the client’s username and password and logs him into the system
2. The system displays the main user interface

**Alternative flow:**

 1a) The name does not match the password

  2) The system prompts an error message

**5.2.3 Access Camera Feed**

**Title:** Access Camera Feed

**Description** The client can access either the live feed or pre-recorded feed in case of an emergency

**Actors:**

1. Client

2. Application Administrator

**Pre-conditions:**

1. The client must be logged in.

**Post-conditions:**

1. System updates the new information and works normally.

**Basic flow:**

1) The client accesses camera feed.

2) The system asks which feed to access live or recorded

3) The client accesses the live feed.

4) The client leaves.

5) The system updates the new information and works normally.

**Alternative flow:**

3a) The client accesses recorded camera feed

4) The client leaves.

5) The system updates the new information and works normally.

**5.2.4 Send Alert**

Title: Send Alert

**Description**: The system detects an emergency, so it sends an alert to the parents.

**Actor:** Child

**Preconditions**:

1. Client is Logged in and system
2. Surveillance camera is monitoring the child.

**Post conditions:**

1. The parents have been notified about the emergency.

**Basic flow:**

1) The system detects an emergency

2) The system notifies the parents of the emergency.

3) The parents confirm to the system that they received the notification.

4) The system updates the new information and keeps monitoring the emergency.

**Alternative flow:**

3a) The parents do not confirm to the system that they received the notification.

4) The system notifies the local authority.

5) The system notifies the parent that the local authority was notified.

6) The system updates the new information and keeps monitoring the emergency.

**5.2.5 Contact Parents**

**Title:** Contact Parents

**Description:** System contacts both parents on the state of the child

**Actor:**

1. Client

2. Application Administrator

**Preconditions:**

1. Client is logged in
2. System has the client’s contact information.

**Post conditions:**

1. System has contacted the parents.

**Basic flow:**

1) The system detects the state of the child.

2) The system contacts the parents on the state of their child.

3)  System updates the new information and works normally.

**Alternative flow:**

1a) The system detects an emergency

2) The system alerts the parents of the emergency

3) The Parents confirm receiving the notification

4) System updates the new information and works normally.

**5.2.6 Contact Authority**

**Title:** Contact Authority

**Description**: System contacts the local authority

**Actors:**

1. Client

2. Application Administrator

**Preconditions:**

1. Client is logged in
2. System has the local authority contact information.

**Post conditions:**

1. System has contacted the local authority.

**Basic flow:**

1) The system detects an emergency.

2) The system contacts the parents on the state of their child.

3) The parents do not confirm receiving the message

4) The system contacts the local authority.

5) The system informs the parent that the local authority was notified.

6)  The system updates the new information and keeps monitoring the emergency.

6 Data Dictionary:

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Field Length** | **Description** |
| Email | Public String | 30 characters | Client email |
| PhoneNumber | Private Integer | 15 characters | Client phone number |
| Username | Public String | 15 characters | Client’s username |
| Password | Private String | 20 characters | Client’s login password |
| Local\_Authority\_Num | Public Constant Integer | 15 characters | Phone number to contact local authority |
| DevEmail | Public Constant String | 30 characters | The developers’ email |

7 Tools:

| **Tools** | **Purpose** |
| --- | --- |
| Plant UML | Drawing Diagrams |
| Microsoft Word | Processing Words |
| Visual Studio Code | Prototyping and Development |
| Github | Project Management |