|  |  |
| --- | --- |
|  | **MINISTRY OF EDUCATION AND TRAINING** |

**FPT UNIVERSITY**

|  |  |
| --- | --- |
| PRC391 Project Document | |
| **E-Pharmacy** | |
| **Group 4 - IS** | |
| **Group members** | Lê Hoàng Hiếu - Leader - SE130560  Nguyễn Thành Đạt - Member - SE130272  Nguyễn Công Thành - Member - SE130396 |
| **Supervisor** | Hồ Hoàn Kiếm |
| **Ext. Supervisor** | N/A |
| **Project Code** | EP |

– **Ho Chi Minh City, 12 November, 2020** –

**Table of Contents**

[**A. Introduction 2**](#_heading=h.206ipza)

[1. Project Information 2](#_heading=h.3cqmetx)

[2. Introduction 3](#_heading=h.1rvwp1q)

[3. Current Situation 3](#_heading=h.4bvk7pj)

[4. Problem definition 3](#_heading=h.2r0uhxc)

[5. Proposed solution 3](#_heading=h.1664s55)

[5.1 Features 4](#_heading=h.3q5sasy)

[5.2. Benefits and trade-offs 4](#_heading=h.25b2l0r)

[6. Role and responsibility 5](#_heading=h.kgcv8k)

[**B. Request Specification 5**](#_heading=h.1jlao46)

[1. Functional Requirements 5](#_heading=h.43ky6rz)

[2. Hardware/Software Requirements 6](#_heading=h.2iq8gzs)

[**C. Software Requirement Specification 7**](#_heading=h.xvir7l)

[1. User Requirement Specification 7](#_heading=h.3hv69ve)

[1.1. Unauthorized user Requirement 7](#_heading=h.1x0gk37)

[1.2. Authorized user Requirement 7](#_heading=h.4h042r0)

[2. System Requirement Specification 8](#_heading=h.2w5ecyt)

[2.1. System Overview Use Case 8](#_heading=h.1baon6m)

[2.2. List of Use Case 8](#_heading=h.3vac5uf)

[2.2.a. Unauthorized user requirement 8](#_heading=h.2afmg28)

[2.2.a.1. Login 8](#_heading=h.pkwqa1)

[2.2.a.2. Register 10](#_heading=h.39kk8xu)

[2.3.b. Authorized user requirement 12](#_heading=h.1opuj5n)

[2.2.b.1. Log out 12](#_heading=h.48pi1tg)

[2.2.b.2. Update profile 14](#_heading=h.2nusc19)

[2.2.b.3. Change password 15](#_heading=h.1302m92)

[2.2.b.4. Show list prescriptions 17](#_heading=h.3mzq4wv)

[2.2.b.5. Show prescription detail 18](#_heading=h.2250f4o)

[2.2.b.6. Delete prescription 20](#_heading=h.haapch)

[2.2.b.7. Detect prescription 21](#_heading=h.319y80a)

[2.2.b.8. Add prescription 22](#_heading=h.1gf8i83)

[**D. Software Design Description 24**](#_heading=h.40ew0vw)

[1. Cloud Service 24](#_heading=h.2fk6b3p)

[1.1. Google Cloud SQL (MySQL) 24](#_heading=h.upglbi)

[1.2. Firebase (ML Kit and Cloud Messaging) 24](#_heading=h.1tuee74)

[1.3. Vultr (IaaS - Deploy Backend) 24](#_heading=h.4du1wux)

[2. Entity Relationship Diagram (ERD) 25](#_heading=h.2szc72q)

[3. Entity Description 26](#_heading=h.184mhaj)

[3.1. Account 26](#_heading=h.3s49zyc)

[3.2. Prescription 26](#_heading=h.279ka65)

[3.3. Drug 27](#_heading=h.meukdy)

[3.4. Prescription Detail 27](#_heading=h.36ei31r)

[3.5. Category 28](#_heading=h.1ljsd9k)

[3.6. Supplier 28](#_heading=h.45jfvxd)

[3.7. Unit 29](#_heading=h.2koq656)

[3.8. Hospital 29](#_heading=h.zu0gcz)

[3.9. Doctor 30](#_heading=h.3jtnz0s)

[4. System Architectural Design 31](#_heading=h.1yyy98l)

[5. Interface 32](#_heading=h.4iylrwe)

[5.1. Login screen 32](#_heading=h.2y3w247)

[5.2. Choose Google Account Dialog 35](#_heading=h.1d96cc0)

[5.3. Sign Up Screen 38](#_heading=h.3x8tuzt)

[5.4. Home Screen 41](#_heading=h.2ce457m)

[5.5. Gallery Picking Screen 44](#_heading=h.rjefff)

[5.6. Scan Result Screen 47](#_heading=h.3bj1y38)

[5.7. Prescription List Screen 50](#_heading=h.rhzx59y0wn2l)

[5.8. Prescription Detail Screen 53](#_heading=h.1qoc8b1)

[5.9. Account Option Bottom Sheet Dialog 56](#_heading=h.4anzqyu)

[5.10. Edit Profile Screen 59](#_heading=h.2pta16n)

[5.11. Change Password Screen 62](#_heading=h.14ykbeg)

[6. Table Relationship 65](#_heading=h.3oy7u29)

# A. Introduction

## 1. Project Information

- Project Name: **E-Pharmacy**

- Project Code: **EP**

- Project Type: **Mobile Application**

- Start Date: 15/09/2020

- End Date: 12/11/2020

## 2. Introduction

Currently, when people have a medical examination. They will receive the result and a drug prescription on paper. With this application, they can easily manager their prescriptions received from hospital without worrying about losing their prescriptions on paper.

## 3. Current Situation

However, when you have more medical examinations, the prescriptions start to pile up and become messy. When you want to look for the old prescription to refill your drug, the problem starts to be noticeable. Looking for the old prescription in a bunch of prescriptions is troublesome. Moreover, the record is only available on paper so it will be inconvenient to carry it around.

## 4. Problem definition

There are many drawbacks of current situation:

* **Hard to keep track of old prescriptions**
* **Inconvenient on paper record**

## 5. Proposed solution

Our proposed solution is to build a system called **E-Pharmacy** to address the current situations so that people can store prescriptions on their phone, manage their list more consistently and it will be conveniently available for them at all times. The E-Pharmacy system includes a mobile application with the following functions:

### 5.1 Features

* Mobile User application:
  + Login: The unauthorized user must login to manage their prescription
  + Login with Google Account: The unauthorized user use google account to login to manage their prescription
  + Register: The unauthorized user can register new account
  + Register with Google Account: The unauthorized user can register new account with Google Account
  + Logout: The authorized user can logout of their account
  + Scan prescription: The authorized user can scan the prescription with camera or uploaded image
  + Create prescription: the authorized user can create and add scanned prescription to their list of prescriptions
  + View prescription list: The authorized user can view their prescription list
  + View Prescription detail: The authorized user can view their prescription detail
  + Delete Prescription: The authorized user can delete their prescription
  + Edit Profile: the authorized user can edit their profile information
  + Change password: the authorized user can change their account password

### 5.2. Benefits and trade-offs

* Benefits:
  + You can store and manage your prescriptions more consistently
* Trade-offs:

○ Request can take a long time when the internet is slowly

○ Depend on Quality of captured/uploaded image

## 6. Role and responsibility

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Full Name** | **Role** | **Position** | **Contact** |
| 1 | Hồ Hoàn Kiếm | Project Manager | Supervisor | kiemhh@fpt.edu.vn |
| 2 | Lê Hoàng Hiếu | Developer | Leader | hieulhse130560@fpt.edu.vn |
| 3 | Nguyễn Thành Đạt | Developer | Member | datntse130272@fpt.edu.vn |
| 4 | Nguyễn Công Thành | Developer | Member | thanhncse130396@fpt.edu.vn |

# B. Request Specification

## 1. Functional Requirements

Following is a list of functionalities of the system. More functionalities that you find appropriate can be added to this list.

* If a wrong password is given, that system will message error for you.
* Unauthorized user must have a valid username and password to login to the system
* Unauthorized user may use Google account as an alternative sign in option
* Unauthorized user must login before he/she can use app’s features
* Authorized user can manage his/her account profile, prescriptions.
* When they want to change their password. They must provide an old password first.
* Prescription can only be detected by scanning using camera or by uploading existed image.

## 2. Hardware/Software Requirements

**For Server**

|  |  |  |
| --- | --- | --- |
| **Hardware** | **Minimum Requirement** | **Recommended** |
| **Internet Connection** | Cable, Wi-Fi (8 Mbps) | Cable, Wi-Fi (50 Mbps or more) |
| **Operation System** | Ubuntu server 16.04 | Ubuntu server 20.04 |
| **Computer Processor** | Intel® Xeon ® 1.4GHz | Intel® Xeon ® Quad Core (12M Cache, 2.50 GHz) |
| **Computer memory** | 1 GB RAM | 4 GB RAM or more |
| **Storage space** | 2.5 GB | 25 GB or more |

**For Mobile Device**

|  |  |  |
| --- | --- | --- |
| **PC** | **Minimum requirements** | **Recommended** |
| **Internet Connection** | Cellular network, Wi-Fi (1 MBps) | Cellular network, Wi-Fi (8 MBps) |
| **Operating System** | Android 8.0 | Android 10.0 or above |
| **Processor** | Snapdragon 435 | Snapdragon 665 or above |
| **Memory** | 1 GB RAM | 4 GB RAM or more |
| **Free space** | 50MB | 100MB or more |

**For Software**

|  |  |  |
| --- | --- | --- |
| **Software** | **Name / Version** | **Description** |
| **Operation System** | Ubuntu desktop 20.04  Windows 10 Pro | Operating system and platform for development |
| **Environment** |  | Specification for developing server and web application |
| **Modeling tool** | Visio | Used to design diagram |
| **IDE** | Intellij 19.3  Android Studio | Programming tools |
| **DBMS** | MySQL Workbench 8.0 CE | Used to create & manage the database for system |
| **Source control** | Git | Used for source control |
| **Android version** | Android 8.0 | Testing mobile environment |

# C. Software Requirement Specification

## 1. User Requirement Specification

### 1.1. Unauthorized user Requirement

Unauthorized users don't have access to use some functions of the system. To use all functions, unauthorized users must login first. Unauthorized user can use following functions:

* Login
* Register

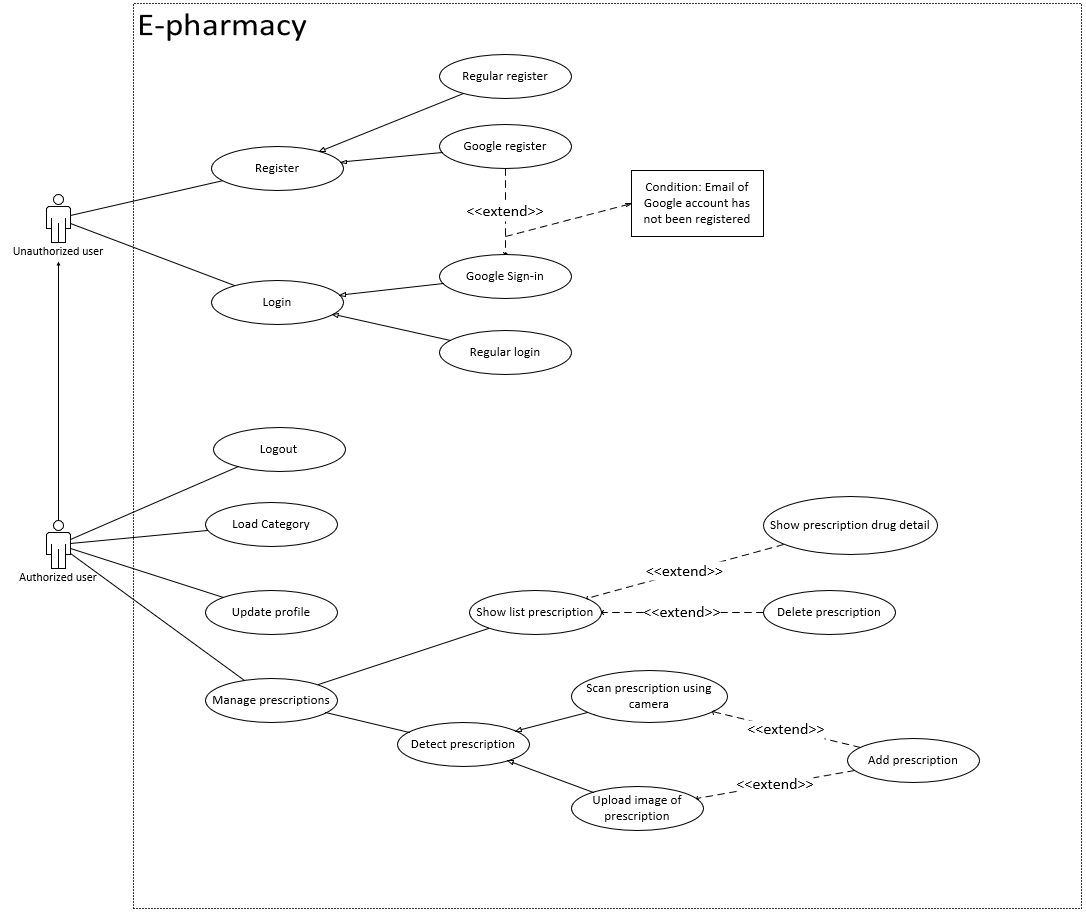
### 1.2. Authorized user Requirement

Authorized user is who already login successfully into the system. Authorized user can use below functions:

* Logout
* Scan & Detect Prescription
* Add Prescription
* Delete Prescription
* Update profile
* Load category

## 2. System Requirement Specification

### 2.1. System Overview Use Case



### 2.2. List of Use Case

#### 2.2.a. Unauthorized user requirement

##### 2.2.a.1. Login

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EP\_UC\_A1** | | | |
| **Use Case No.** | EP\_UC\_A1 | **Use Case Version** | 1.0 |
| **Use Case Name** | Login | | |
| **Author** | DatNT | | |
| **Date** | 12/11/2020 | **Priority** | High |
| **Actor:**   * Unauthorized user.   **Summary:**   * This use case shows how unauthorized users will login to system.   **Goal:**   * Authorize user.   **Triggers:**   * Unauthorized user sends login command.   **Preconditions:**   * User has not logged in to the system. * User’s devices are connected to the Internet   **Post conditions:**   * Success: User login to the system and move to corresponding screen * Fail: N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | Unauthorized are currently staying at login screen | Application provides two sign-in options for user: Google Sign-in or Sign in with their own username and password. Following is what system will display:  - Google Sign-in: Google icon  - Sign in with username and password: 2 text fields and 1 Sign in button | | 2.1 | Google Sign-in: Unauthorized user press on Google icon and choose their account | User will be navigated to the home screen of application | | 2.2 | Sign in with username and password: Fill in username and password fields, then press log in button | User will be navigated to the home screen of application  [Alternative 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | Wrong username or password. | Display message: “Incorrect username/password”. |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * When authorization is completed, system must generate token for authorization corresponding logged in user. * After login to system, user will be navigated to the home screen. | | | |

##### 2.2.a.2. Register

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EP\_UC\_A2** | | | |
| **Use Case No.** | EP\_UC\_A2 | **Use Case Version** | 1.0 |
| **Use Case Name** | Register | | |
| **Author** | HieuLH | | |
| **Date** | 12/11/2020 | **Priority** | Normal |
| **Actor:**   * Unauthorized user.   **Summary:**   * This use case allows guest to register new account.   **Goal:**   * New account is created by guest.   **Triggers:**   * Guest sends the register command.   **Preconditions:**   * No account is currently logged in. * User’s devices are connected to the Internet   **Post conditions:**   * Success: Guest has an account. * Fail: N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | Guest are currently staying at login screen | Application provides two registration options: Google Sign-in and Signup Tab  - Google Sign-in: Google icon  - Signup tab: In signup tab, there are 6 input text fields, 1 combo box, 1 date selector and 1 signup button | | 2.1 | Google Sign-in: Guest press on Google icon and choose their account | User will be navigated to the home screen of the application | | 2.2.1 | Guest taps on signup tab | System requires identity information from guest:  - Username: editable text field  - Name: editable text field  - Email: editable text field  - Password: editable text field  - Re-password: editable text field  - Address: editable text field  - Birth date: date selector  - Gender: combo box | | 2.2.2 | Guest inputs information. |  | | 2.2.3 | Guest sends command to register to system | Application checks the validation of the inputted fields.  [Alternative 1]  [Alternative 2] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | Inputted fields are all valid | User will be navigated to the home screen of the application | | 2 | One of the inputted fields is invalid | Re-input field that is invalid |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * After registration succeeded, Login action is automatically triggered, a token is sent back from back-end for this user | | | |

#### 2.3.b. Authorized user requirement

##### 2.2.b.1. Log out

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EP\_UC\_B1** | | | |
| **Use Case No.** | EP\_UC\_B1 | **Use Case Version** | 1.0 |
| **Use Case Name** | Logout | | |
| **Author** | ThanhNC | | |
| **Date** | 12/11/2020 | **Priority** | Normal |
| **Actor:**   * Authorized User.   **Summary:**   * This use case describes how Authorized user logout.   **Goal:**   * Clear all the tokens and sessions.   **Triggers:**   * Authorized User sends logout command.   **Preconditions:**   * User has logged in application.   **Post conditions:**   * Success: All sessions and tokens will be cleared. Now, the user will be an unauthorized user and move to the login screen. * Fail: N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User sends a logout command. | Clear session and token, change to login display. |   **Alternative Scenario:** N/A  **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * All session and token must be cleared. Change current screen to login screen. | | | |

##### 2.2.b.2. Update profile

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EP\_UC\_B2** | | | |
| **Use Case No.** | EP\_UC\_B2 | **Use Case Version** | 1.0 |
| **Use Case Name** | Update profile | | |
| **Author** | ThanhNC | | |
| **Date** | 12/11/2020 | **Priority** | Normal |
| **Actor:**   * Unauthorized user.   **Summary:**   * This use case allows authorized user to update his/her profile.   **Goal:**   * Updated user profile.   **Triggers:**   * Authenticate user taps “Edit profile” button.   **Preconditions:**   * User has logged in application. * User’s devices are connected to the Internet.   **Post conditions:**   * Success: User updates his/her profile successfully * Fail: N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | Authorized user taps “Edit profile” button | System shows current user information:  - Username: read-only text field  - Name: editable text field  - Male: gender combo box  - Email: read-only text field  - Birth date: date selector  - Address: editable text field | | 2 | User fills in their new information |  | | 3 | User taps “Confirm” button | System toasted a short message “Update successfully” |   **Alternative Scenario:** N/A   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * All of user information is updated, and new information is displayed in Edit Profile screen | | | |

##### 2.2.b.3. Change password

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EP\_UC\_B3** | | | |
| **Use Case No.** | EP\_UC\_B3 | **Use Case Version** | 1.0 |
| **Use Case Name** | Change password | | |
| **Author** | DatNT | | |
| **Date** | 12/11/2020 | **Priority** | Normal |
| **Actor:**   * Authorized user.   **Summary:**   * This use case describes authorized user who want to change password.   **Goal:**   * Changed user’s password.   **Triggers:**   * Authorized user taps “Change password” button in “Edit profile” screen.   **Preconditions:**   * User has logged in application. * Logged in account is not Google account. * User’s devices are connected to the Internet.   **Post conditions:**   * Success: Toasted message “Password updated”. * Fail: N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | Authorized user taps “Change password” button on Edit Profile screen | System requires user’s old and new passwords:  - Old password: editable text field  - New password: editable text field  - Confirm new password: editable text field | | 2 | User fills in 3 mentioned text fields |  | | 3 | User taps “Confirm” button | System toasted a short message “Password updated” |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * New password is updated and user is able use new password for logging in | | | |

##### 2.2.b.4. Show list prescriptions

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EP\_UC\_B4** | | | |
| **Use Case No.** | EP\_UC\_B4 | **Use Case Version** | 1.0 |
| **Use Case Name** | Show list prescriptions | | |
| **Author** | HieuLH | | |
| **Date** | 12/11/2020 | **Priority** | Normal |
| **Actor:**   * Authorized user   **Summary:**   * This use case describes how user is able to see his/her added prescriptions.   **Goal:**   * List of prescriptions is showed.   **Triggers:**   * User taps Prescriptions List button   **Preconditions:**   * User has logged in to the system. * User’s devices are connected to the Internet.   **Post conditions:** N/A  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User taps Prescription List button | Application shows list of added prescriptions  [Alternative 1] |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** | | 1 | Empty prescription list | Application shows red message “List is empty” |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:** N/A | | | |

##### 2.2.b.5. Show prescription detail

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EP\_UC\_B5** | | | |
| **Use Case No.** | EP\_UC\_B5 | **Use Case Version** | 1.0 |
| **Use Case Name** | Show list prescription detail | | |
| **Author** | DatNT | | |
| **Date** | 12/11/2020 | **Priority** | Normal |
| **Actor:**   * Authorized user   **Summary:**   * This use case describes how user see his/her specific prescription.   **Goal:**   * Selected prescription’s detail is displayed.   **Triggers:**   * User taps on any prescription on Prescription List screen   **Preconditions:**   * User has logged in the application. * Prescription List screen is displayed and is not empty.   **Post conditions:**   * Success: Prescription detail is displayed. * Fail: N/A.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User taps on a prescription in Prescription List screen | Application shows selected prescription’s detail:  - ID: read-only text field  - Created date: read-only text field  - List of drugs in this prescription  - Delete button |   **Alternative Scenario:** N/A   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:** N/A | | | |

##### 2.2.b.6. Delete prescription

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EP\_UC\_B6** | | | |
| **Use Case No.** | EC\_UC\_B6 | **Use Case Version** | 1.0 |
| **Use Case Name** | Delete prescription | | |
| **Author** | DatNT | | |
| **Date** | 12/11/2020 | **Priority** | Normal |
| **Actor:**   * Authorized user   **Summary:**   * This use case describe how user removes his/her prescription.   **Goal:**   * Prescription is deleted   **Triggers:**   * User taps Delete button   **Preconditions:**   * Prescription detail is displayed   **Post conditions:**  **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User taps on Delete button | User is navigated to Prescription List screen and deleted prescription is not displayed |   **Alternative Scenario:**   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * Deleted prescription is removed and cannot be displayed in Prescription List screen | | | |

##### 2.2.b.7. Detect prescription

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EP\_UC\_B7** | | | |
| **Use Case No.** | CS\_CM\_UC\_C4 | **Use Case Version** | 1.0 |
| **Use Case Name** | Detect prescription | | |
| **Author** | DatNT | | |
| **Date** | 12/11/2020 | **Priority** | High |
| **Actor:**   * Authorized user.   **Summary:**   * This use case describes how user’s prescription drug is detected.   **Goal:**   * List of detected drugs.   **Triggers:**   * Admin captured image by camera or uploaded existing image.   **Preconditions:**   * User has logged in application. * User’s devices are connected to the Internet.   **Post conditions:**   * Success: List of detected drugs is displayed on screen. * Fail: N/A   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1.1 | User taps Camera button | Camera devices is activated | | 1.2 | User taps Upload Image button | Gallery is displayed | | 2 | User capture image or confirm image | User is navigated to home screen, application starts detecting drugs and show list of detected drugs |   **Alternative Scenario:** N/A   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:** N/A | | | |

##### 2.2.b.8. Add prescription

|  |  |  |  |
| --- | --- | --- | --- |
| **USE CASE – EP\_UC\_B8** | | | |
| **Use Case No.** | EP\_UC\_B8 | **Use Case Version** | 1.0 |
| **Use Case Name** | Add prescription | | |
| **Author** | ThanhNT | | |
| **Date** | 12/11/2020 | **Priority** | High |
| **Actor:**   * Authorized user.   **Summary:**   * This use case describes how user adds new prescription.   **Triggers:**   * User captured or confirmed image.   **Preconditions:**   * List of detected drugs is displayed   **Post conditions:**   * New prescription is added.   **Main Success Scenario:**   |  |  |  | | --- | --- | --- | | **Step** | **Actor Action** | **System Response** | | 1 | User captured or confirmed image | Application shows list of detected drugs | | 2 | User taps Confirm button | New prescription is added and user is navigated to home screen |   **Alternative Scenario:** N/A   |  |  |  | | --- | --- | --- | | **No** | **Cause** | **System Response** |   **Exceptions:** N/A  **Relationships:** N/A  **Business Rules:**   * New prescription is added and can be displayed on Prescription List screen | | | |

# D. Software Design Description

## 1. Cloud Service

### 1.1. Google Cloud SQL (MySQL)

* MySQL:
* Advantages:
* Support both SQL and NoSQL
* Best suited for Ubuntu server
* The cost is lower than SQL Server
* Can use the full features if accessing IP in local software
* Disadvantages:
* Not so fast
* Do not support tool in platform
* Quite hard to use for beginner due to configuration

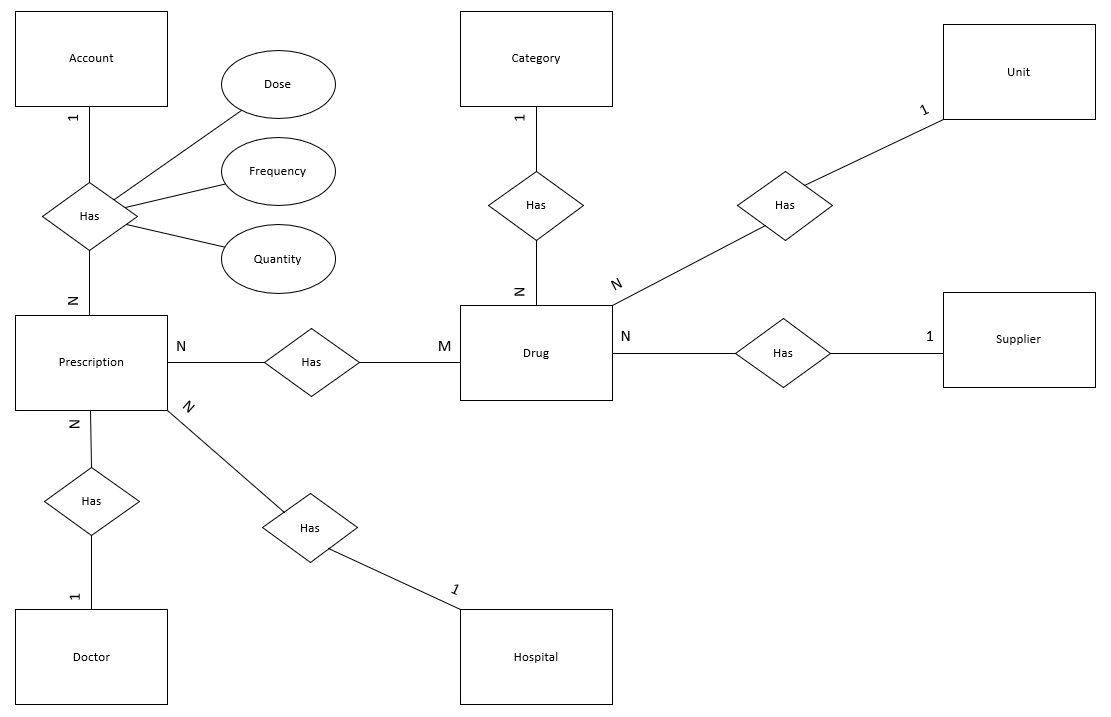
### 1.2. Firebase (ML Kit and Cloud Messaging)

* ML Kit:
* Advantages:
* Easy to approach
* Easy to integrate with Native Android project
* Support library to interact
* Included in Firebase Console with many other functions
* Strongly support by Google
* Disadvantages:
* Requires money to analyze other languages than English
* Cloud Messaging:
* Advantages:
* Easy to integrate with both Native Android and Spring Boot
* Support library to interact
* Reliable in sending notifications
* Included in Firebase Console with many other functions
* Strongly support by Google
* Disadvantages:
* First step of setting up may meet some struggles

### 1.3. Vultr (IaaS - Deploy Backend)

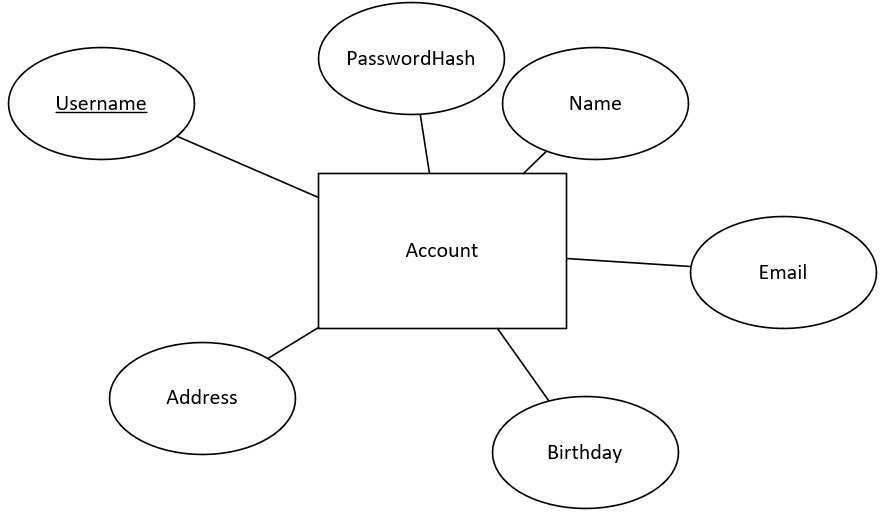
* Ubuntu server 20.04:
* Advantages:
* Open source, suitable for education project
* Many documents and references
* Low renting price
* High customization
* Widely used by software developing community
* System requires from low machine
* Disadvantages:
* Lack of UI
* Configure by self from beginning to the finish line
* Hard to use for beginner

## 2. Entity Relationship Diagram (ERD)

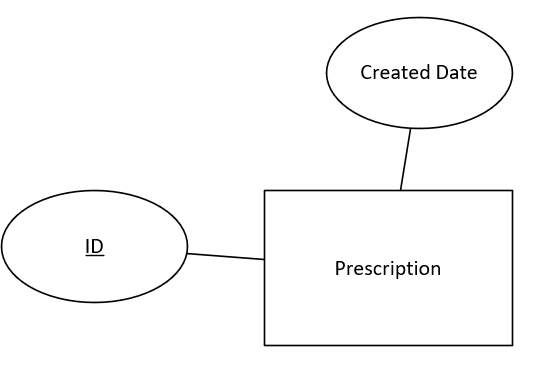


## 3. Entity Description

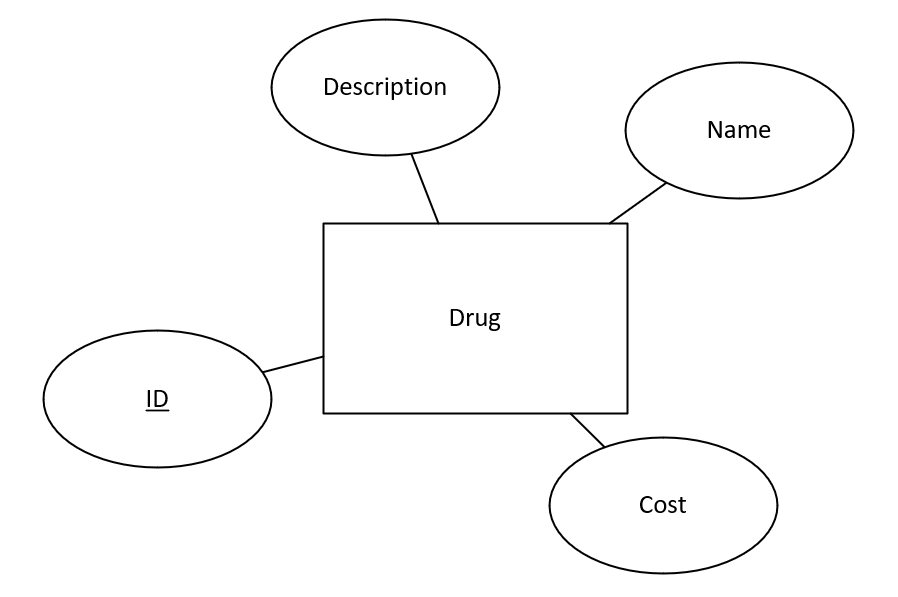
### 3.1. Account



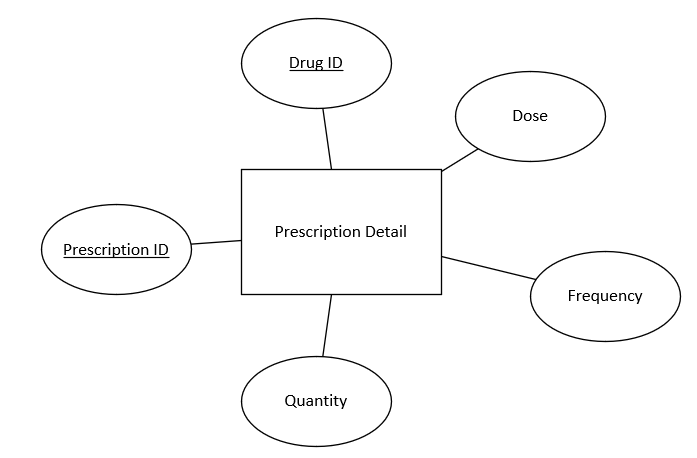
### 3.2. Prescription



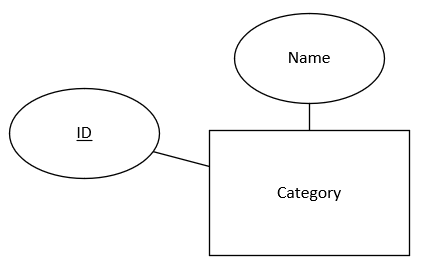
### 3.3. Drug



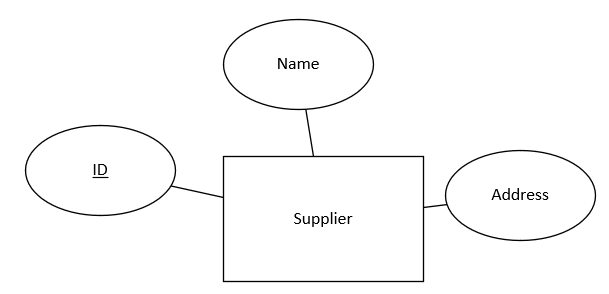
### 3.4. Prescription Detail



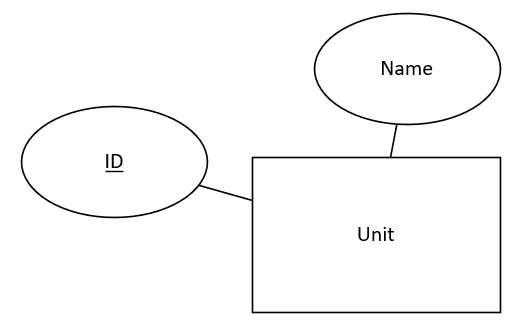
### 3.5. Category



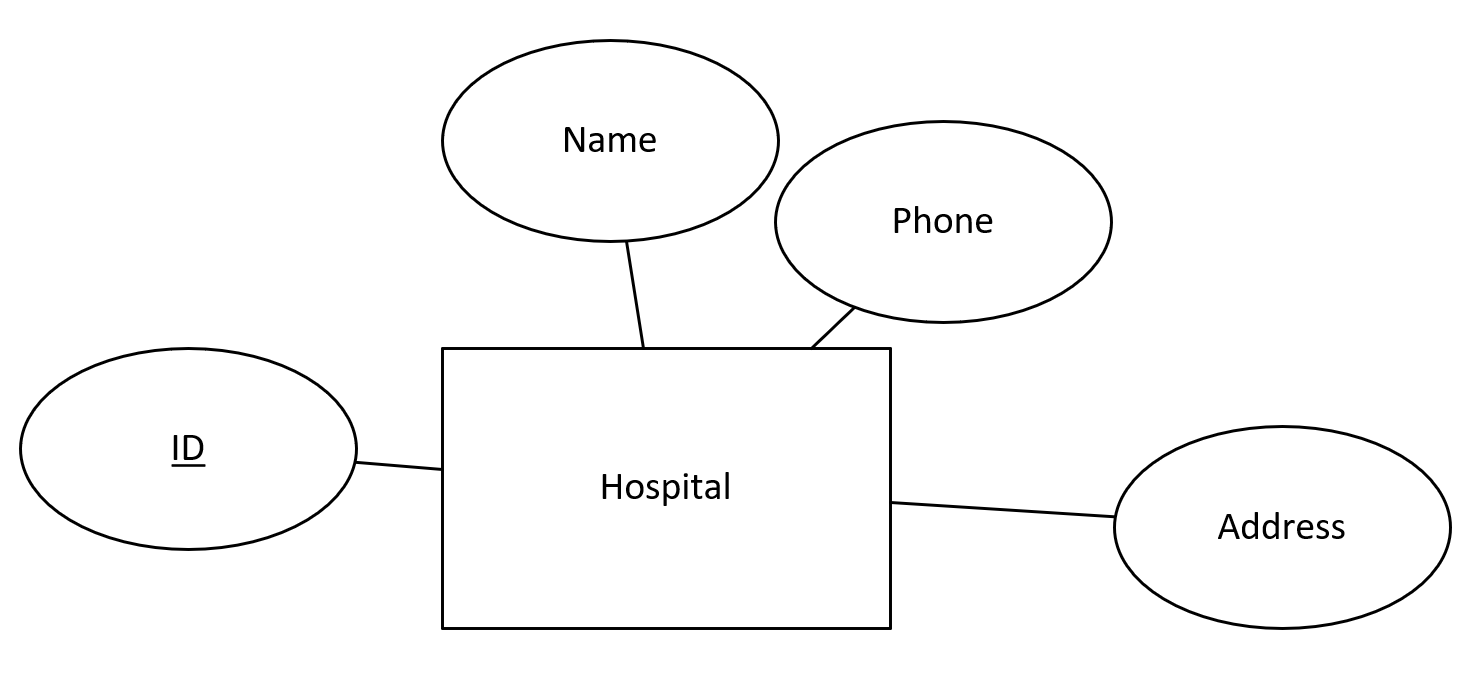
### 3.6. Supplier



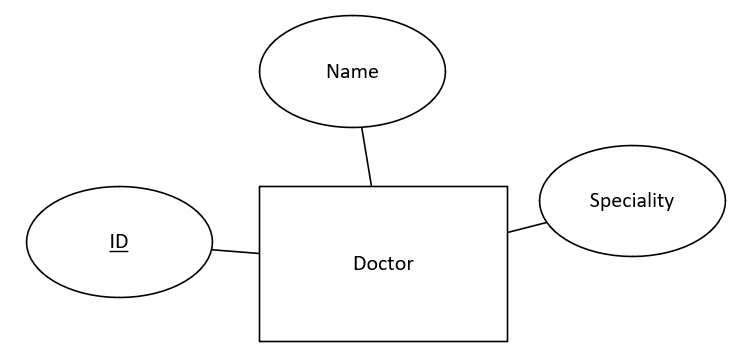
### 3.7. Unit



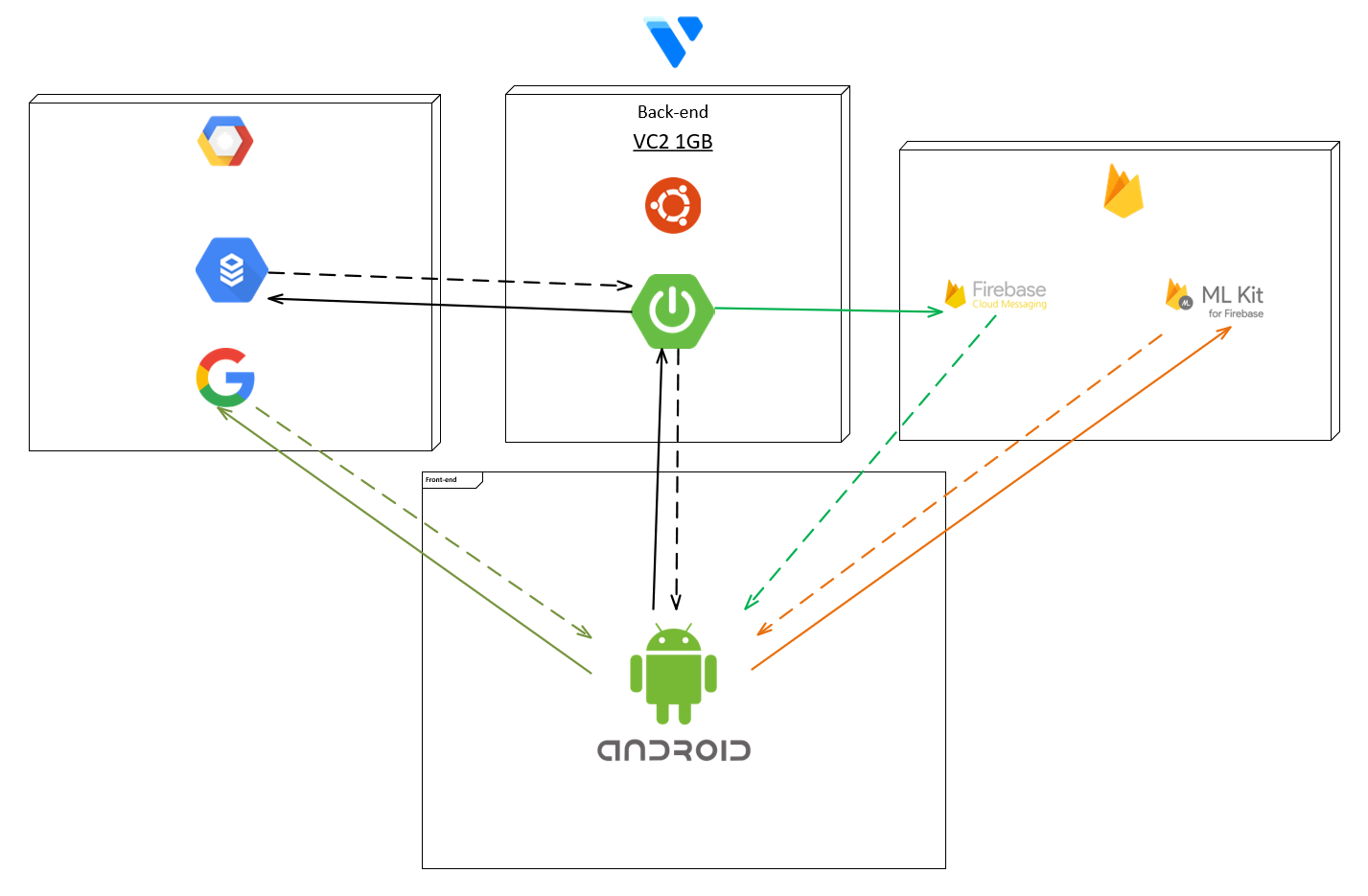
### 3.8. Hospital



### 3.9. Doctor

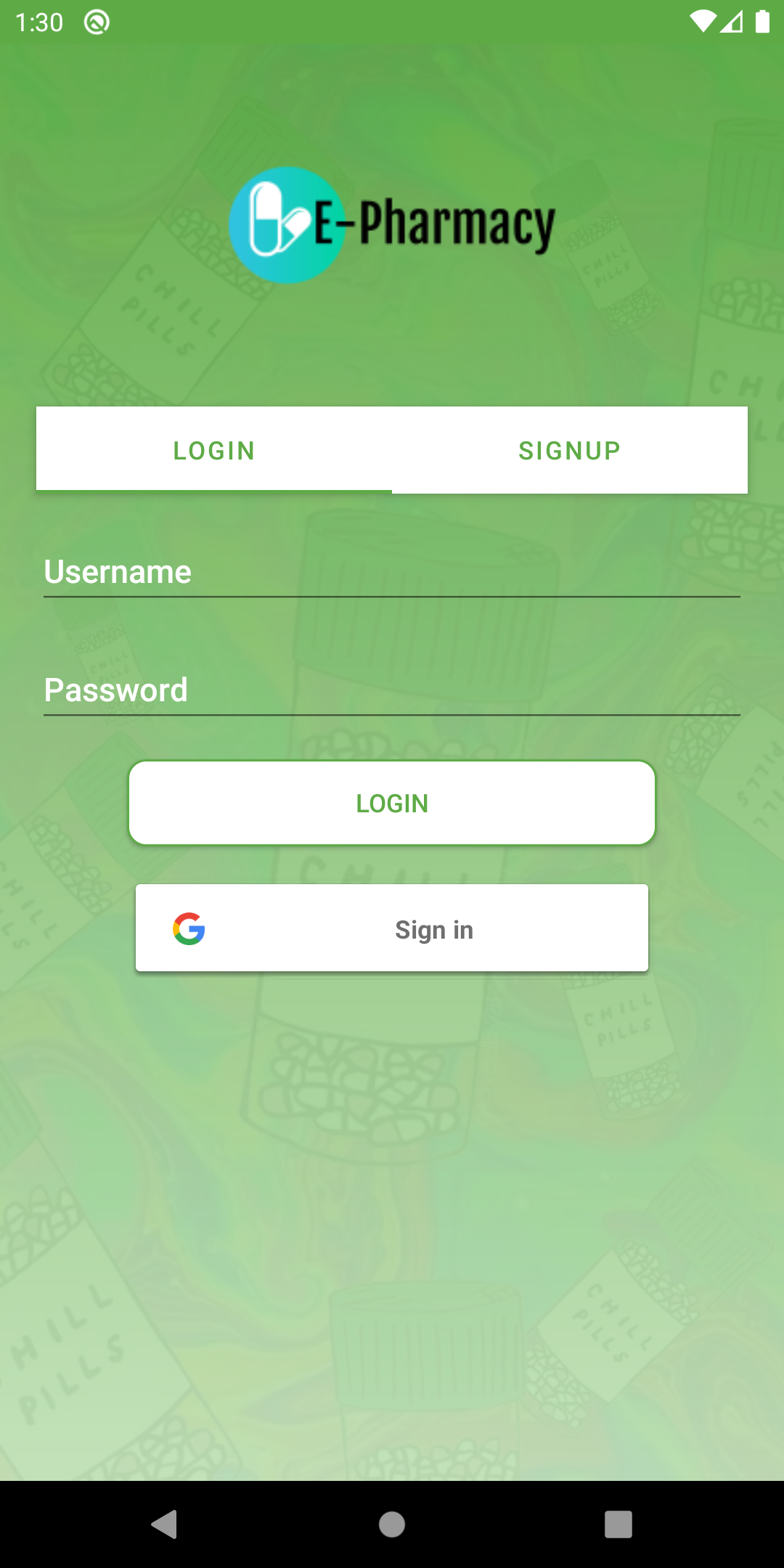


## 4. System Architectural Design



## 5. Interface

### 5.1. Login screen



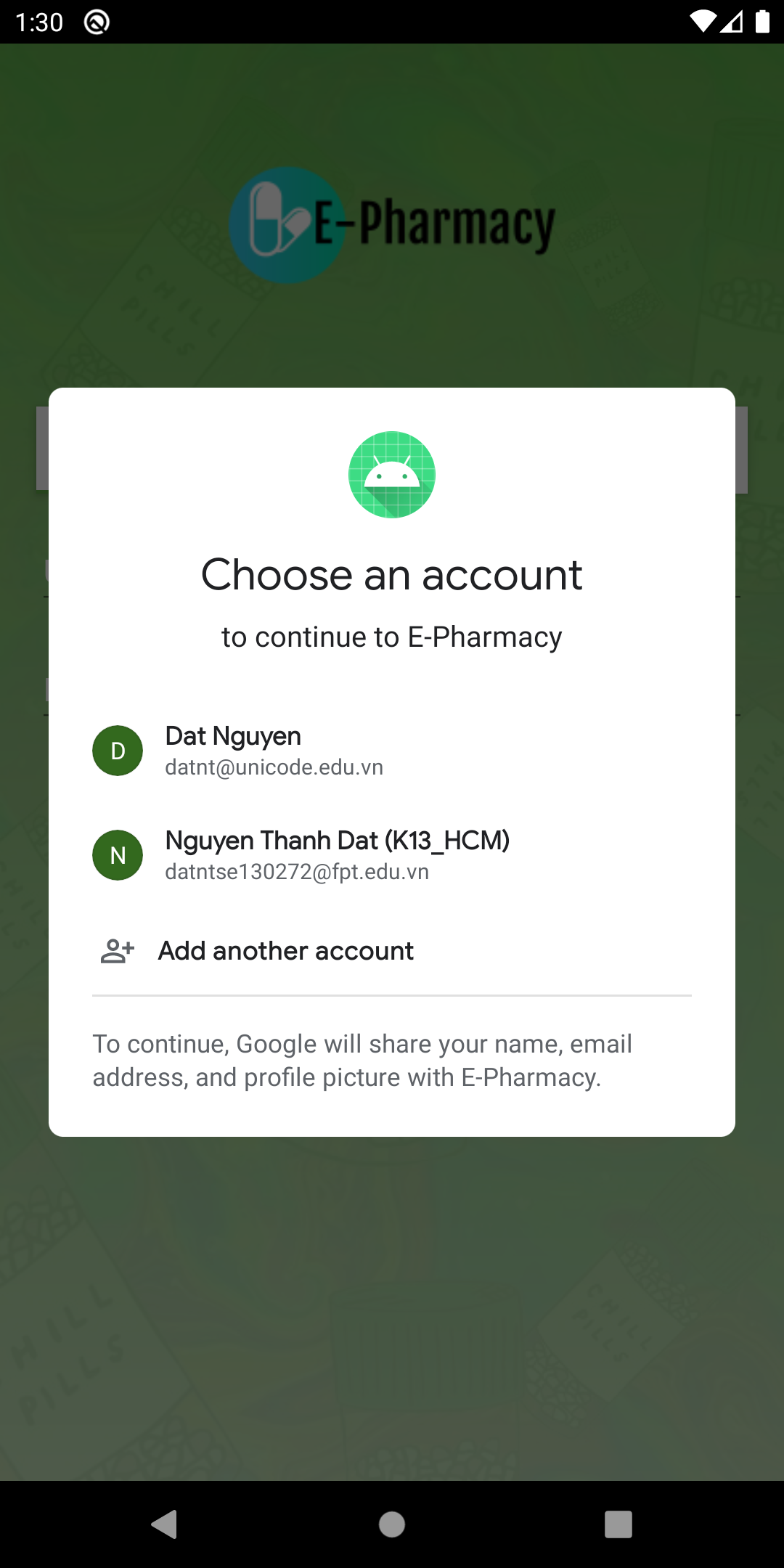
***Fields***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |
| 1 | Username | Username of the account | No | Yes | EditText | String | 32 |
| 2 | Password | Password of the account | No | Yes | EditText | String | 60 |

***Buttons/Hyperlinks***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 3 | Login | Login to system | All fields must be input | Users will be navigated to the home screen. |
| 4 | Login with Google | Login to system or register new account with your google account | N/A | Google account choosing Dialog will pop up |
| 5 | SignUp Tab | Navigate to Sign Up Tab | N/A | User will be navigated to the Sign Up Tab |

### 5.2. Choose Google Account Dialog



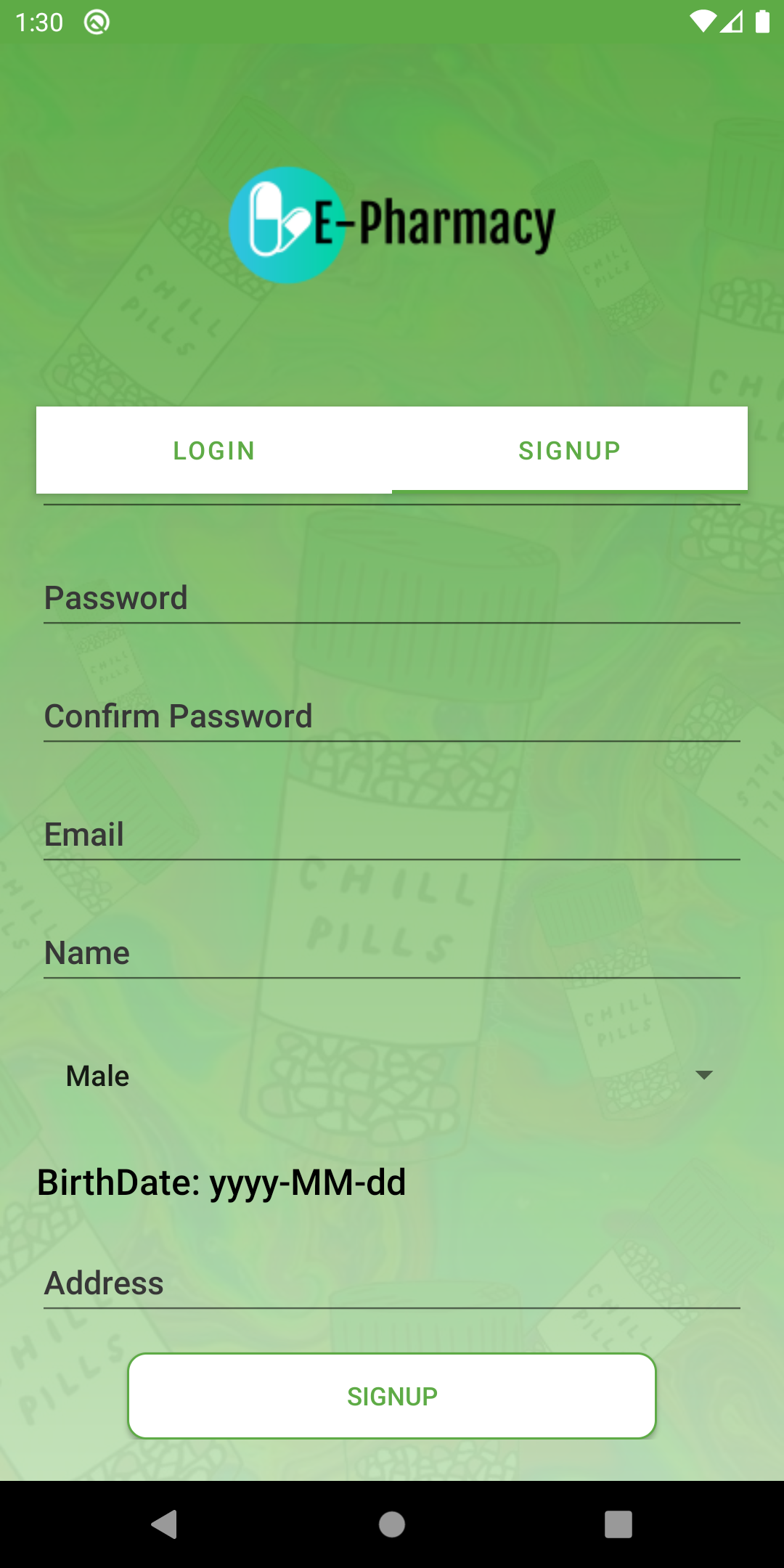
***Fields***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |

***Buttons/Hyperlinks***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 1 | Choose Google Account | User choose an account to login or register | N/A | User will be navigated to Home Screen |

### 5.3. Sign Up Screen



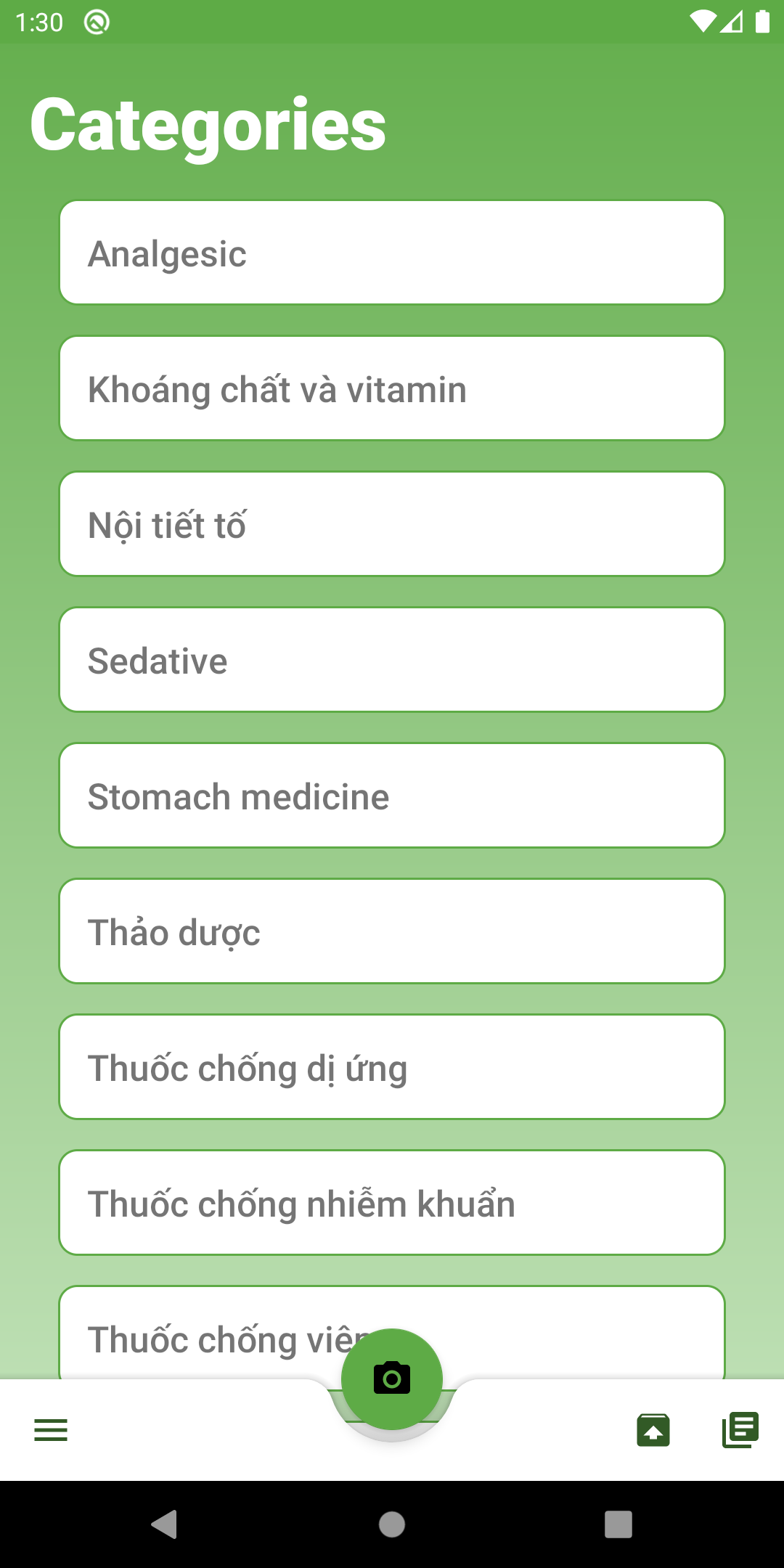
***Fields***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |
| 1 | Username | Account’s name | No | Yes | EditText | String | 32 |
| 2 | Password | Account’s password | No | Yes | EditText | String | 60 |
| 3 | Confirm Password | Confirm Password | No | Yes | EditText | String | 60 |
| 4 | Email | User’s email | No | Yes | EditText | String | 320 |
| 5 | Name | User’s name | No | Yes | EditText | String | 45 |
| 6 | Gender | User’s gender | No | Yes | Spinner | boolean | 1 |
| 7 | Birthdate | User’s birthdate | No | Yes | DatePicker | String | 10 |
| 8 | Address | User’s address | No | Yes | EditText | String | 50 |

***Buttons/Hyperlinks***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 9 | Sign Up | Sign up new account | All fields must be input | User will be navigated to home screen |

### 5.4. Home Screen



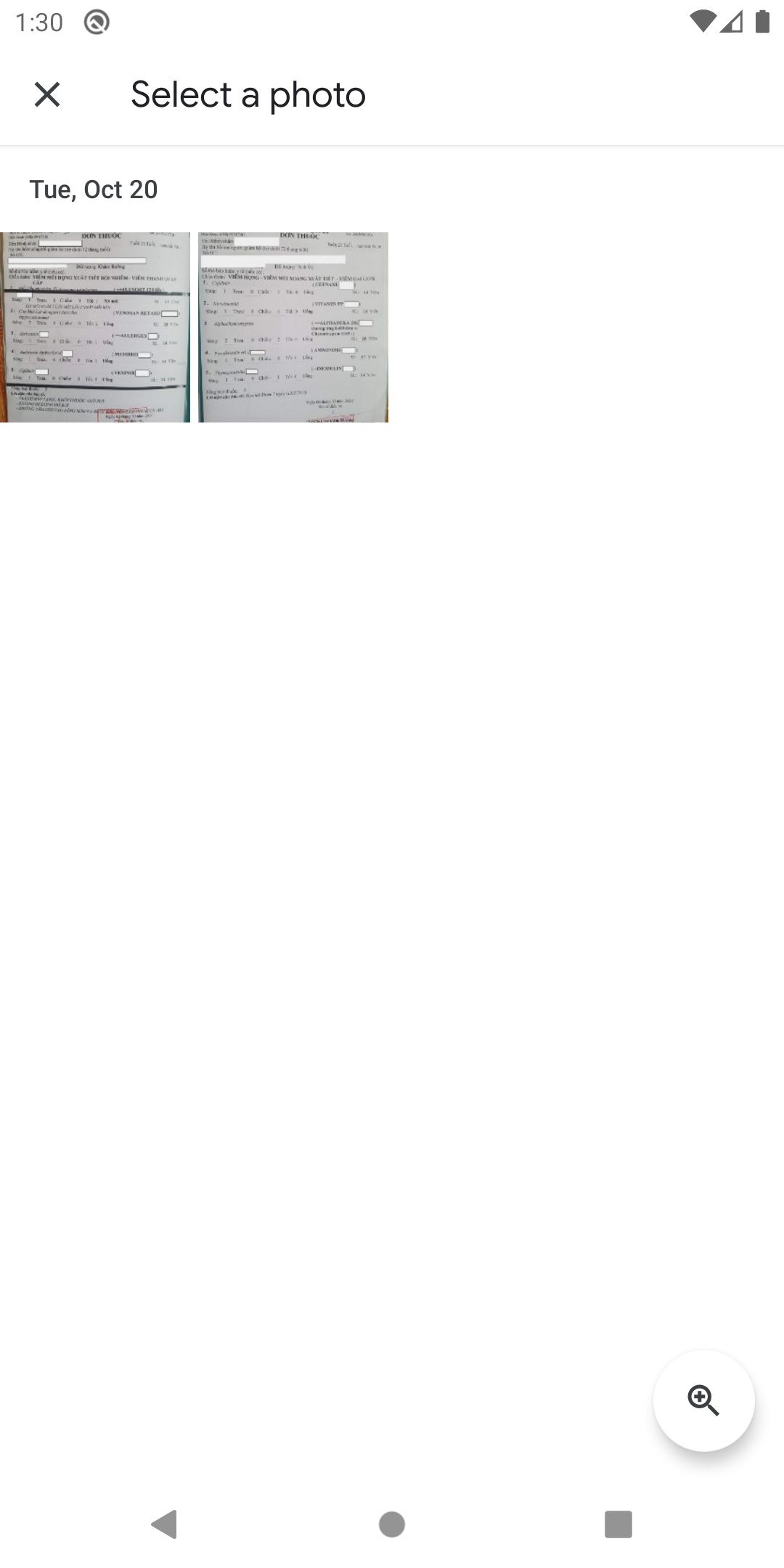
***Fields***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |

***Buttons/Hyperlinks***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 1 | Main Scan Button | Open Camera to Scan Prescription | N/A | User will be navigate to camera screen |
| 2 | Account Hamburger Button | Open Account option for user | N/A | Account bottom sheet will pop up |
| 3 | Upload Image | User can choose an image to be scan instead of taking a picture | N/A | User will be navigate to gallery picking screen |
| 4 | List Prescription | Navigate to Prescription List Screen | N/A | User will be navigate to Prescription List Screen |

### 5.5. Gallery Picking Screen



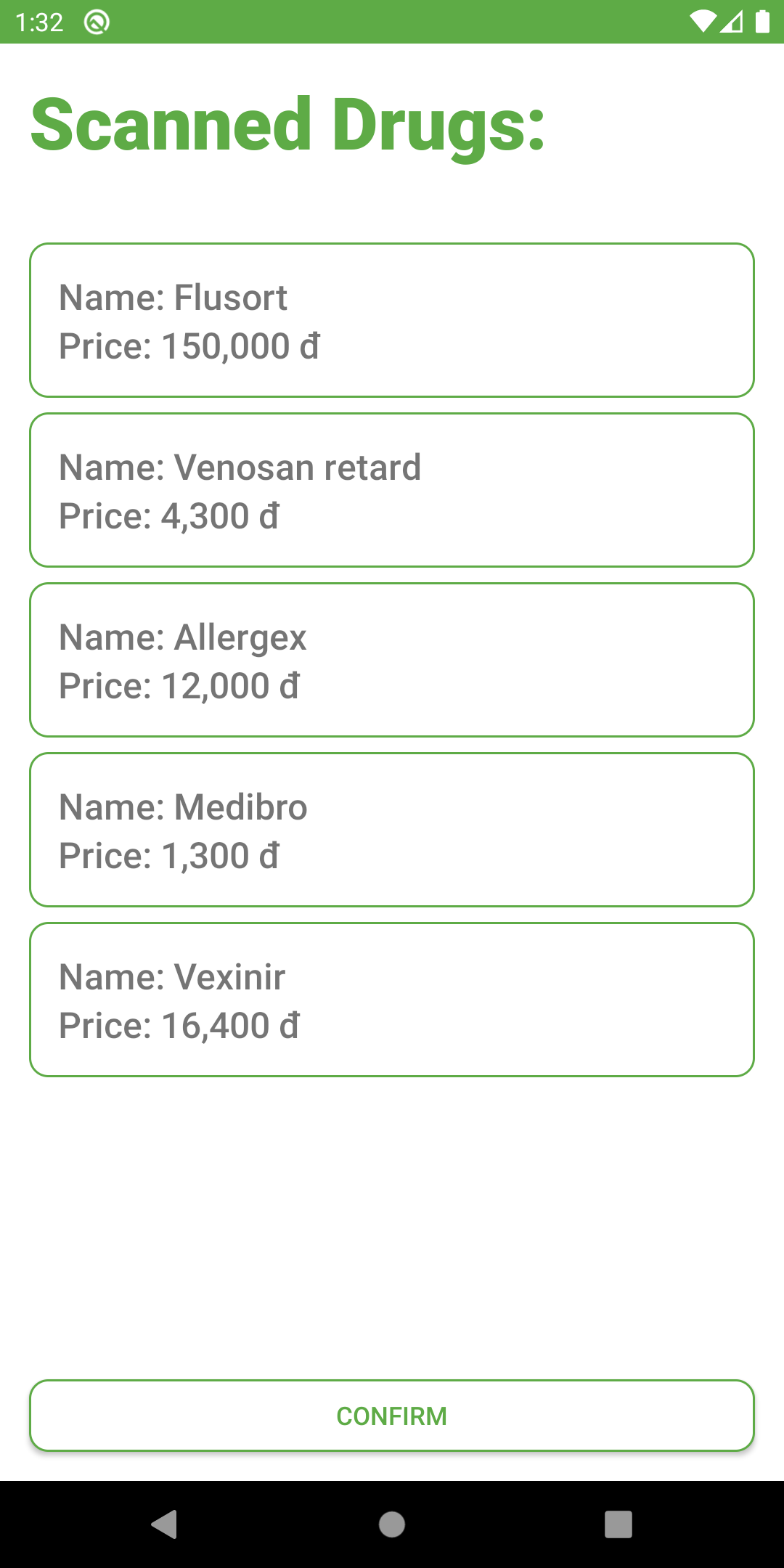
***Fields***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |

***Buttons/Hyperlinks***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 1 | Choose a image | User choose an image to be scan | N/A | User will be navigate back to home screen and image will be process |

### 5.6. Scan Result Screen



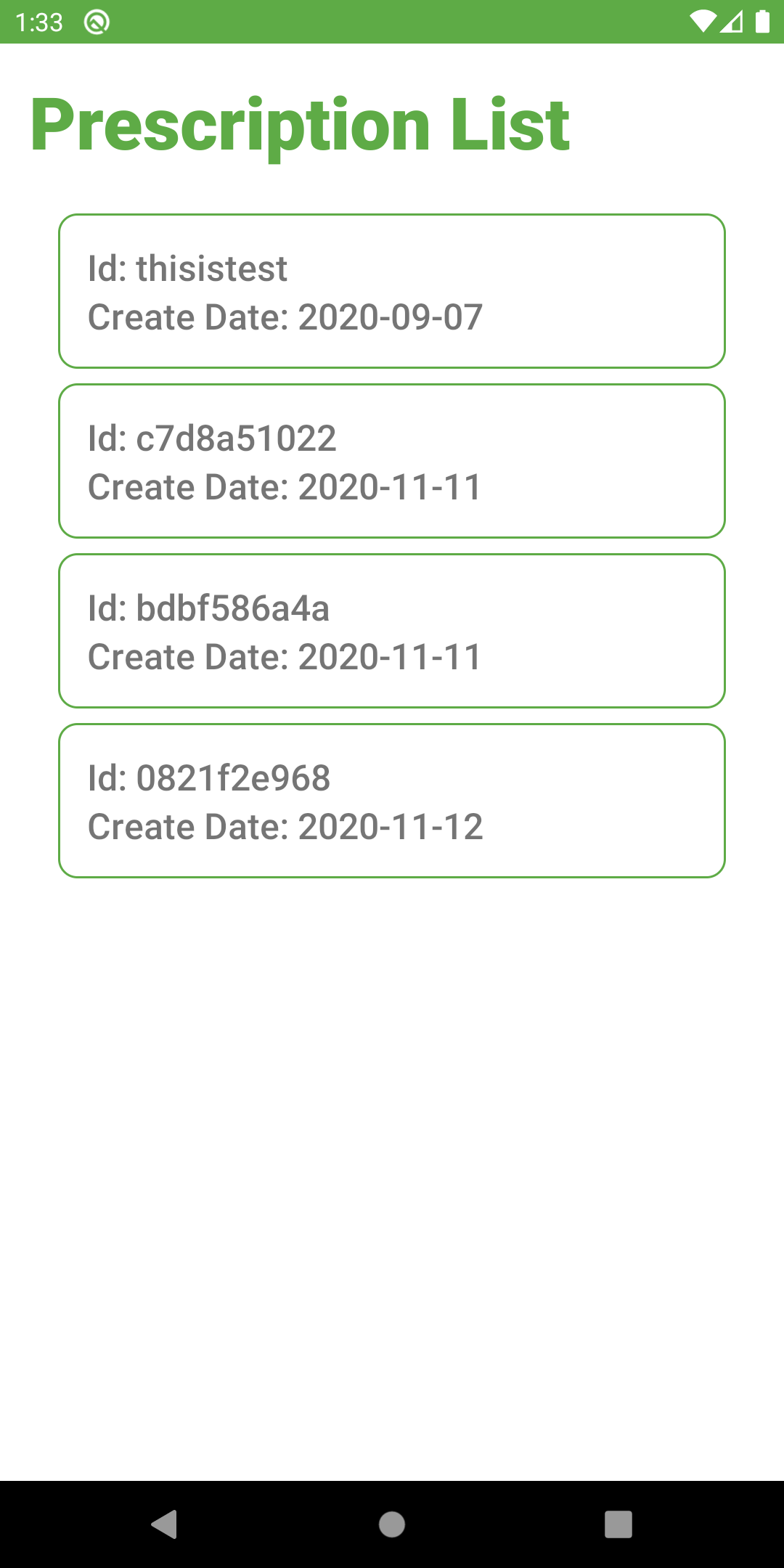
***Fields***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |

***Buttons/Hyperlinks***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 1 | Confirm | User confirm that they want to create prescription | N/A | Prescription will be created and user will be navigate back to home screen |

### 5.7. Prescription List Screen



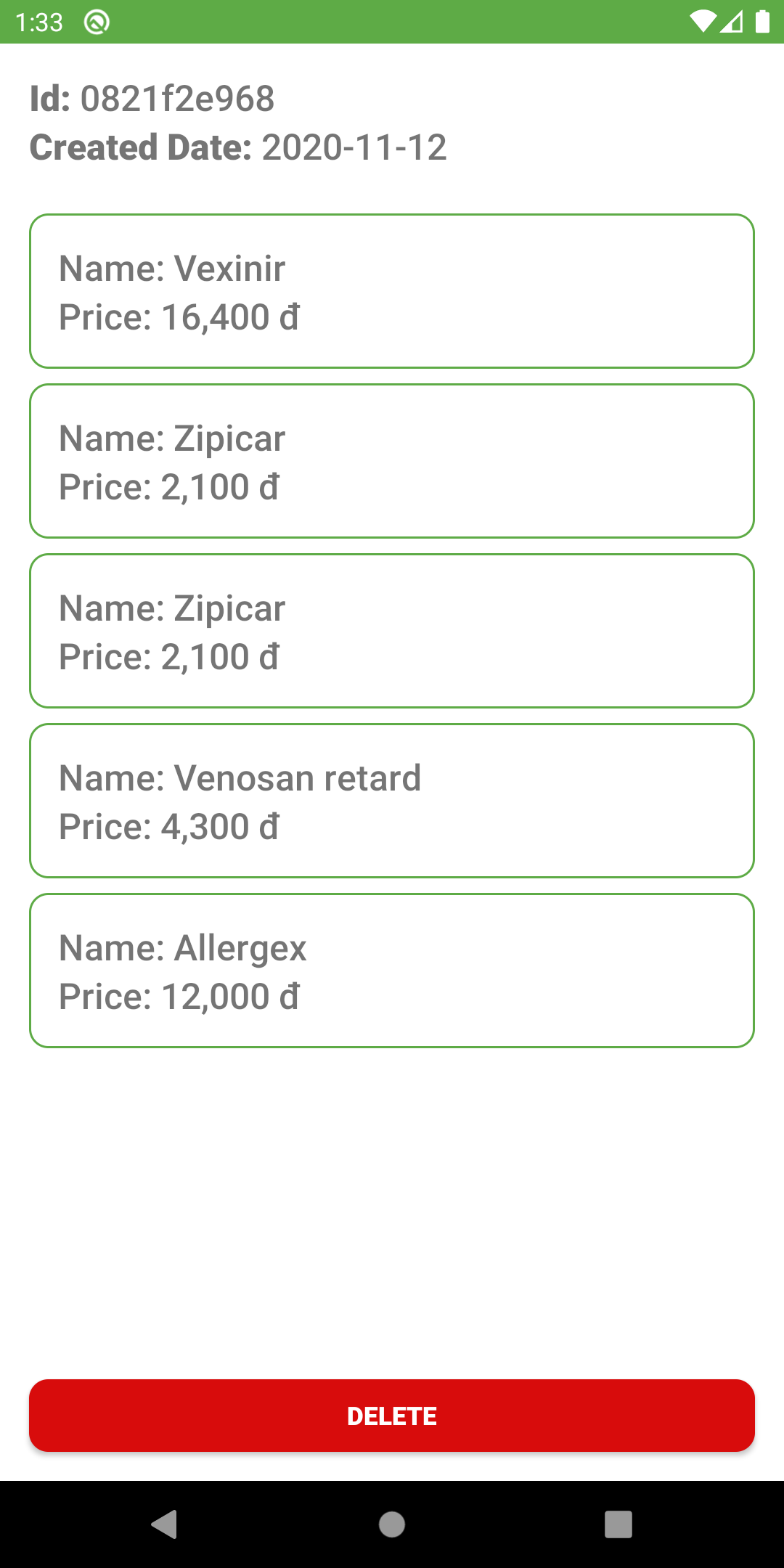
***Fields***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |

***Buttons/Hyperlinks***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 1 | Prescription Item | User can see prescription detail | N/A | User will be navigate to prescription detail screen |

### 5.8. Prescription Detail Screen



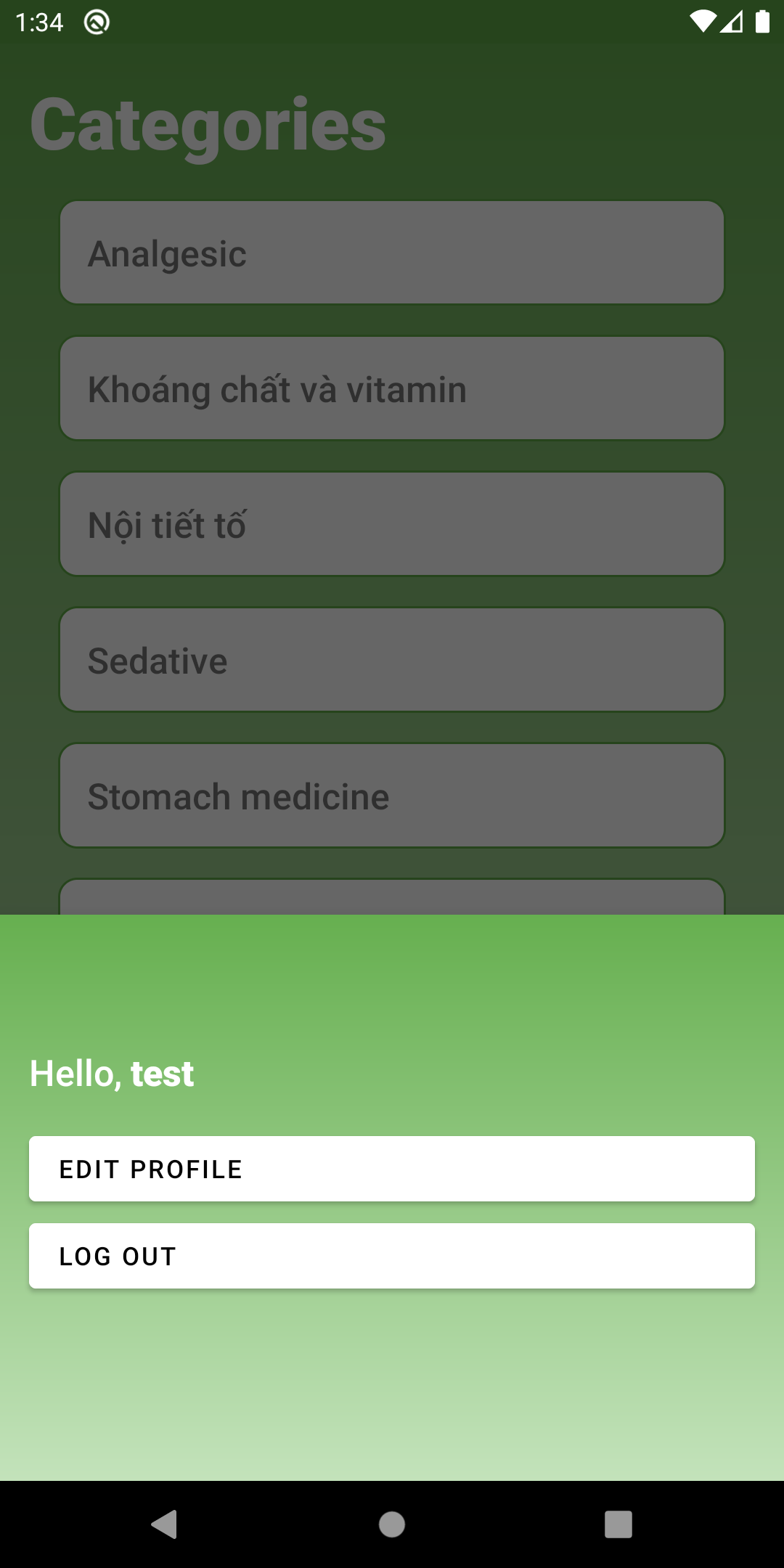
***Fields***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |
| 11 | Id | Show Prescription Id | Yes | Yes | TextView | String | 10 |
| 22 | Created Date | Show Prescription Created Date | Yes | Yes | TextView | String | 10 |

***Buttons/Hyperlinks***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 1 | Delete | User want to delete prescription | N/A | Prescription will be deleted, and user will be navigated back to prescription list screen |

### 5.9. Account Option Bottom Sheet Dialog



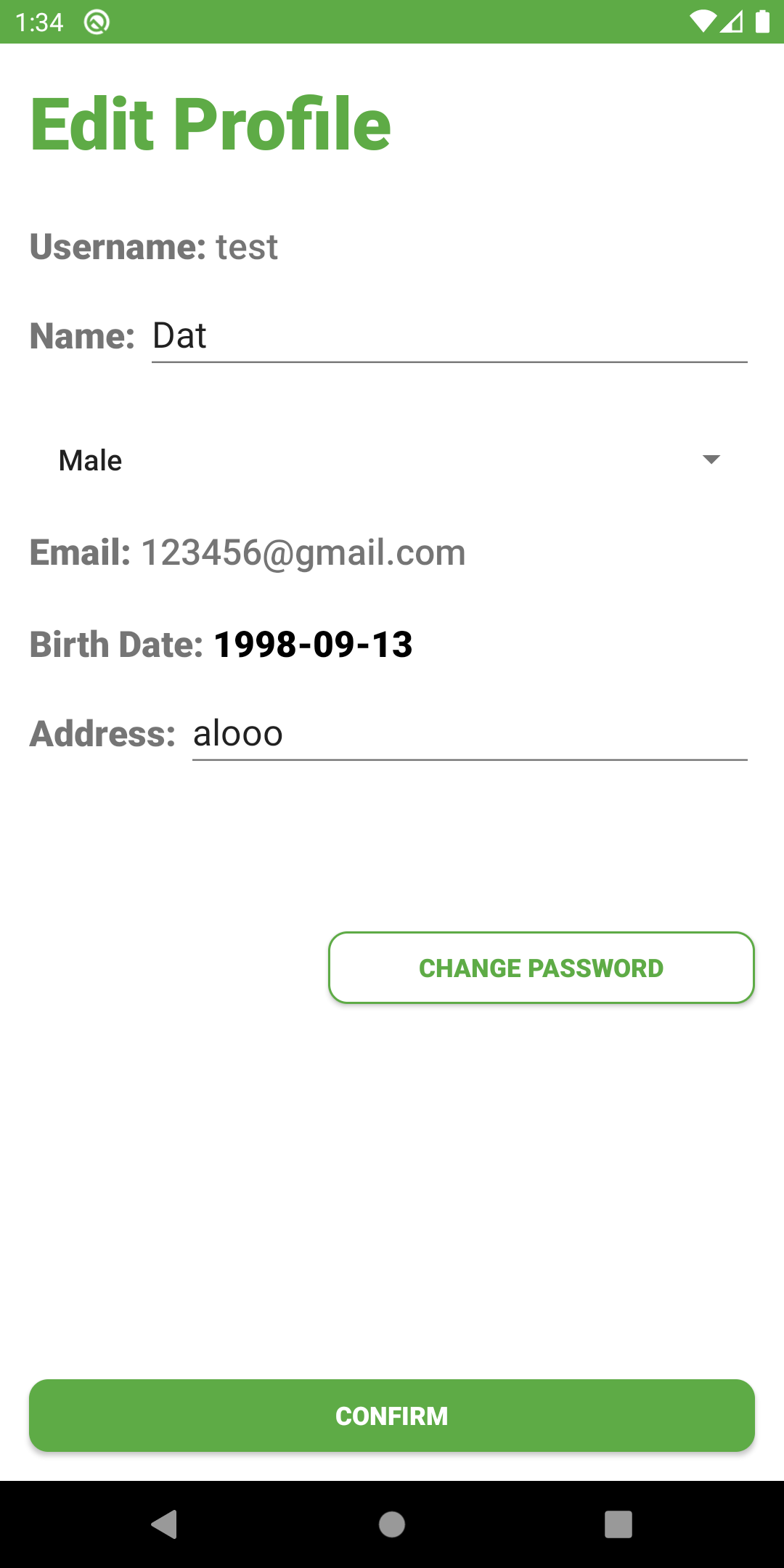
***Fields***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |
| 11 | Username | Username will be display | Yes | Yes | TextView | String | 50 |

***Buttons/Hyperlinks***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 2 | Edit profile | User choose to navigate to edit profile screen | N/A | User will be navigated to edit profile screen |
| 33 | Logout | Logout of account | N/A | Account will be log out, user will be navigated to login screen |

### 5.10. Edit Profile Screen



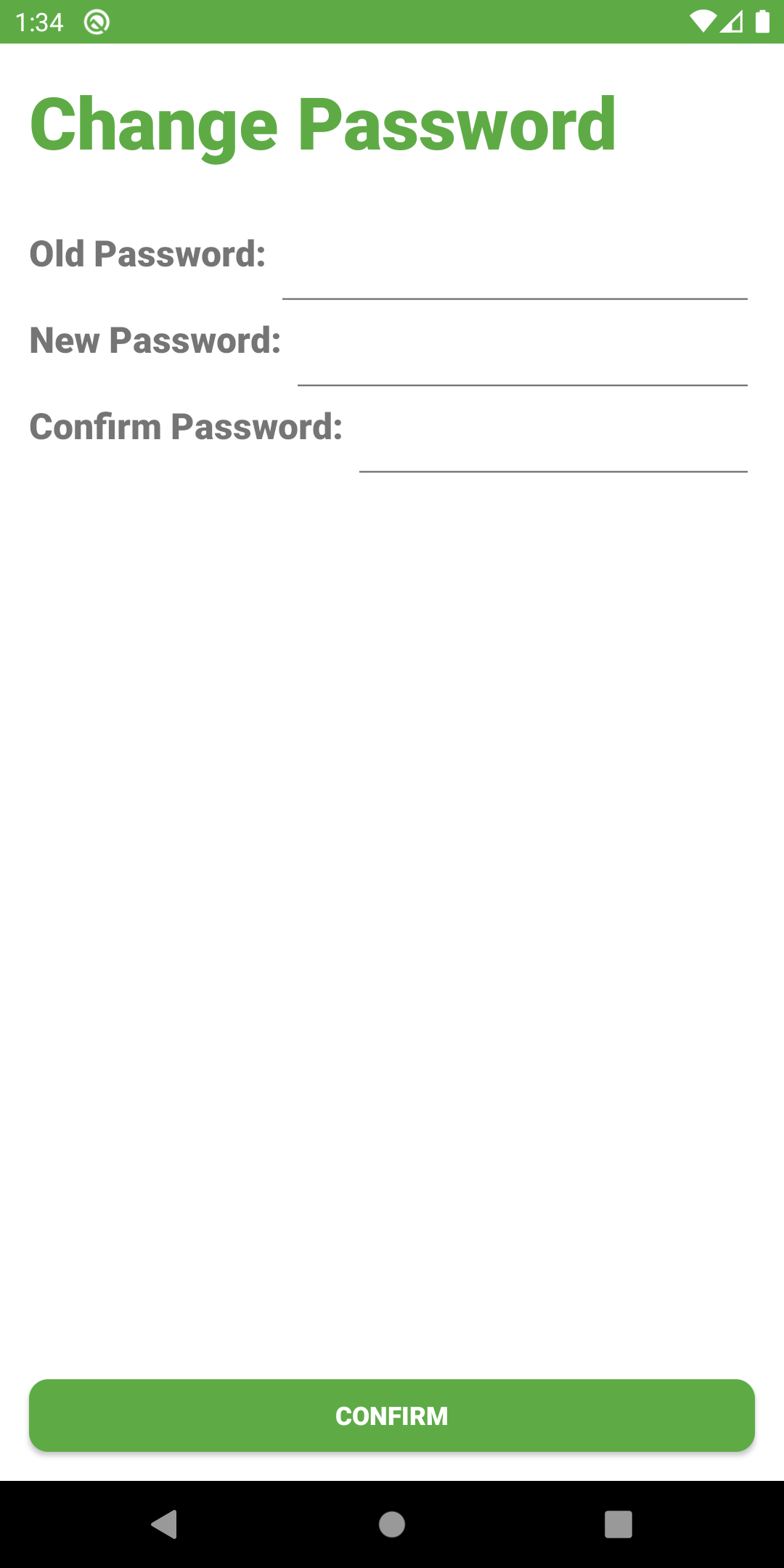
***Fields***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |
| 11 | Username | Account’s username | Yes | Yes | TextView | String | 32 |
| 2 | Name | Account’s name | No | Yes | EditText | String | 45 |
| 3 | Gender | Account’s gender | No | Yes | Spinner | boolean | 1 |
| 4 | Email | Account’s email | Yes | Yes | TextView | String | 320 |
| 5 | Birthdate | Account’s birthdate | No | Yes | DatePicker | String | 10 |
| 6 | Address | Account’s address | No | Yes | EditText | String | 50 |

***Buttons/Hyperlinks***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 7 | Change Password | Go to change password screen | N/A | User will be navigated to change password screen |
| 8 | Confirm | Confirm edited profile | All fields must be inputted | Profile will be updated with inputted fields, user will be navigate back to home screen |

### 5.11. Change Password Screen



***Fields***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Description** | **Read only** | **Mandatory** | **Control Type** | **Data Type** | **Length** |
| 11 | Old Password | Account’s old password | No | Yes | EditText | String | 60 |
| 2 | New Password | New password | No | Yes | EditText | String | 60 |
| 3 | Confirm Password | Confirm new password | No | Yes | EditText | String | 60 |

***Buttons/Hyperlinks***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Function** | **Description** | **Validation** | **Outcome** |
| 4 | Confirm | Confirm changes | All fields must be inputted and validated | Account’s password will be change, user will be navigated back to edit profile screen |

## 6. Table Relationship

