

Yarmouk University

Faculty of Information Technology and Computer Sciences

Department of Computer Sciences

Documentation for

Smart Dr. Diagnosis Application (Smart Dr.D)

Supervisor:

Dr. Ra'ed M Al-khatib

Project Team:

|  |  |  |
| --- | --- | --- |
| **No.** | **Student Name** | **University No.** |
|  | Ammar Mahafza | **2015901133** |
|  | Dania alkhateeb | **2015901198** |
|  | Lama Abu Sohyuon | **2015901123** |

Contents

[**List of Figures** iii](#_Toc10983449)

[**List of tables** v](#_Toc10983450)

[Abstract vi](#_Toc10983451)

[1. Introduction 1](#_Toc10983452)

[2. Background and Literature review 2](#_Toc10983453)

[3. Contribution and Objectives 6](#_Toc10983454)

[4. System analysis 6](#_Toc10983455)

[4.1. Requirements and specifications 6](#_Toc10983460)

[1. Functional requirements 6](#_Toc10983461)

[2. Non-Functional requirements 7](#_Toc10983462)

[3. Software and Hardware Requirement 8](#_Toc10983463)

[4.2. Use case Diagrams 9](#_Toc10983464)

[4.2.1. Use case Diagram 9](#_Toc10983465)

[4.2.2. Use case scenario 11](#_Toc10983466)

[4.3. Data flow Diagram 18](#_Toc10983472)

[4.3.1. Context diagram 18](#_Toc10983478)

[4.3.2. Data flow diagram L1 19](#_Toc10983479)

[5. Algorithmic Design 20](#_Toc10983480)

[5.1. Pseudo code. 20](#_Toc10983486)

[6. Flow Chart 23](#_Toc10983487)

[6.1. Flow Chart for patient 23](#_Toc10983491)

[6.2. Flow Chart for doctor 24](#_Toc10983492)

[7. Screen shots for project 25](#_Toc10983493)

[8. References 42](#_Toc10983494)

**List of Figures**

[Figure 1 general screen for WebMD Symptom Checker Application 2](#_Toc10983146)

[Figure 2 four levels for Gout Central empowers app 3](#_Toc10983147)

[Figure 3 Advances in Chronic Kidney Disease app 4](#_Toc10983148)

[Figure 4 Use case 9](#_Toc10983149)

[Figure 5 Context Diagram 18](#_Toc10983150)

[Figure 6 Data flow level 1 19](#_Toc10983151)

[Figure 7 Flow chart – patient 23](#_Toc10983152)

[Figure 8 Flow chat - doctor 24](#_Toc10983153)

[Figure 9 Splash Screen Smart DrD 25](#_Toc10983154)

[Figure 10 The main Body 26](#_Toc10983155)

[Figure 11 Register for Doctor 27](#_Toc10983156)

[Figure 12 Users Login 28](#_Toc10983157)

[Figure 13 Add Account for login 29](#_Toc10983158)

[Figure 14 main Menu for Doctor 30](#_Toc10983159)

[Figure 15 Update Account Information 31](#_Toc10983160)

[Figure 16 Add question and Reults 32](#_Toc10983161)

[Figure 17 Select Disease by Patient 33](#_Toc10983162)

[Figure 18 Show question by disease Selection 34](#_Toc10983163)

[Figure 19 Show question by answer 35](#_Toc10983164)

[Figure 20 show next question by previous answer 36](#_Toc10983165)

[Figure 21 show last question by the tree of answers 37](#_Toc10983166)

[Figure 22 show symptoms diagnosis 38](#_Toc10983167)

[Figure 23 add data to doctor register 40](#_Toc10983168)

[Figure 24 Show message when register successfully 41](#_Toc10983169)

**List of tables**

[Table 1 login use case 11](#_Toc10983170)

[Table 2 register use case 12](#_Toc10983171)

[Table 3 Manage account use case 13](#_Toc10983172)

[Table 4 Add symptom use case 14](#_Toc10983173)

[Table 5 Add medical diagnosis use case 15](#_Toc10983174)

[Table 6 Select body part use case 16](#_Toc10983175)

[Table 7 Select symptom use case 17](#_Toc10983176)

# Abstract

The Smart Dr. Diagnosis (Smart Dr.D) application will be developed application on smartphone to uncovered patients diagnose their sickness through answer some suggested questions. There is more than one user in Smart Dr. Diagnosis (Smart Dr.D) application while categorized into two types (doctor, patient). The patient can get information about doctors and clinics. There is a Google maps helps patient to get address for clinics by show direction.

تطبيق الطبيب الذكي سيتم تطويره على الهواتف الذكية والذي يساعد المرضى بتشخيص أمراضهم من خلال طرح مجموعة من الأسئلة المقترحة من النظام حسب الحالة التي التي يتم اختيارها, ويتوقع من النظام بعد ذلك عمل التشخيص المناسب بناءُ على الاجابات من المريض. وسيعمل التطبيق من خلال المهام المتعلقة بالطبيب والمريض حيث يتم الاختيار بينهما للتعامل مع الميزات التي يطرحها التطبيق لاحقا. في حال تم استخدام التطبيق من خلال الطبيب يتم تزويد التطبيق بالأعراض الخاصة بالأمراض, وعند استخدامه من خلال المريض يتم تشخيص المرض وعند الأنتهاء من التشخيص يتم استعراض عناويين العيادات الخاصة بنتائج التشخيص من خلال استخدام خرائط جوجل للوصول إلى المواقع وأتجاهاتها.

# 

# Introduction

People need to go to doctor, but they have not enough courage to treatment, in addition to a shyness of answer for all questions.

In this system we suggest an application called Smart Dr. Diagnosis application (Smart Dr.D), Dr. Diagnosis is used to help all age groups and both gender to know their problem or illness by answering some of questions about their symptoms which the application can provides.

In addition, the smart Dr.D application provides menus for doctors to manage the application symptoms of disease that can affect patients and the indication of the disease potential, and then show clinics address with the google maps to determine trends towards. Phones and tablet using android operating system is widely and growth quickly; therefore the access to the application has become easily so it can cover a wide range of users.

# Background and Literature review

In this section, a detail for some previous research will be explained.   
In [1] aWebMD Symptom Checker, identifying the illness and possible conditions and treatment related to your symptoms. This tool does not provide medical advice it is intended for informational purposes only. It is not a substitute for professional medical advice, diagnosis or treatment.

In[2] the researcher introduced aCare after Kidney Transplant App**,** this App for patients is a convenient way to learn how to stay healthy with a kidney transplant, as seen in figure 1. It provides the answers to frequently asked questions about how and why it is important to follow all of the instructions and recommendations from your transplant team.

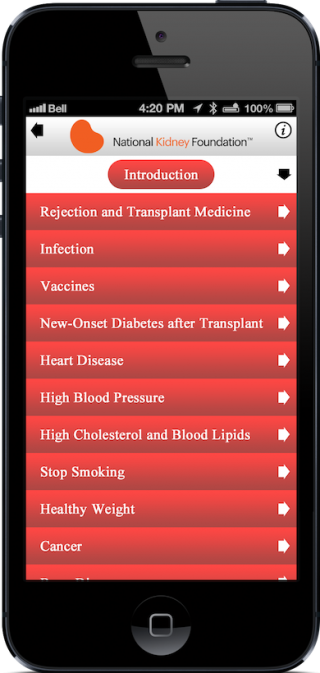


Figure 1 general screen for WebMD Symptom Checker Application

In the app Gout Central [3] empowers patients with the most essential tools and information for controlling gout and protecting their kidneys, as seen in [figure 2]. Guidance is provided on the optimal use of nutrition, lifestyle, and medication for the prevention and treatment of gout flares.

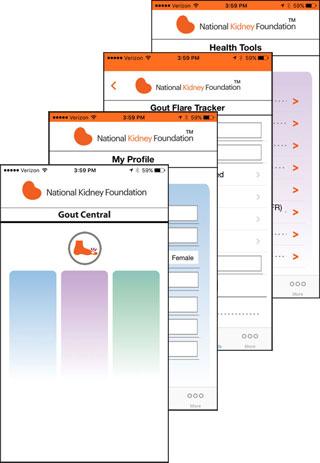


Figure 2 four levels for Gout Central empowers app

In [4] the appAdvances in Chronic Kidney Disease (ACKD). Keep up with the most important advances in the science and practice of nephrology. The Advances in Chronic Kidney Disease app is available for iOS and Android, bringing you the convenience of reading your favorite articles online or offline, as seen in figure 3.

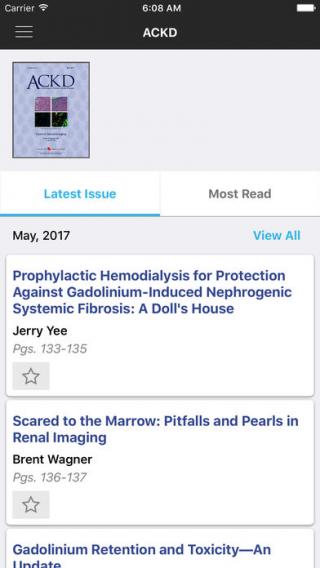


Figure 3 Advances in Chronic Kidney Disease app

In Parent friendly app [5], Milestone Tracker. To track your child’s developmental milestones from age 2 months to 5 years!, Photos and videos in this app illustrate each milestone and make tracking them for your child easy and fun.

Get a summary of your child’s milestones; view tips and activities to support your child’s development; and find out what to do if you ever have a developmental concern.

The proposed Smart Dr.D system to help users to take advice about disease by symptoms that user himself feeling.

In our proposed project we will try to discover the disease through accurate questions for the patient and then suggest the appropriate doctor to treat this disease.

# Contribution and Objectives

The objective of the Smart Dr. Diagnosis (Smart Dr.D) application is to help patients diagnose their sickness in easy way get from potential symptom and suggest the nearest clinics to diagnosis sick by determine map from current location to target selection, additionally description about doctor and clinics. The doctor types must have his/ her register on it, after that the doctor can use the services of Smart Dr. Diagnosis (Smart Dr.D) through easy steps of questions to describe the sickness case. The user types no need to register on it, when user start the application, can use the services of Smart Dr. diagnosis (Smart Dr.D) through easy steps of questions to determine sickness case, then describe this case and suggest doctor appropriate with information.

# System analysis



# Requirements and specifications

# Functional requirements

Functional Requirements for **doctor**

* The system should allow doctor to sign up or register.
* The system should allow doctor to select human body parts.
* The system should allow doctor to add the symptoms question.
* The system should allow doctors to add result diagnostic symptoms.

Functional Requirements for **patient**

* The system should allow patient to start using features directly.
* The system should allow patient to select human body parts.
* The system should allow patient to answer the diagnostic symptoms question.
* The system show patient appropriate symptoms result.
* The system should allow patient to get description about the suitable medical care center.
* The system should allow patient to get map direction about clinics address.

# Non-Functional requirements

We describe the application works.

1. Easy to use.  
   Using GUI with functionality the same of popular applications.
2. Flexibility.  
   The doctor can add many symptoms and questions.
3. According to the system requirement.

The application uses the minimum resources from device.

1. Easy to debug and maintain.  
   The application can upgrade monthly.

# Software and Hardware Requirement

1. Platform Smartphone using Android OS
2. API version 2.3 or latest for (Google API maps).
3. Internet connection for GPS maps direction.

# Use case Diagrams

# Use case Diagram



Figure 4 Use case

This UML use case diagram example shows actor and use cases for a Smart Dr.D. **Smart Dr.D** system or module supports some of the many job duties of an administrator. Administrator adds the human body parts images, collects information from the doctor by emails registered.

For the patient that will stay in the application she or he should answer questions. The system might also receive feedback, record them in a database.

# Use case scenario

Table 1 login use case

|  |  |
| --- | --- |
| Use case ID | Dr.D 01 |
| Use case name | Login. |
| Actor | Administrator, Doctor |
| Pre-condition | There are no preconditions associated with this use case. |
| Post-condition | There is no post conditions associated with this use case. |
| Normal flow | 1. The system validates the actor’s password and logs him/her into the system. 2. The system displays the Main Form and the use case ends. |
| Alternative flow | 1. Invalid Name / Password   If in the basic flow the system cannot find the name or the password is invalid, an error message is displayed. The actor can type in a new name or password or choose to cancel the operation, at which point the use case ends. |



Table 2 register use case

|  |  |
| --- | --- |
| Use case ID | Dr.D 02 |
| Use case name | Register. |
| Actor | Doctor |
| Pre-condition | Before this use case begins the user has no account onto the system. |
| Post-condition | The account has created successfully. |
| Normal flow | 1. The doctor selects "Sign Up." 2. The system displays a blank register form. 3. The doctor adds information account is performed at this step for each selected offering. 4. The system saves the information. 5. The system prompts for account created successfully. |
| Alternative flow | 1. Invalid account created.   If in the basic flow the system cannot create account or the some information is invalid, an error message is displayed. The actor can type in a new information or choose to cancel the operation. |

Table 3 Manage account use case

|  |  |
| --- | --- |
| Use case ID | Dr.D 03 |
| Use case name | Manage account. |
| Actor | Doctor, administrator |
| Pre-condition | Before this use case begins the user has logged onto the system. |
| Post-condition | Updated user information successfully. |
| Normal flow | 1. The user selects "my profile." 2. The system displays the user form. 3. The user enters or updates information. 4. The system validates the data to insure the proper data format. 5. If the data is valid the system updates new information. |
| Alternative flow | 1. Invalid data account.   If in the basic flow the system cannot update account or the some information is invalid, an error message is displayed. The actor can type in a new information or choose to cancel the operation. |

Table 4 Add symptom use case

|  |  |
| --- | --- |
| Use case ID | Dr.D 04 |
| Use case name | Add symptom. |
| Actor | Doctor |
| Pre-condition | Before this use case begins the doctor has logged onto the system. |
| Post-condition | Add symptoms information successfully. |
| Normal flow | 1. The doctor selects "add a symptom." 2. The system displays a blank symptom form. 3. The doctor enters the following information for the symptom: name, description, status, and type. 4. The system validates the data to insure the proper data format and searches for an existing symptom with the specified name. If the data is valid the system creates a new symptom and assigns a unique system-generated id number. This number is displayed, so it can be used for subsequent uses of the system. 5. Steps 2-4 are repeated for each symptom added to the system. When the doctor is finished adding symptoms to the system the use case ends. |
| Alternative flow | 1. Invalid data symptom.   Modify and Delete a symptom. |

Table 5 Add medical diagnosis use case

|  |  |
| --- | --- |
| Use case ID | Dr.D 05 |
| Use case name | Add medical diagnosis. |
| Actor | Doctor |
| Pre-condition | Before this use case begins the doctor has logged onto the system. |
| Post-condition | Add medical diagnosis information successfully. |
| Normal flow | 1. The doctor selects "Add medical diagnosis." 2. The system displays a symptom form. 3. Steps 2 are repeated for each symptom added to the system. 4. The doctor enters the following information for the medical diagnosis: name, description. 5. The system validates the data to insure the proper data format and searches for an existing medical diagnosis with the specified name. If the data is valid the system creates a new medical diagnosis and assigns a unique system-generated id number. This number is displayed, so it can be used for subsequent uses of the system. |
| Alternative flow | 1. Invalid data account.   Cancel operation for a medical diagnosis. |

Table 6 Select body part use case

|  |  |
| --- | --- |
| Use case ID | Dr.D 06 |
| Use case name | Select human body part. |
| Actor | Patient |
| Pre-condition | None. |
| Post-condition | View details for sub body part. |
| Normal flow | 1. The system retrieves and displays the image of body. 2. The patient selects the image offerings that he/she wishes to ask for the upcoming symptoms. 3. The system verifies that the selected offerings do not with body shape. |
| Alternative flow | 1. Invalid select body image.   No operation executes and the system prompt error message. |

Table 7 Select symptom use case

|  |  |
| --- | --- |
| Use case ID | Dr.D 07 |
| Use case name | Select symptom. |
| Actor | Patient |
| Pre-condition | None. |
| Post-condition | View details for symptom. |
| Normal flow | 1. The system retrieves and displays the image of body. 2. The patient selects the image offerings that he/she wishes to ask for the upcoming symptoms. 3. The patient answers the question offerings that he/she upcoming by symptoms. 4. Steps 3 are repeated for each question symptom added to the system. 5. When the patient is finished answering questions the use case ends |
| Alternative flow | If in the basic flow the system cannot find any question from the previous answer for the patient, a message is displayed. Once the patient acknowledges the message the use case terminated. |

# Data flow Diagram



# Context diagram



Figure 5 Context Diagram

Context Diagrams communicate the interactions and flow of data between processes. This diagram is called data flow diagram level 0; it shows the general interaction between the patient, doctor, administrator, map and Smart Dr.D system application in an abstract view, which contains the main functions of this application which are communicated. All of those main operations and other operations will be detailed more, next in [Figure 6].

# Data flow diagram L1



Figure 6 Data flow level 1

# Algorithmic Design



# Pseudo code.

Pseudo code for patient:

Inputs:

A special bipartite tree graph which has set of vertices d1,…., dm as diagnoses and s1,… sn as symptoms and each pair (di, sj) has two edges representing P(Sj|di) and P(di|Sj)

*A presenting body set, PresentingBody.*

*A presenting sub body set, PresentingSubBody.*

*A presenting doctor set, PresentingDoctor.*

Outputs:

*Set of Diagnosis, Diagnosis Included.*

*A presenting sub body set, PresentingSubBody.*

*A presenting doctor set, PresentingDoctor.*

*A presenting doctor set, PresentingElicited;*

*Threshold for DiagenesesToBeElicited .*

Begin:

*symptomsFound = null.*

*Select PresentingSubBody*

*For each S in PresentingSubBody*

*Add s to Symptoms ;*

*Add s to SymptomsFound ;*

*For each di to where P(di|s) >= Di*

*Add di to DiagenesesToBeElicited*

*End for*

*While (DiagenesesToBeElicited <> null AND PresentingElicited <> null*

*Ask patient S*

*doctorInfo()*

*End while*

*symptomsFound = D*

*exit*

*end for*

*doctorInfo(){*

*For each s in DoctorsymptomsFound*

*}*

Pseudo code for doctor:

Inputs:

*A presenting sub body set, PresentingSubBody.*

*A presenting doctor set, PresentingDoctor.*

*Threshold for symptoms S.*

*Threshold for disease D.*

Outputs:

*Set of Diagnosis, Diagnosis Included.*

*A presenting sub body set, PresentingSubBody.*

*A presenting doctor set, PresentingDoctor.*

*Threshold for symptoms S.*

*Threshold for disease D.*

Begin:

*D = null.*

*S = null.*

*Select PresentingBody*

*Select PresentingSubBody*

*For each S in PresentingSubBody*

*Add question for S*

*Next()*

*Add Counter*

*If counter equal 4 then*

*Disease()*

*end if*

*end for*

*next(){*

*for each D in PresentingSubBody*

*symptomsFound=S*

*end for*

*}*

*Disease(){*

*for each D in PresentingSubBody*

*DiseaseFound=D*

*end for*

*}*

# Flow Chart



# Flow Chart for patient



Figure 7 Flow chart – patient

This is a patient flowchart example that shows how process cases shall be processed. This flowchart uses decision shapes intensively in representing alternative flows.

# Flow Chart for doctor



Figure 8 Flow chat - doctor

This is a doctor flowchart example that shows how process cases shall be processed. This flowchart uses decision shapes intensively in representing alternative flows.

# Screen shots for project



Figure 9 Splash Screen Smart DrD

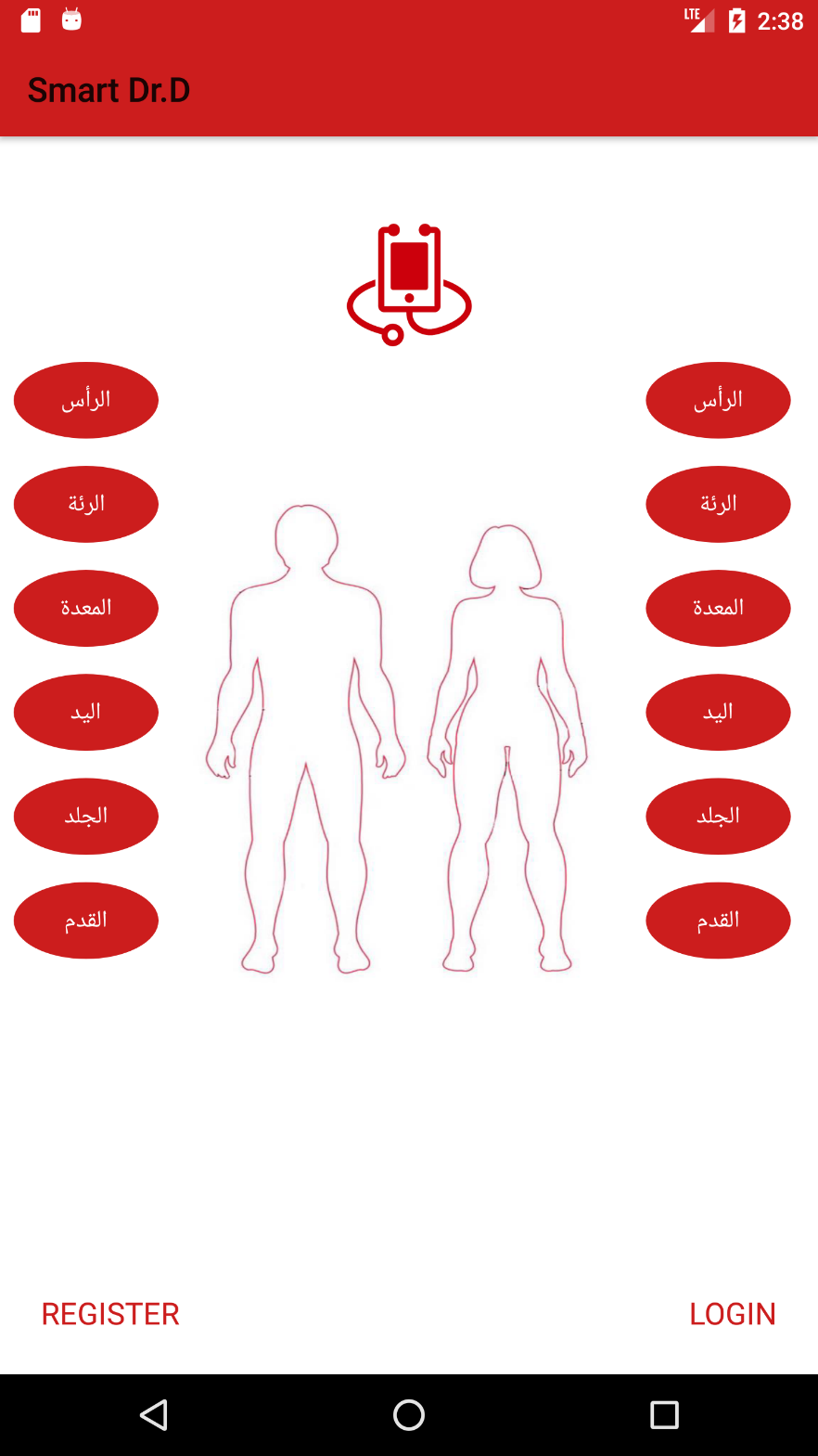


Figure 10 The main Body

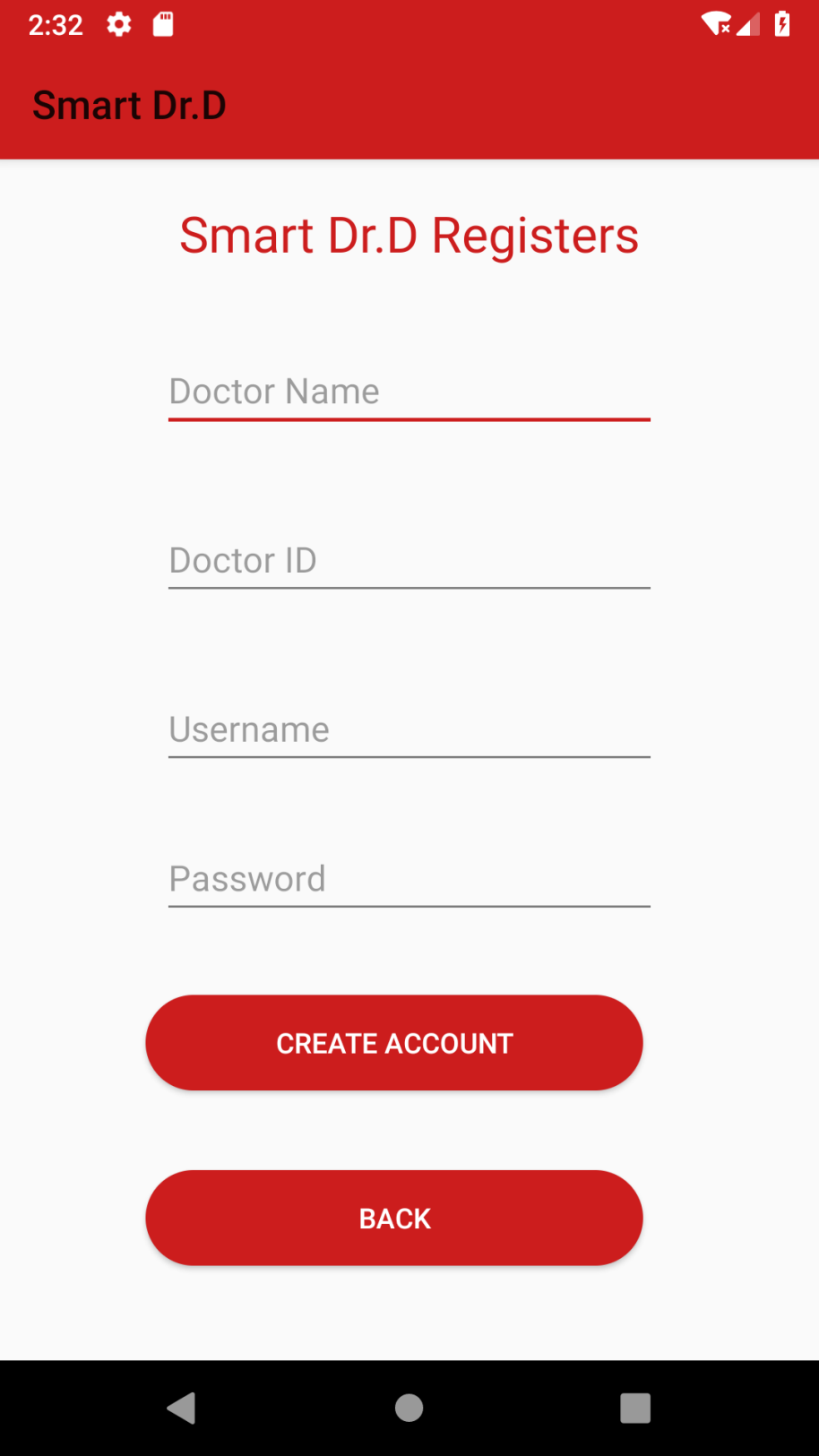


Figure 11 Register for Doctor

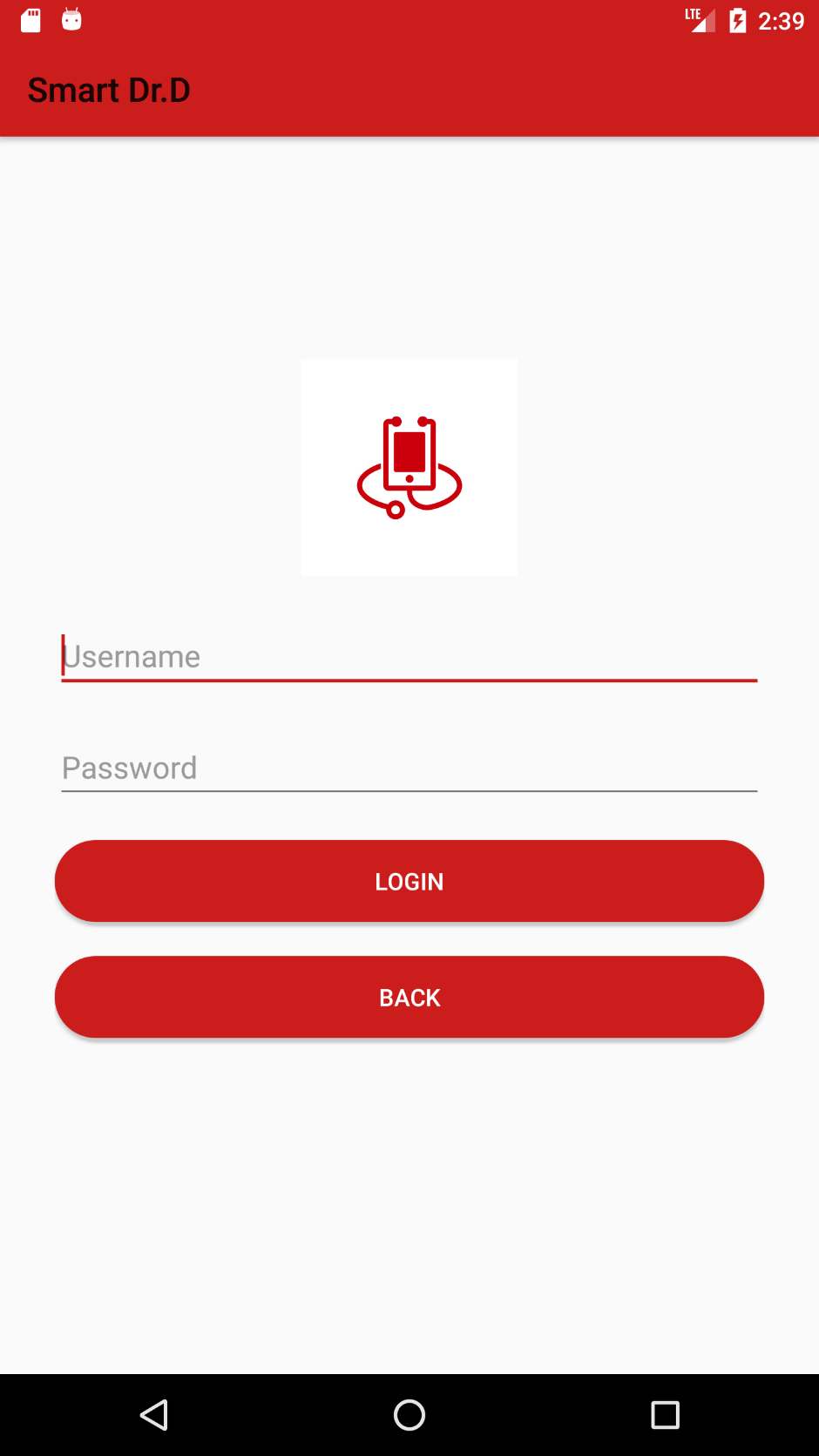


Figure 12 Users Login

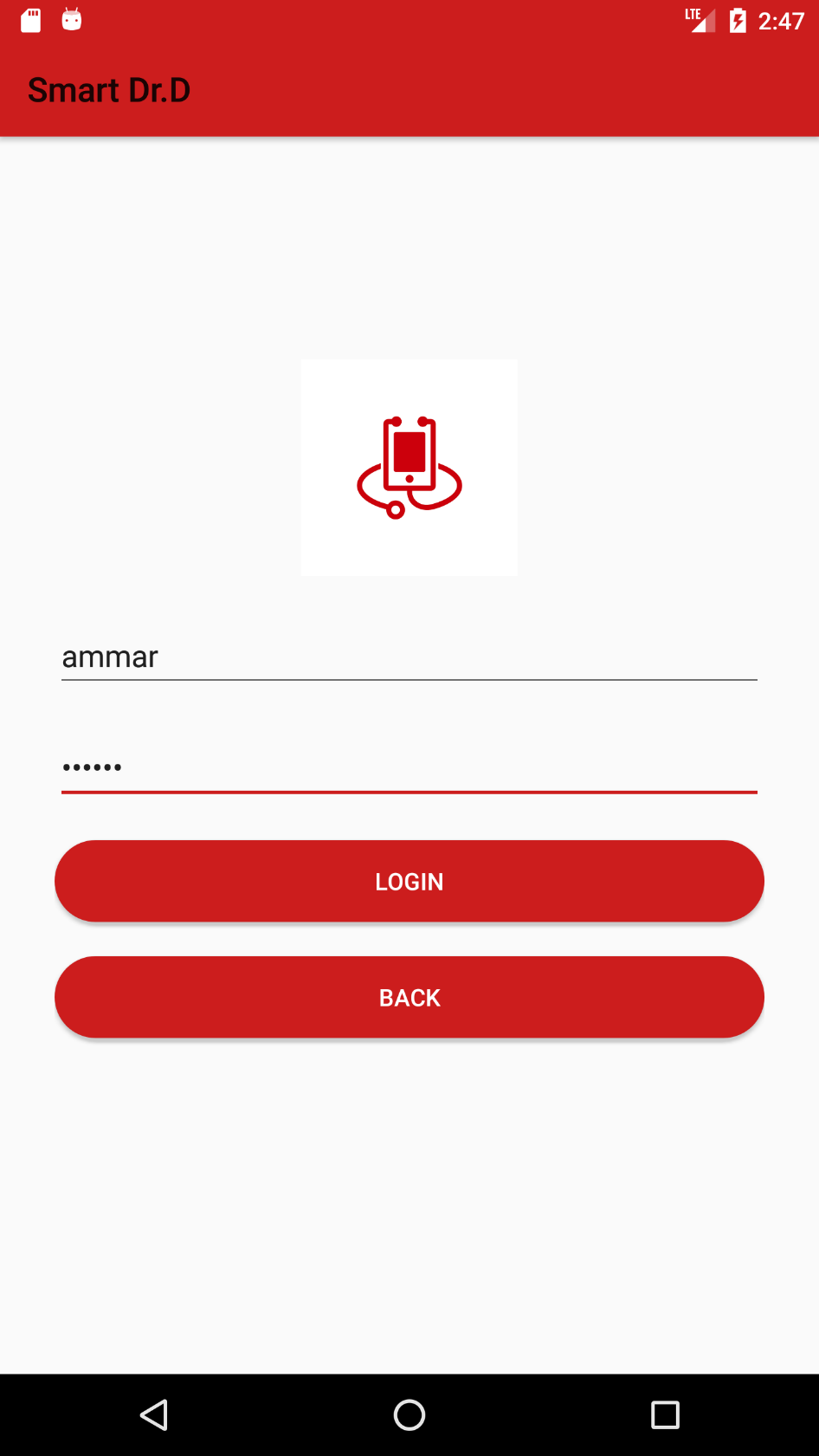


Figure 13 Add Account for login

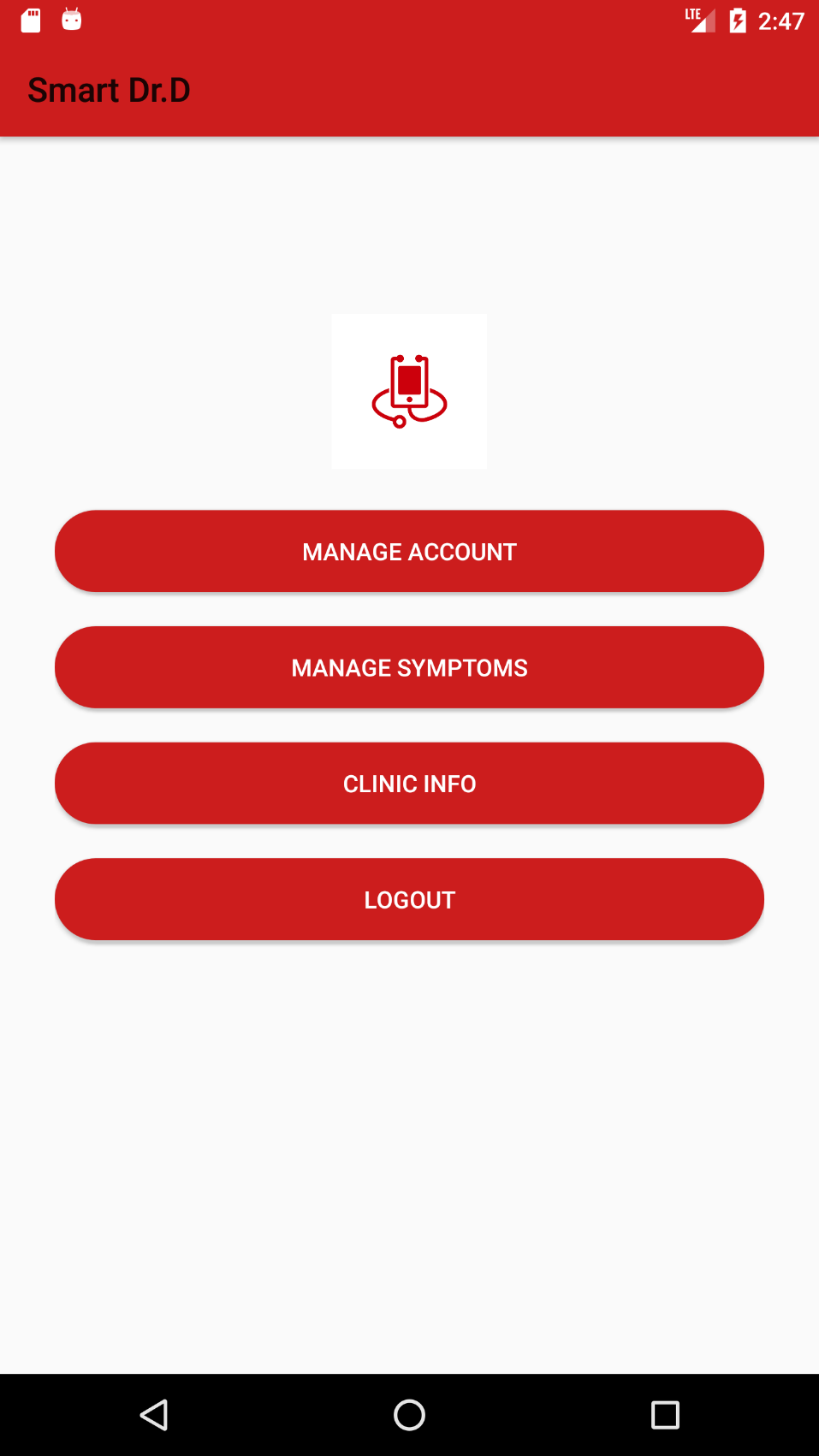


Figure 14 main Menu for Doctor

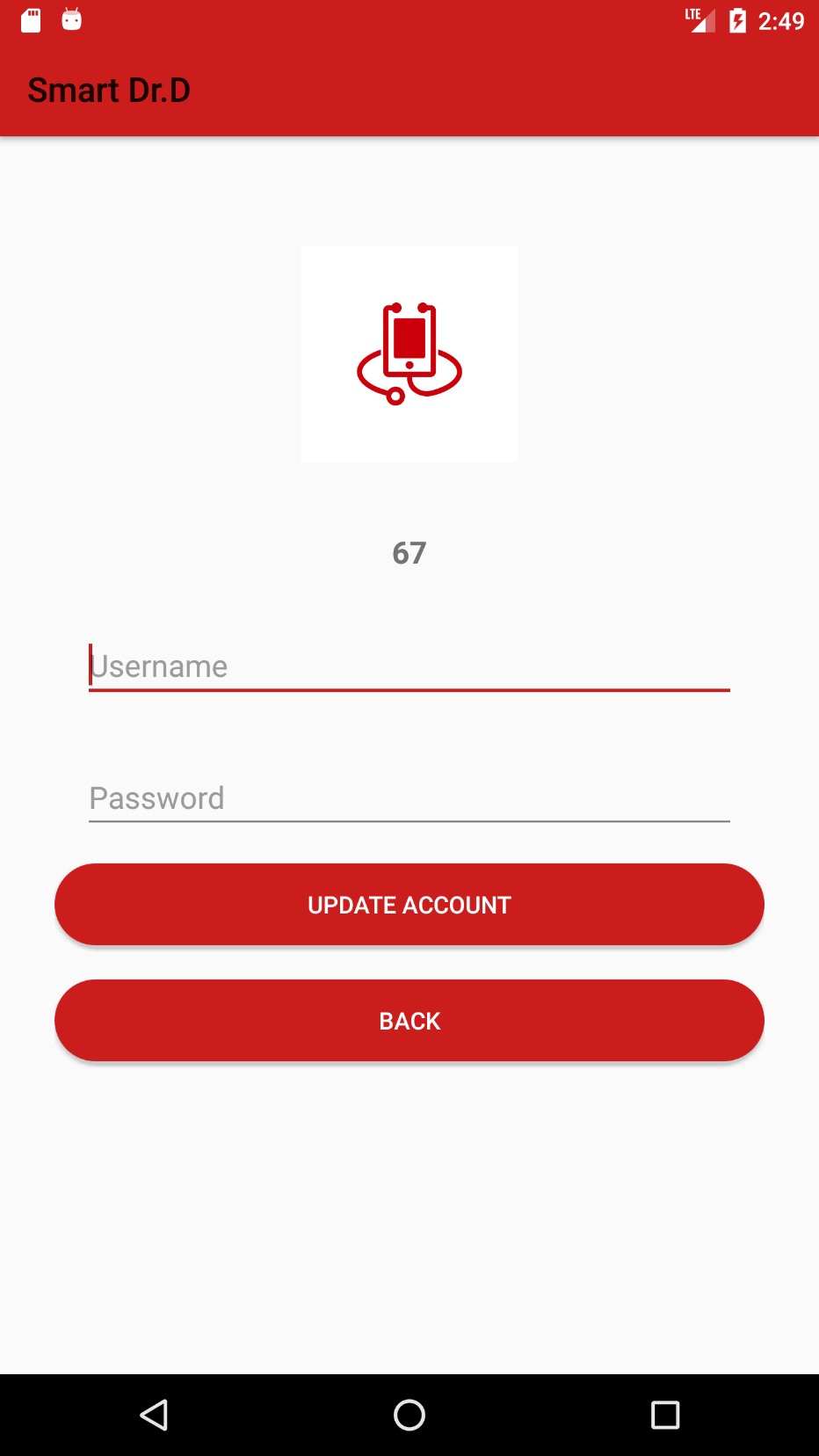


Figure 15 Update Account Information

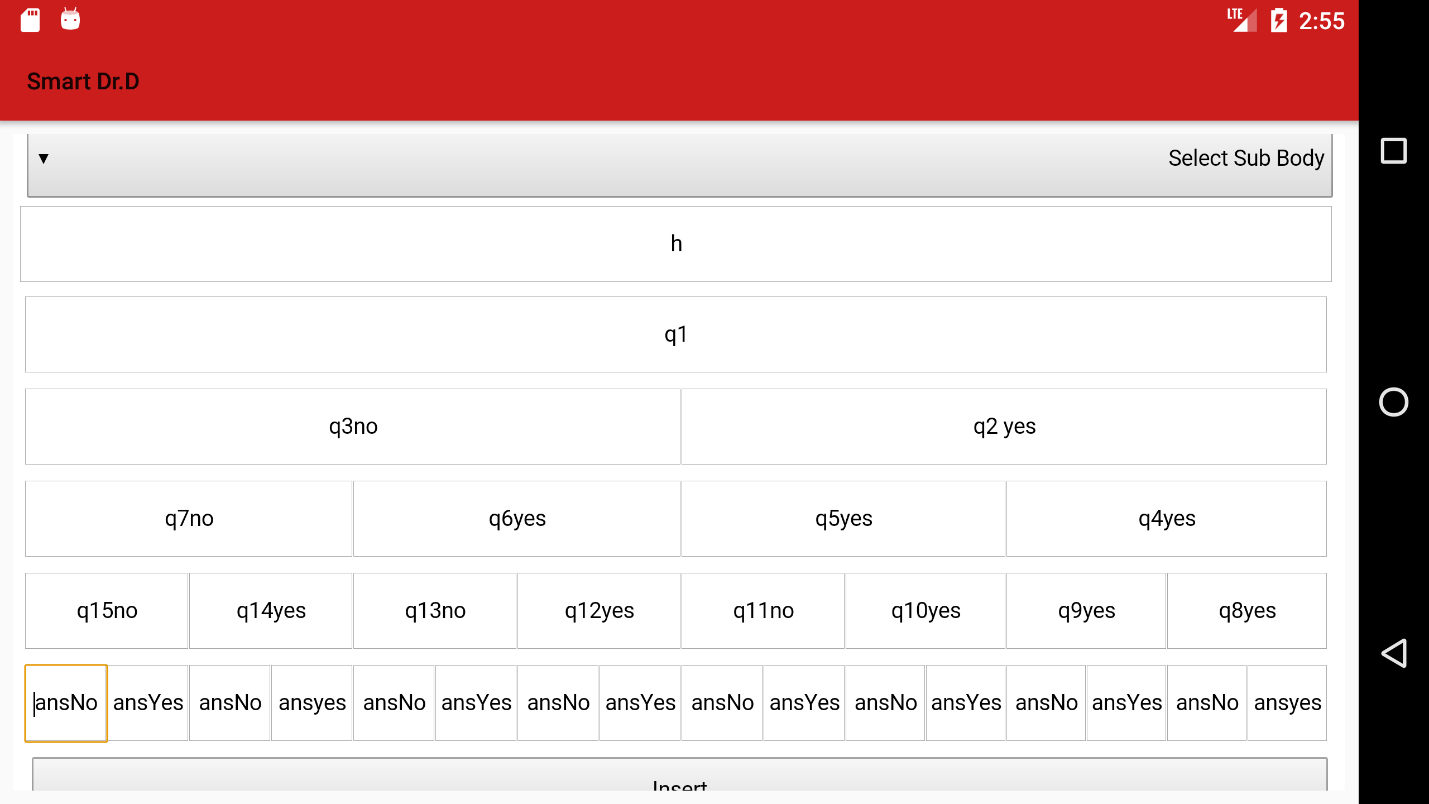


Figure 16 Add question and Reults

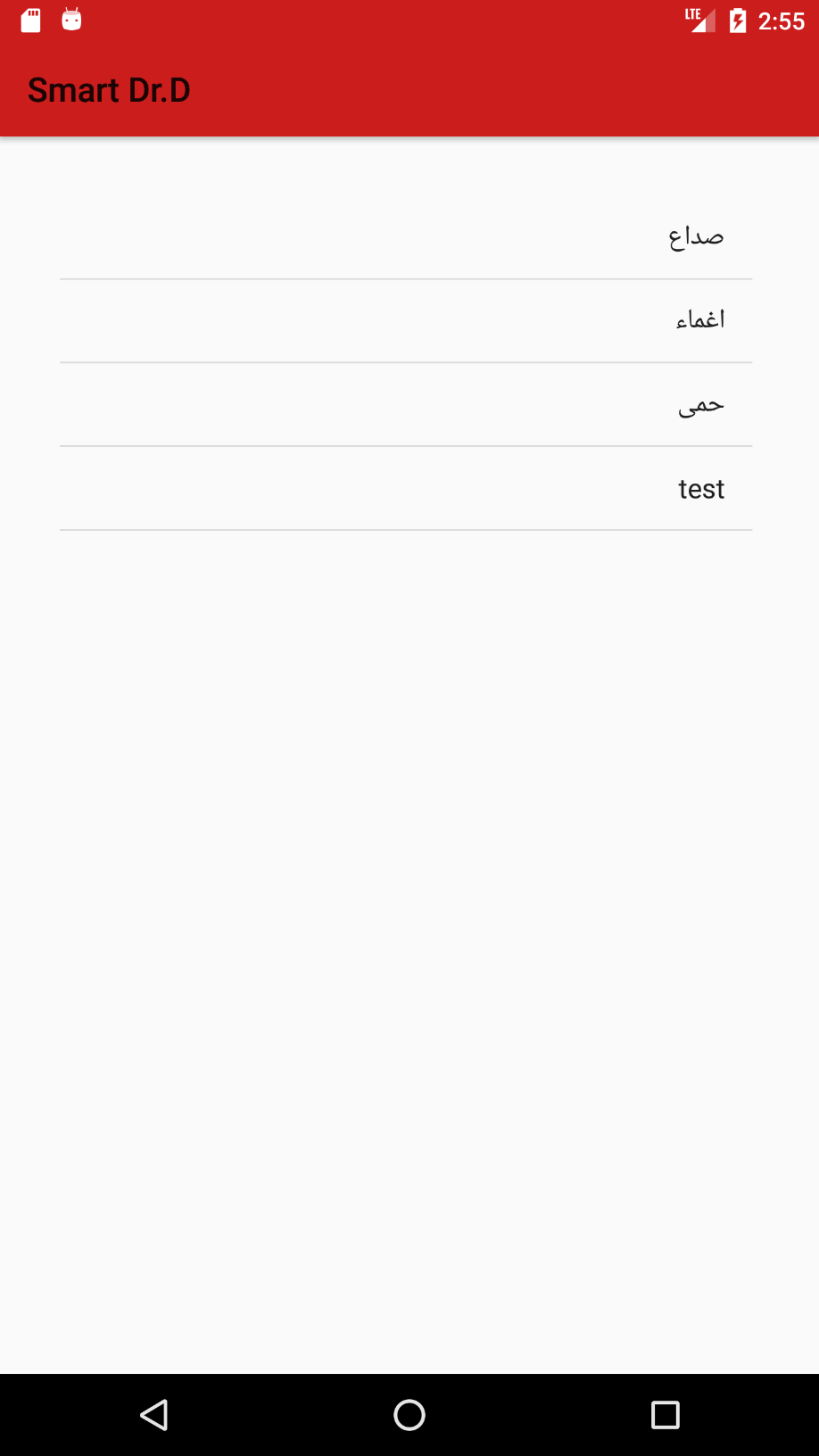


Figure 17 Select Disease by Patient



Figure 18 Show question by disease Selection



Figure 19 Show question by answer



Figure 20 show next question by previous answer



Figure 21 show last question by the tree of answers

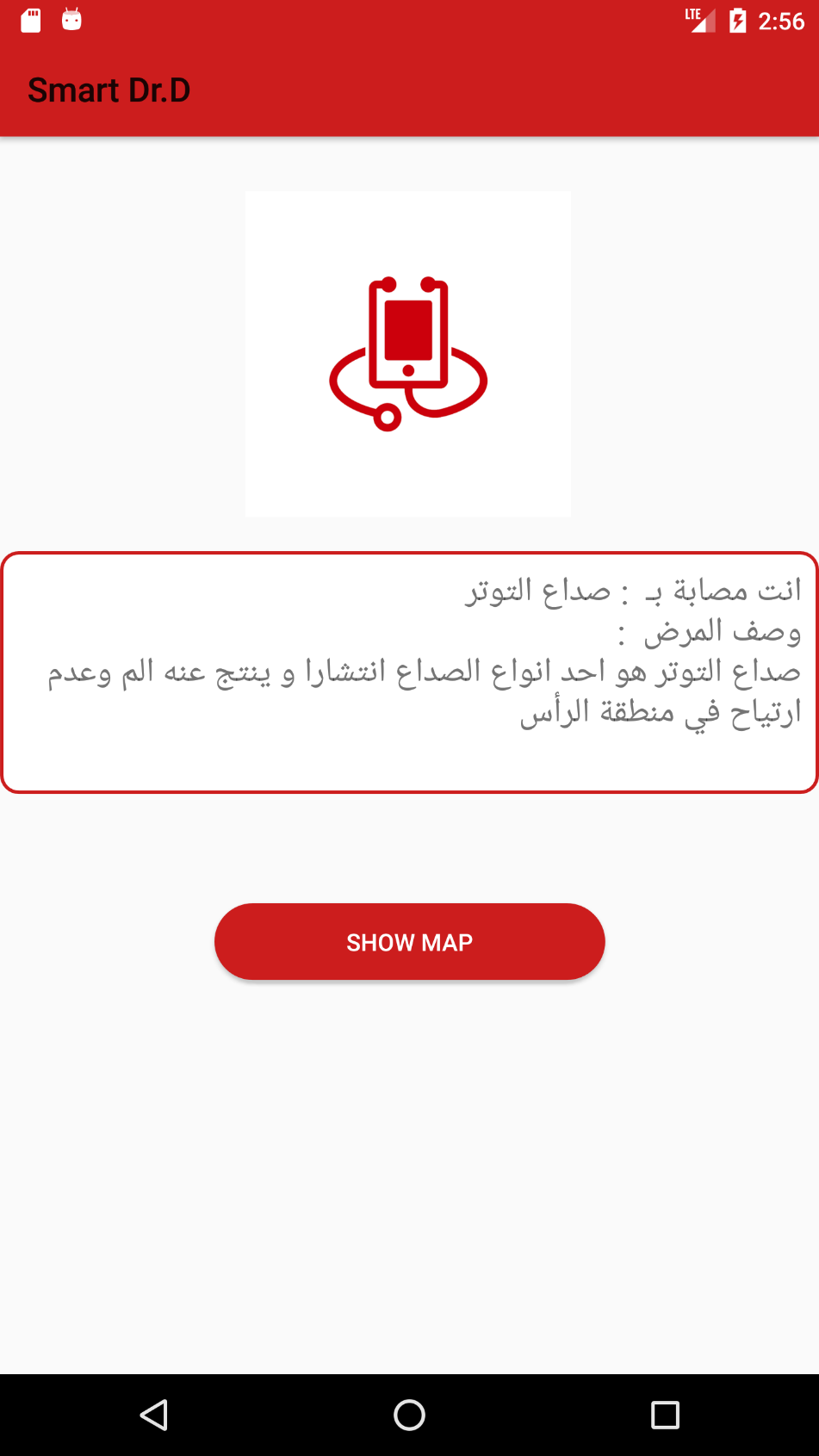


Figure 22 show symptoms diagnosis

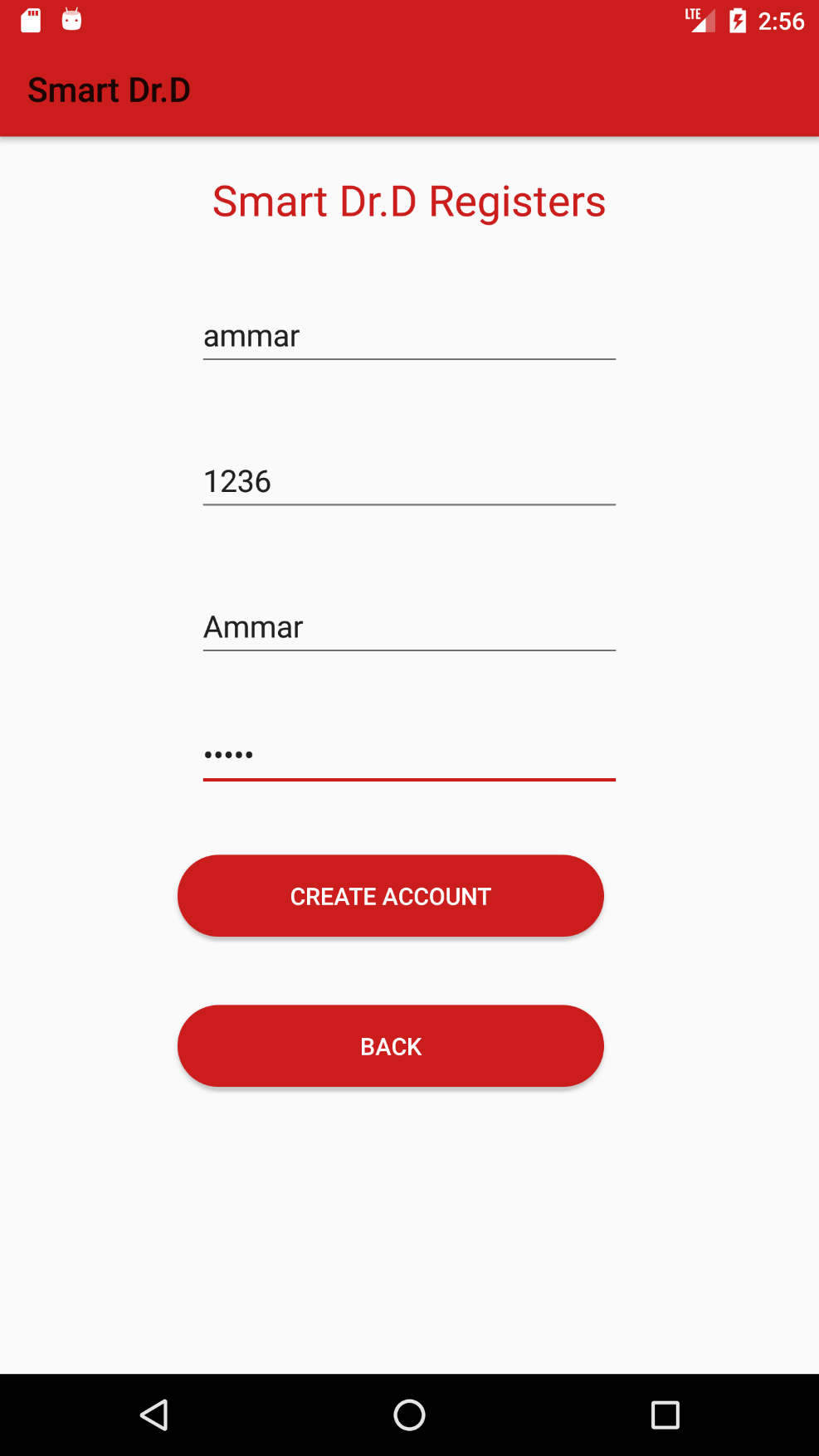
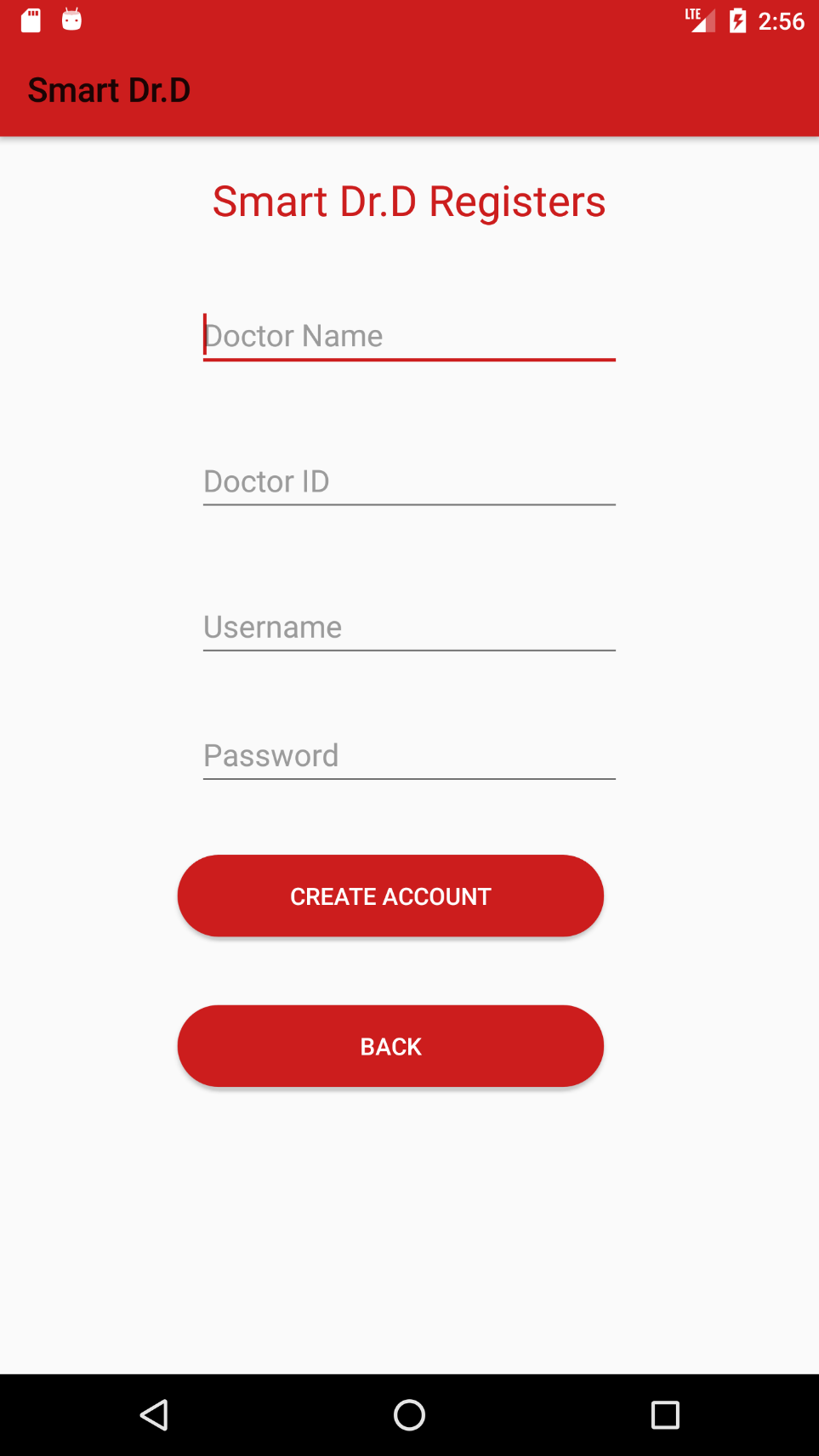


Figure 23 add data to doctor register

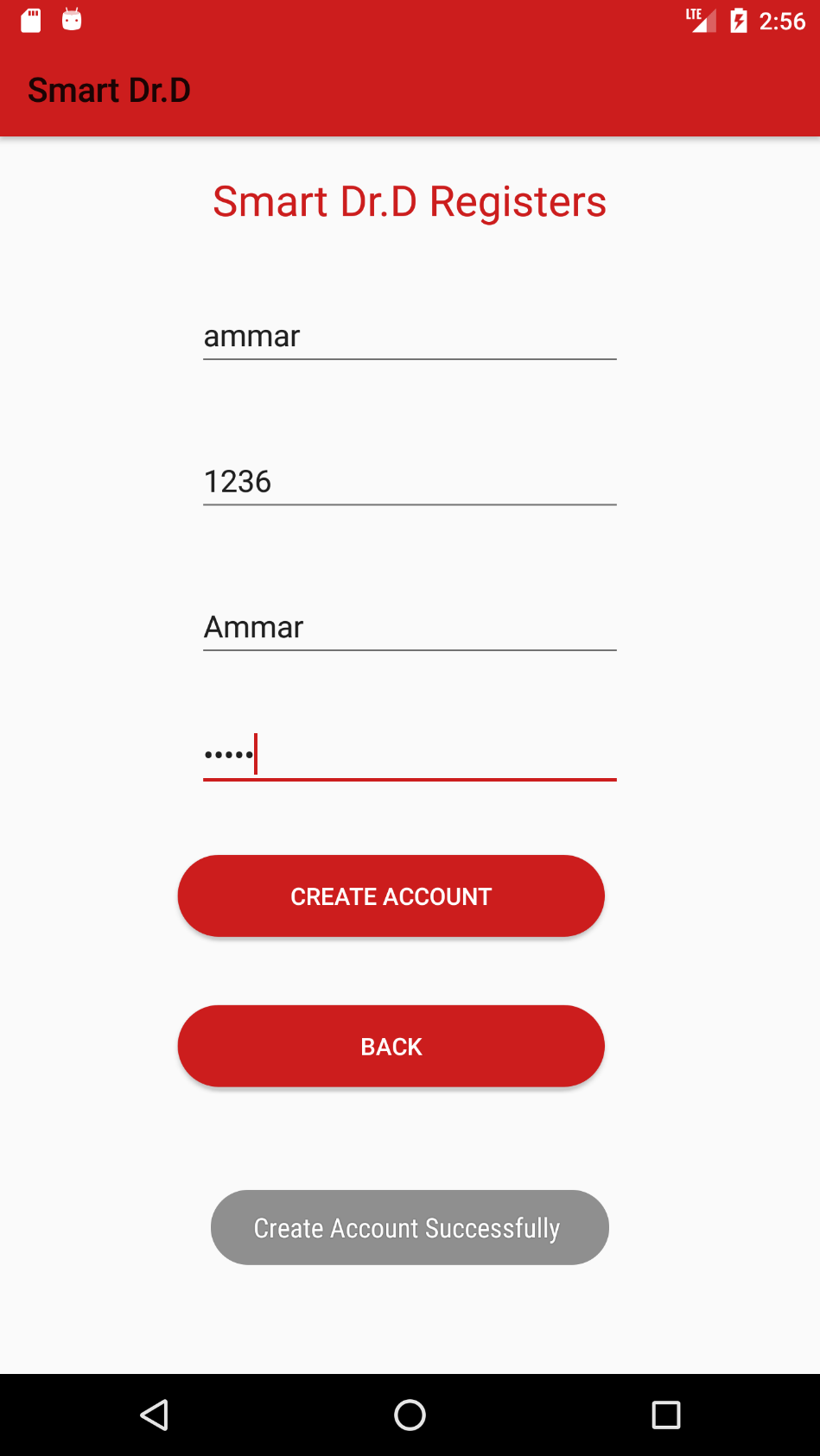


Figure 24 Show message when register successfully

# References

1. https://symptoms.webmd.com/default.htm#/info

https://www.kidney.org/apps/patients/care-after-kidney-transplant-app

1. https://www.kidney.org/apps/patients/gout-central

https://www.kidney.org/apps/professionals/advances-chronic-kidney-disease-ackd

http://ruralinstitute.umt.edu/371-milestone-tracker-app