EZMedi

SWE Final Presentation

Group 4: Feiluan, Tony, Jacqueline, Sally

Content

01 Introduction

02 Project Scope

03 Requirements

04 Design & Implementation

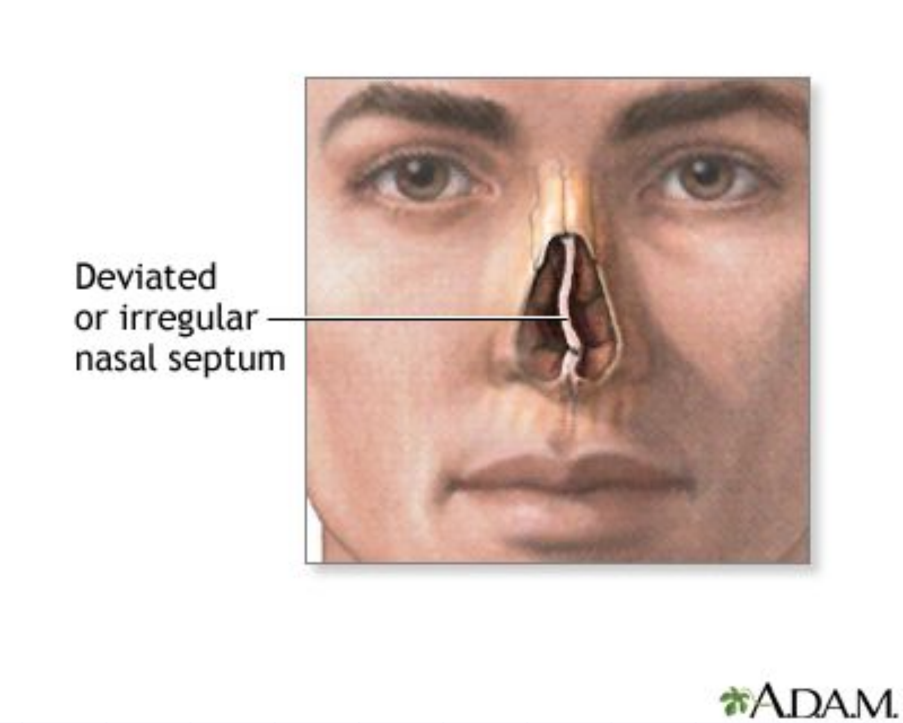
05 Testing & Demo

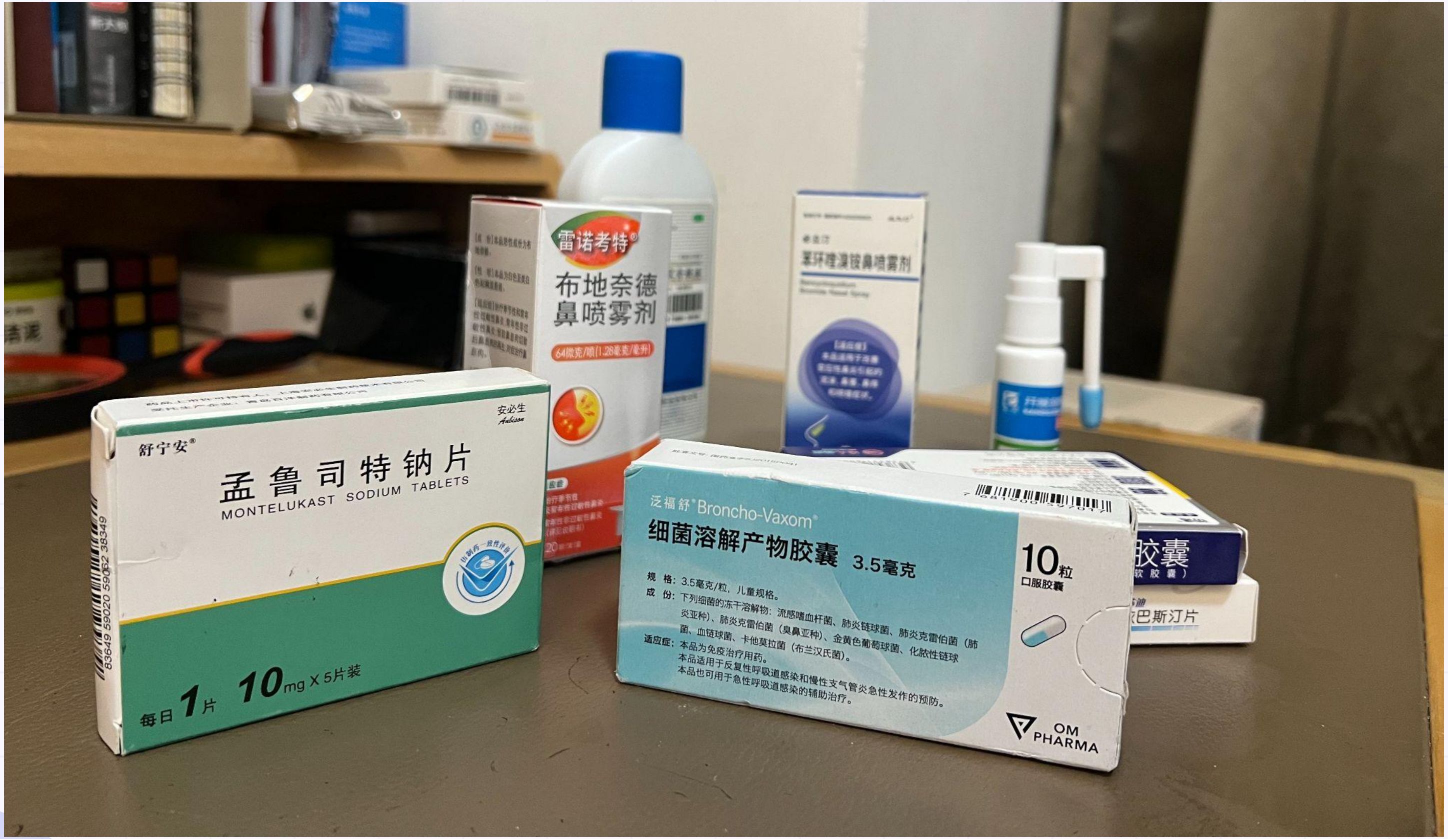
06 Improvements & Conclusion

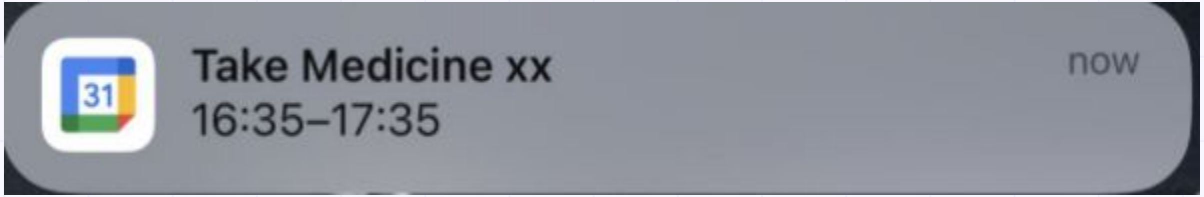
Introduction

Why Did We Develop Such an Application?

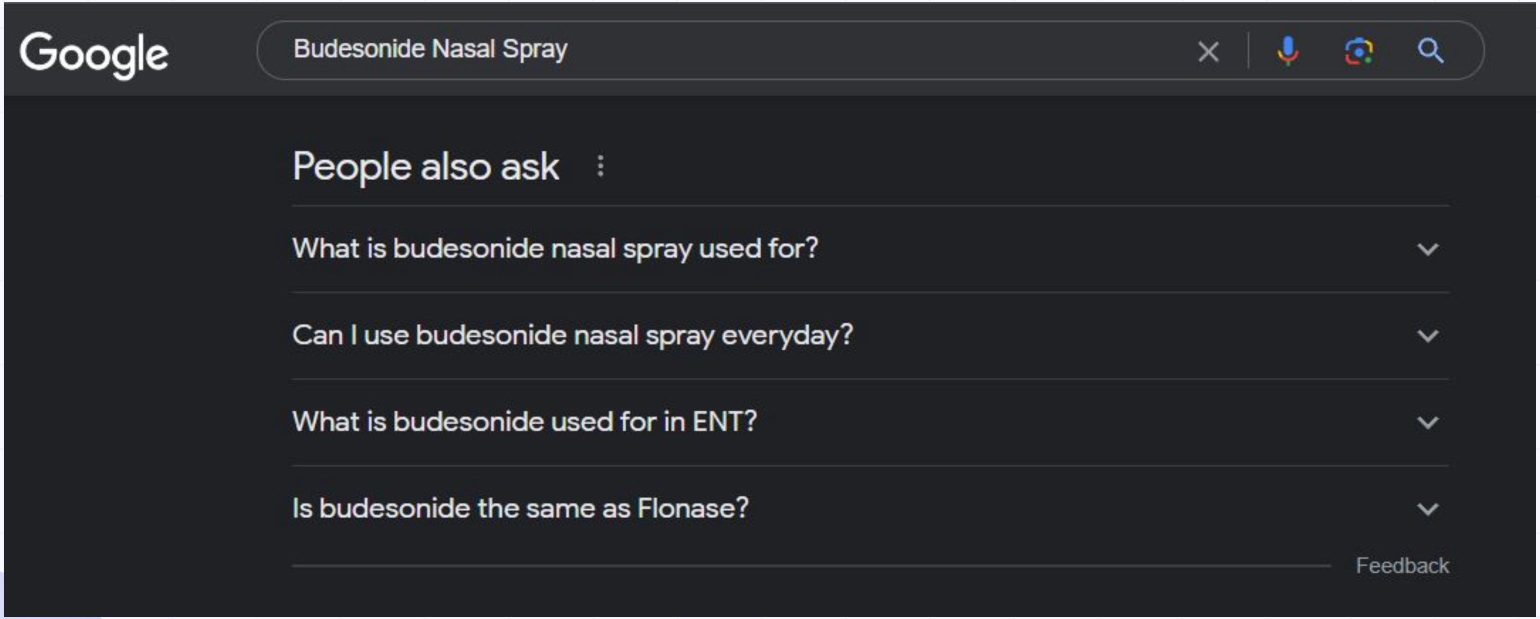
Introduction




舒宁安®
安必生
Aubison
孟鲁司特钠片
MONTELUKAST SODIUM TABLETS
每日1片 10mg X 5片装
雷诺考特®
布地奈德鼻喷雾剂
64微克/喷(1.28毫克/毫升)
苯环喹溴铵鼻喷雾剂
泛福舒® Broncho-Vaxom®
细菌溶解产物胶囊 3.5毫克
规 格：3.5毫克/粒，儿童规格。
成 份：下列细菌的冻干溶解物：流感嗜血杆菌、肺炎链球菌、肺炎克雷伯菌（肺炎亚种）、肺炎克雷伯菌（臭鼻亚种）、金黄色葡萄球菌、化脓性链球菌、血链球菌、卡他莫拉菌（布兰汉氏菌）。
适应症：本品为免疫治疗用药。
本品适用于反复性呼吸道感染和慢性支气管炎急性发作的预防。
本品也可用于急性呼吸道感染的辅助治疗。
10粒
口服胶囊
OM PHARMA
胶囊
巴斯汀片



Take Medicine xx
16:35–17:35



Budesonide Nasal Spray

People also ask

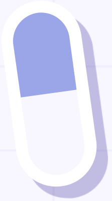
What is budesonide nasal spray used for?

Can I use budesonide nasal spray everyday?

What is budesonide used for in ENT?

Is budesonide the same as Flonase?

Why not design an application that achieve both?



EZMedi

Empower Your Wellness Journey

Project Scope

Objectives

The system aims to effectively allow patients with chronic diseases to better manage their medications and **facilitate the process of medicine-taking.**

Registered users will be able to update the patient’s personal medical profile as well as generate reminders when needed. The system is primarily intended to **connect users with a list of their own medications and reminders.**

Scope

Integration with API & Databases

RxNorm Medicine Database
Google Firebase

Limitations

Regulatory
Hardware
Control Functions

Reliability
Accessibility
External Interfaces

Requirements

Requirements

User Requirements

Functional Requirements

Non-Functional Requirements

User Requirements

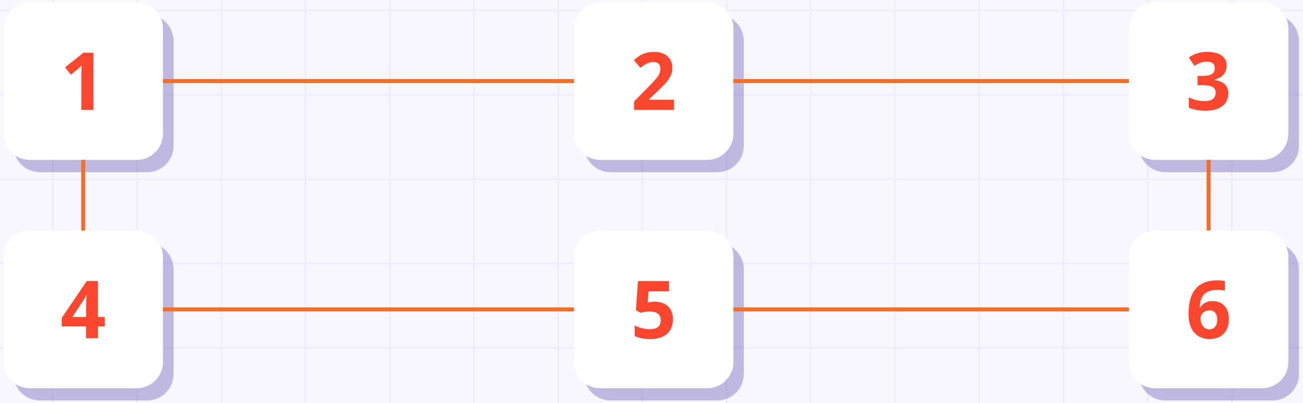
- **Simple and Intuitive User Interface**
- *Limited User Technical Knowledge*

- **Accessible Design Features**
- *Possible High Proportion of Users with Disabilities*

- **Language Considerations for UAE Users**
- *Varying Levels of English Proficiency*

- **Provides All Necessary Medication Information**
- *Varying Educational Backgrounds*

Functional Requirements



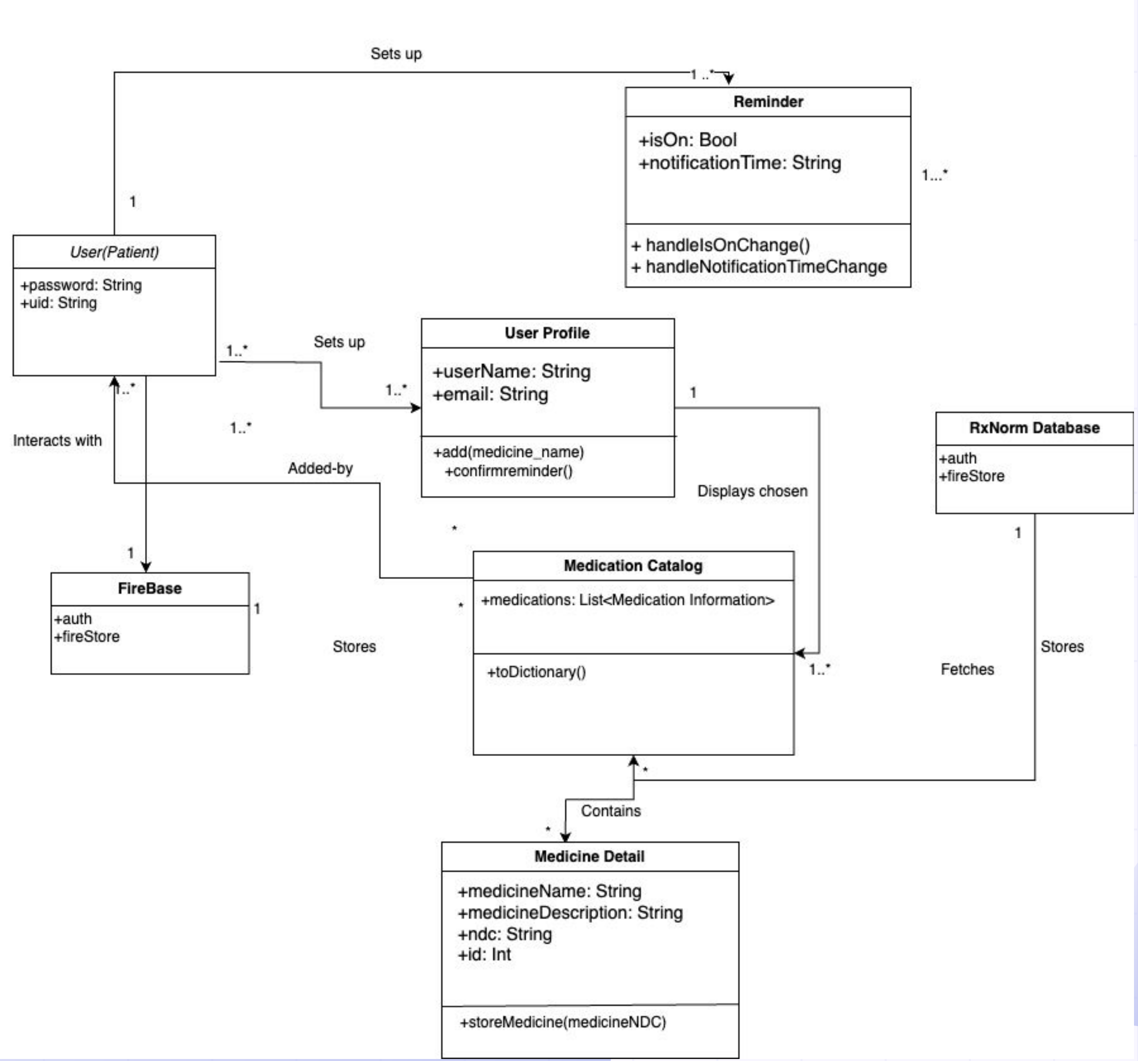
1. User Authentication
2. Medicine Identification
3. Medicine Search
4. Medicine Reminders
5. Medication Input
6. Personal Profile Management

Non-functional Requirements

Performance	● Support up to 100 concurrent users. ● Handle a load of 40–80 transactions per minute. ● Quick response times (e.g. shall identify the medicine through scanning the medicine’s QR code in at most 0.5 seconds).
Security and Privacy	● Compliance with HIPAA, ISO/IEC 27001, HON code principles, and ISO 13485:2016. ● Data privacy measures to protect user information.
Reliability	● 24/7 availability with minimal downtime. ● Fault tolerance and incident response procedures (e.g. When the system unexpectedly shuts down, the finished search is already stored in the database).
Usability	● User-friendly interface. ● The system should be designed to work on various devices with different iOS systems, suitable for different iPhone screen sizes and iOS systems after iOS12.
Effectiveness	● The system shall accurately identify medications through barcode scanning with a success rate of at least 90%.

Design & Implementation

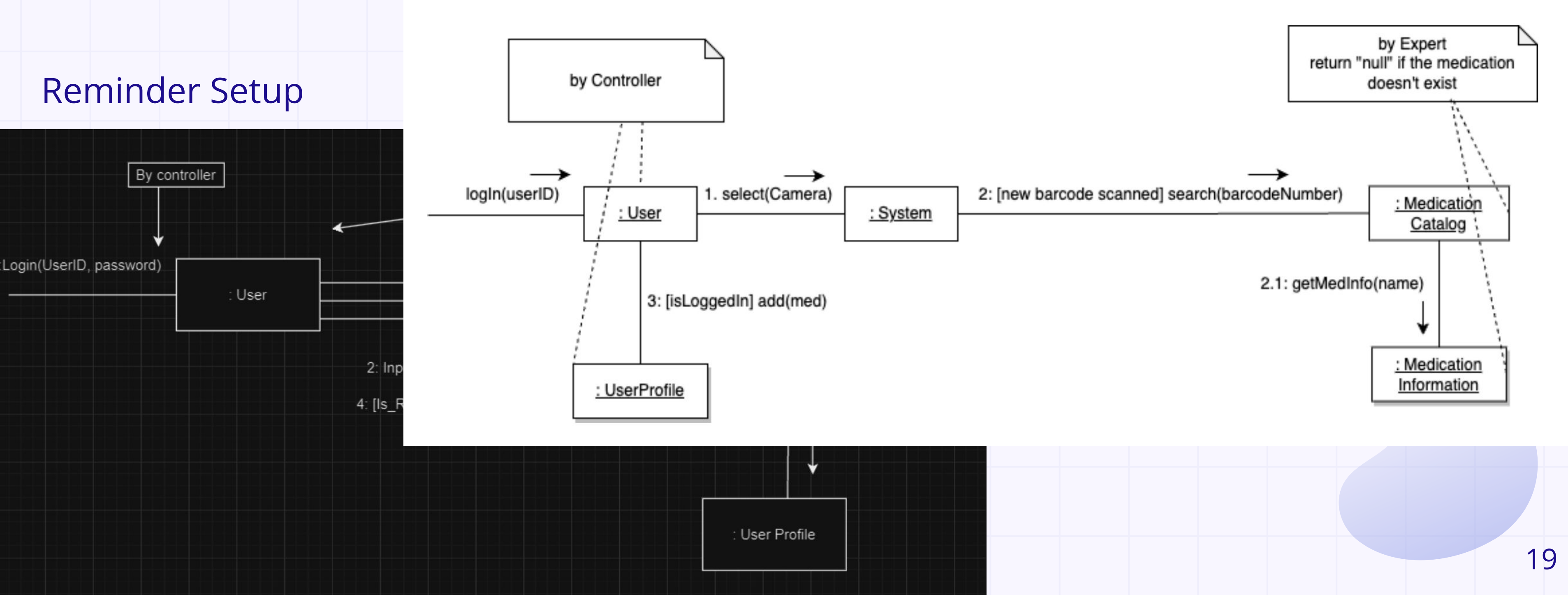
Class Diagram



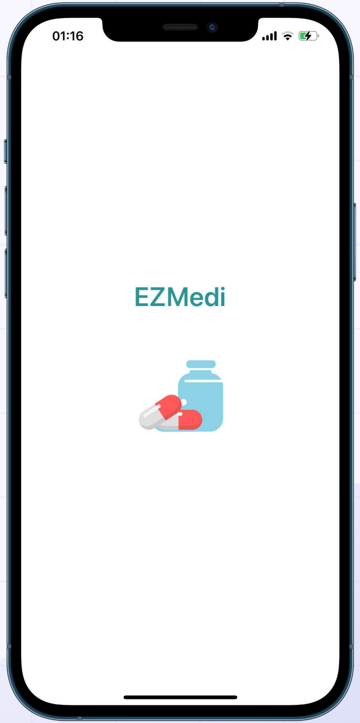
Interaction Diagram

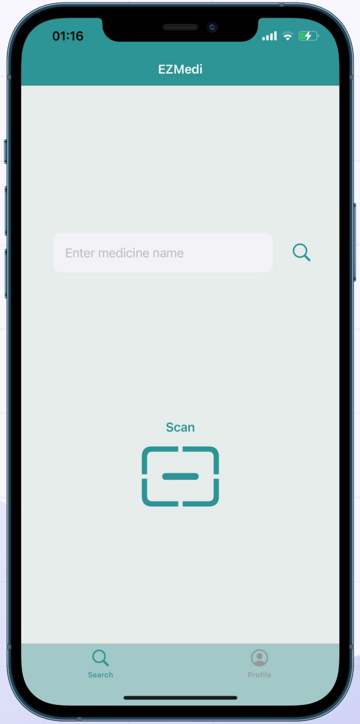
Search Medicine by Scanning Barcode

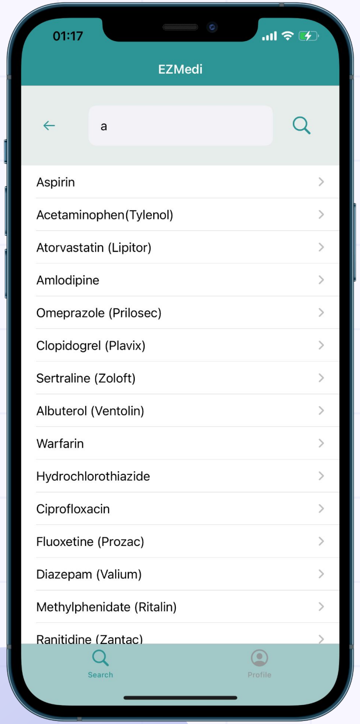
Reminder Setup



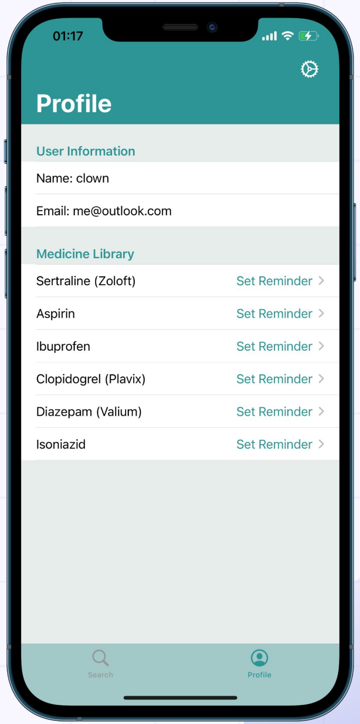
User Interface Design







- Aspirin
- Acetaminophen(Tylenol)
- Atorvastatin (Lipitor)
- Amlodipine
- Omeprazole (Prilosec)
- Clopidogrel (Plavix)
- Sertraline (Zoloft)
- Albuterol (Ventolin)
- Warfarin
- Hydrochlorothiazide
- Ciprofloxacin
- Fluoxetine (Prozac)
- Diazepam (Valium)
- Methylphenidate (Ritalin)
- Ranitidine (Zantac)



Profile

User Information

Name: clown

Email: me@outlook.com

Medicine Library

Sertraline (Zoloft)	Set Reminder
Aspirin	Set Reminder
Ibuprofen	Set Reminder
Clopidogrel (Plavix)	Set Reminder
Diazepam (Valium)	Set Reminder
Isoniazid	Set Reminder

Development Tool

XCode

Swift

FireBase

RxNorm

Challenges

- **Database**

  Firebase: maintain user logged in status

  RxNorm API Fetching

- **UI Design**

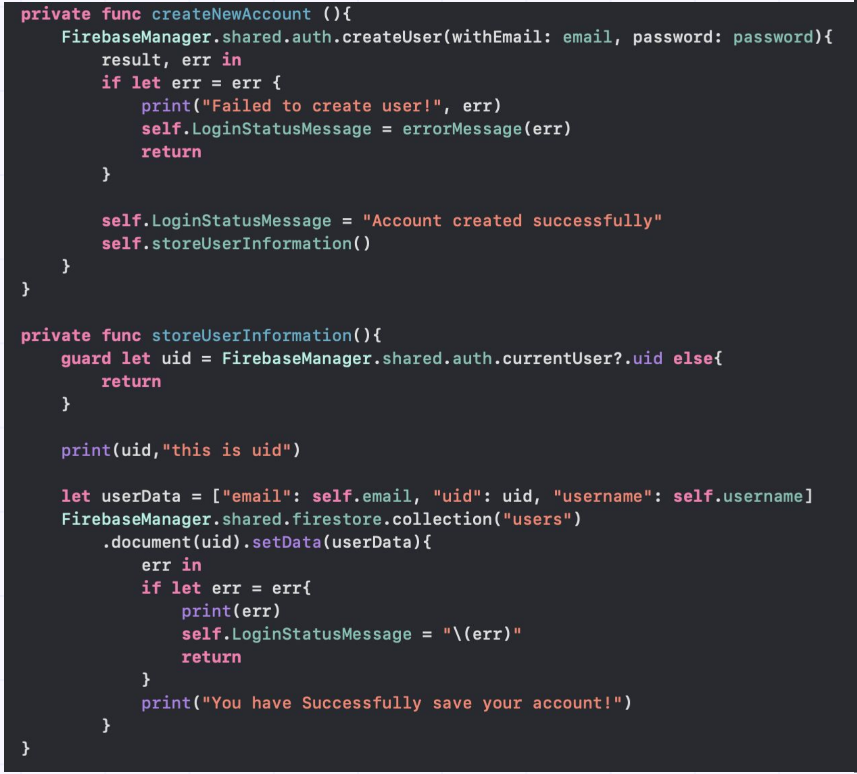
  User Experience

  Design consistency for different iPhone models

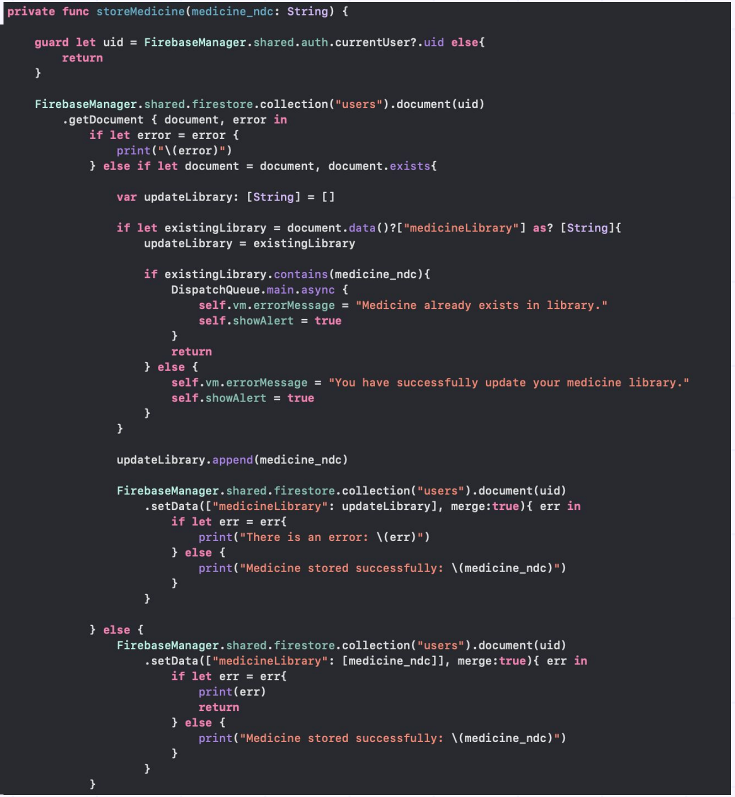
  Adherence to Standards

- **New Programming Language**

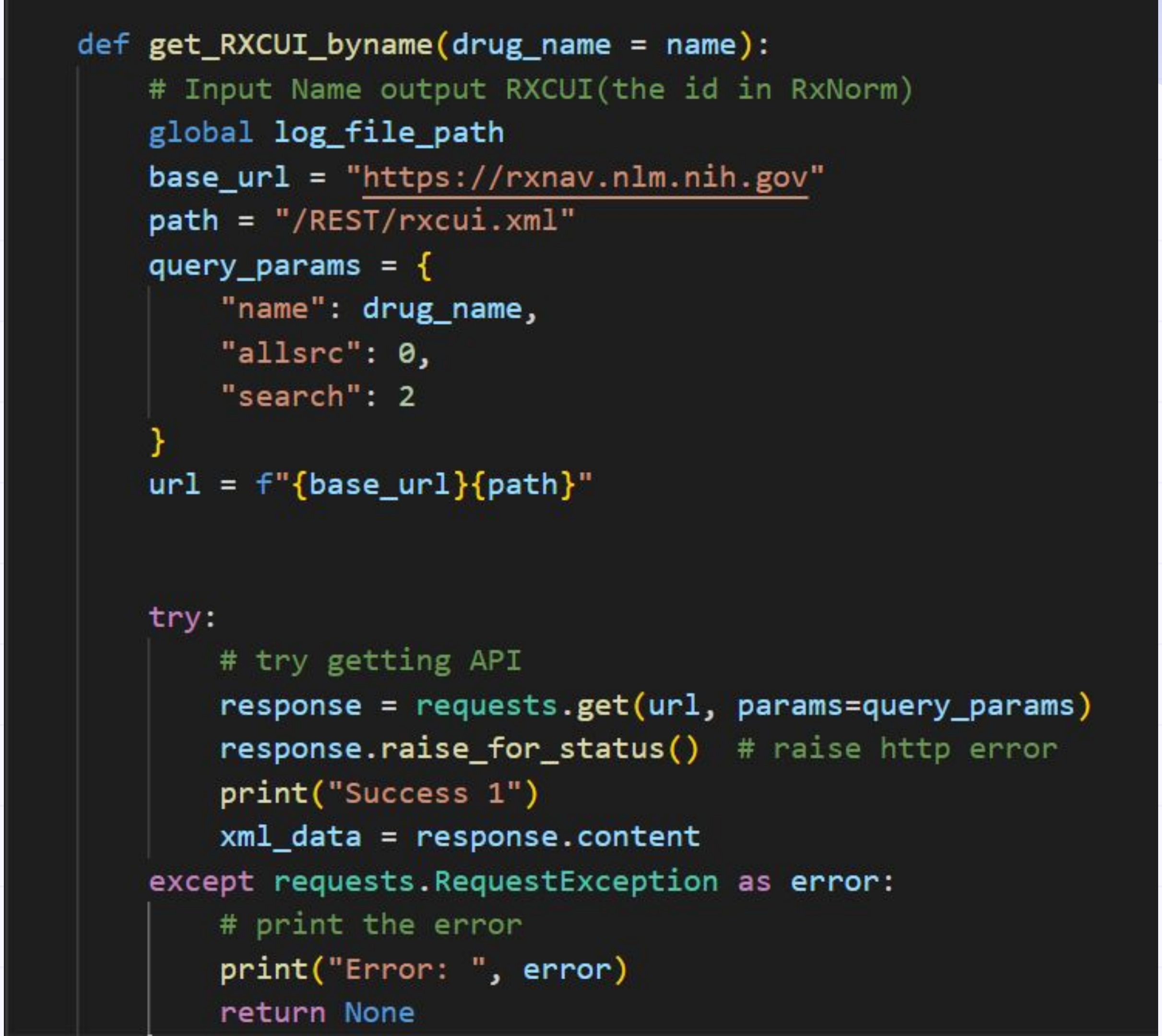
Code Highlights



```swift
private func createNewAccount (){
    FirebaseManager.shared.auth.createUser(withEmail: email, password: password){
        result, err in
        if let err = err {
            print("Failed to create user!", err)
            self.LoginStatusMessage = errorMessage(err)
            return
        }

        self.LoginStatusMessage = "Account created successfully"
        self.storeUserInformation()
    }
}

private func storeUserInformation(){
    guard let uid = FirebaseManager.shared.auth.currentUser?.uid else{
        return
    }

    print(uid,"this is uid")

    let userData = ["email": self.email, "uid": uid, "username": self.username]
    FirebaseManager.shared.firestore.collection("users")
        .document(uid).setData(userData){
            err in
            if let err = err{
                print(err)
                self.LoginStatusMessage = "\(err)"
                return
            }
            print("You have Successfully save your account!")
        }
}
```

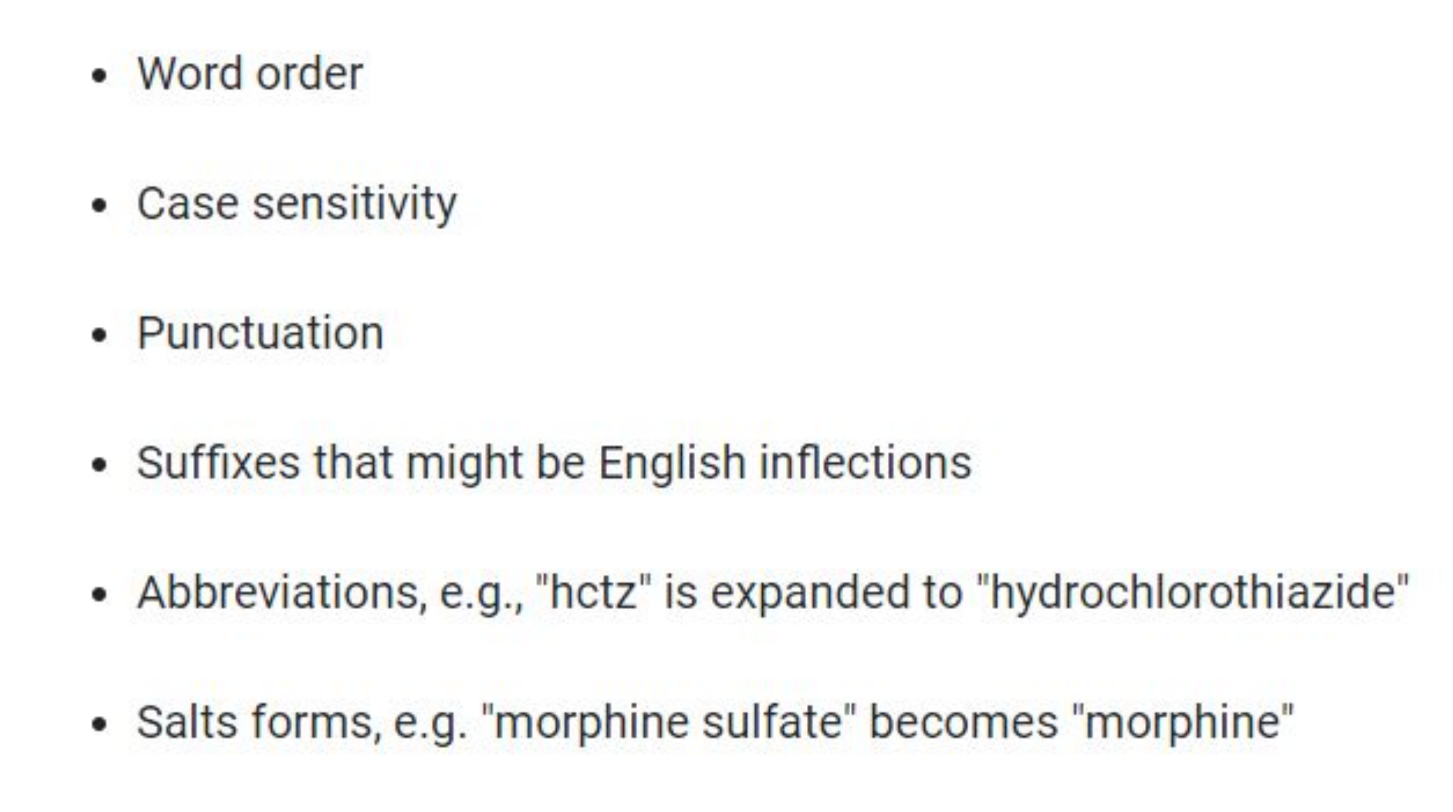


```swift
private func storeMedicine(medicine_ndc: String) {

    guard let uid = FirebaseManager.shared.auth.currentUser?.uid else{
        return
    }

    FirebaseManager.shared.firestore.collection("users").document(uid)
        .getDocument { document, error in
            if let error = error {
                print("\(error)")
            } else if let document = document, document.exists{

                var updateLibrary: [String] = []

                if let existingLibrary = document.data()?["medicineLibrary"] as? [String]{
                    updateLibrary = existingLibrary

                    if existingLibrary.contains(medicine_ndc){
                        DispatchQueue.main.async {
                            self.vm.errorMessage = "Medicine already exists in library."
                            self.showAlert = true
                        }
                        return
                    } else {
                        self.vm.errorMessage = "You have successfully update your medicine library."
                        self.showAlert = true
                    }
                }

                updateLibrary.append(medicine_ndc)

                FirebaseManager.shared.firestore.collection("users").document(uid)
                    .setData(["medicineLibrary": updateLibrary], merge:true){ err in
                        if let err = err{
                            print("There is an error: \(err)")
                        } else {
                            print("Medicine stored successfully: \(medicine_ndc)")
                        }
                    }

            } else {
                FirebaseManager.shared.firestore.collection("users").document(uid)
                    .setData(["medicineLibrary": [medicine_ndc]], merge:true){ err in
                        if let err = err{
                            print(err)
                            return
                        } else {
                            print("Medicine stored successfully: \(medicine_ndc)")
                        }
                    }
            }
```



```
def get_RXCUI_byname(drug_name = name):
    # Input Name output RXCUI(the id in RxNorm)
    global log_file_path
    base_url = "https://rxnav.nlm.nih.gov"
    path = "/REST/rxcui.xml"
    query_params = {
        "name": drug_name,
        "allsrc": 0,
        "search": 2
    }
    url = f"{base_url}{path}"


    try:
        # try getting API
        response = requests.get(url, params=query_params)
        response.raise_for_status()  # raise http error
        print("Success 1")
        xml_data = response.content
    except requests.RequestException as error:
        # print the error
        print("Error: ", error)
        return None
```

Code Highlights



- Word order
- Case sensitivity
- Punctuation
- Suffixes that might be English inflections
- Abbreviations, e.g., "hctz" is expanded to "hydrochlorothiazide"
- Salts forms, e.g. "morphine sulfate" becomes "morphine"

Testing

Test Design

Blackbox testing

Unit testing

Boundary testing

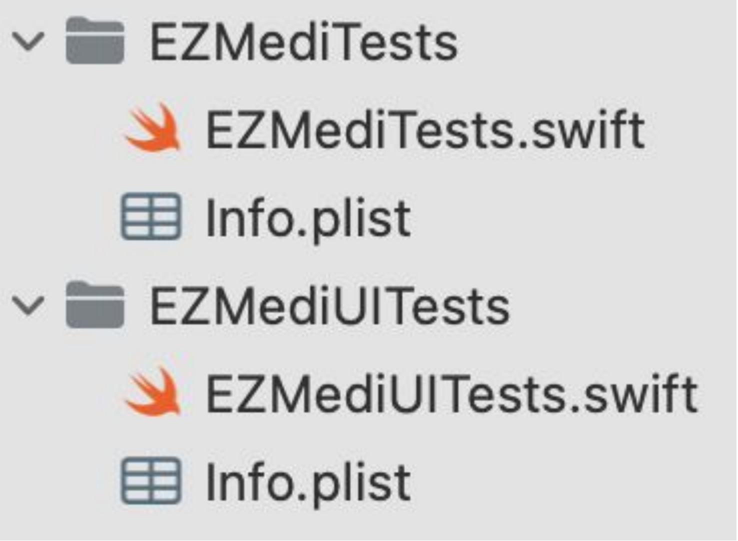
Equivalence Class Partitioning

White Box testing

System testing

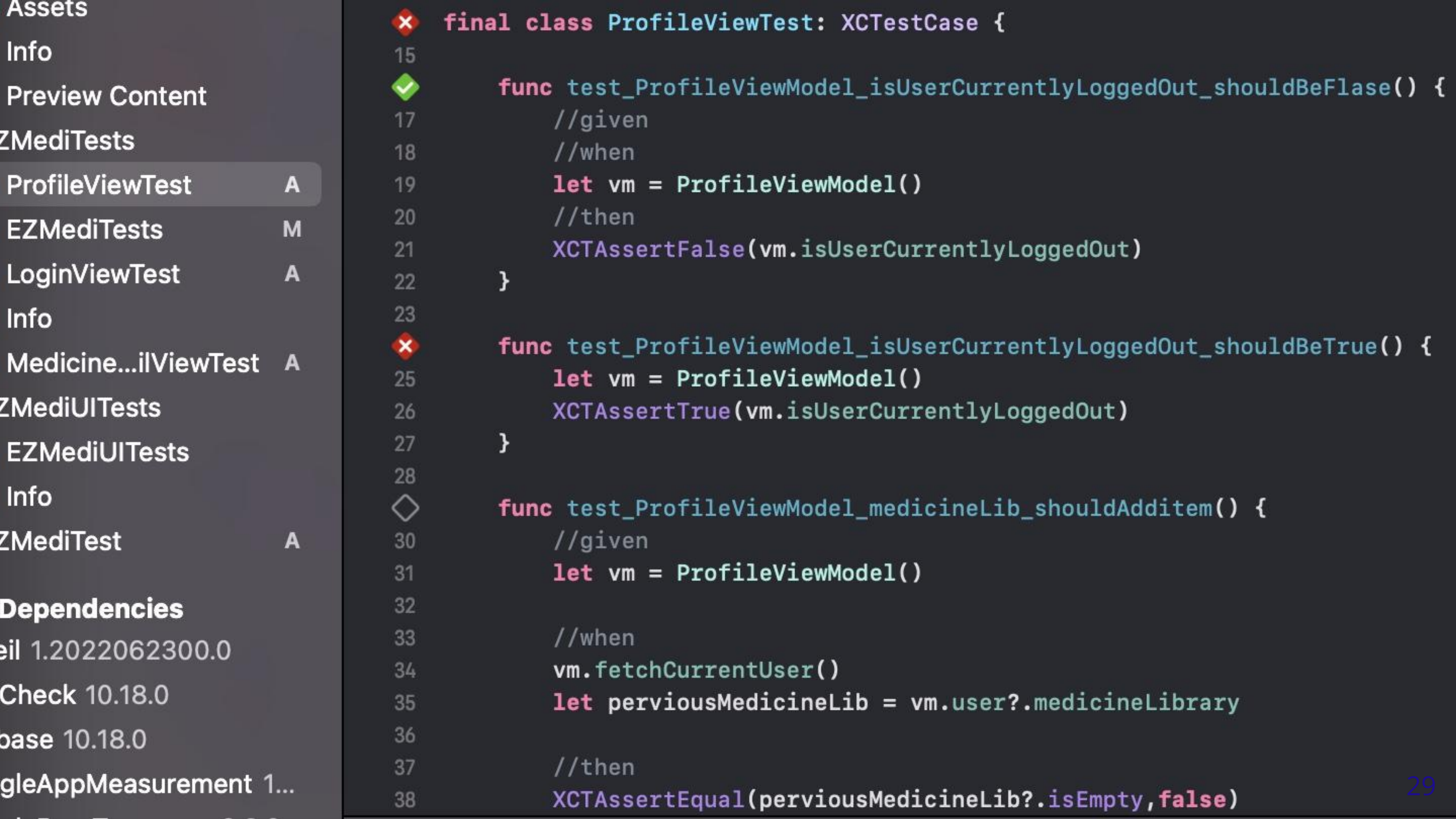
Unit testing

Testing

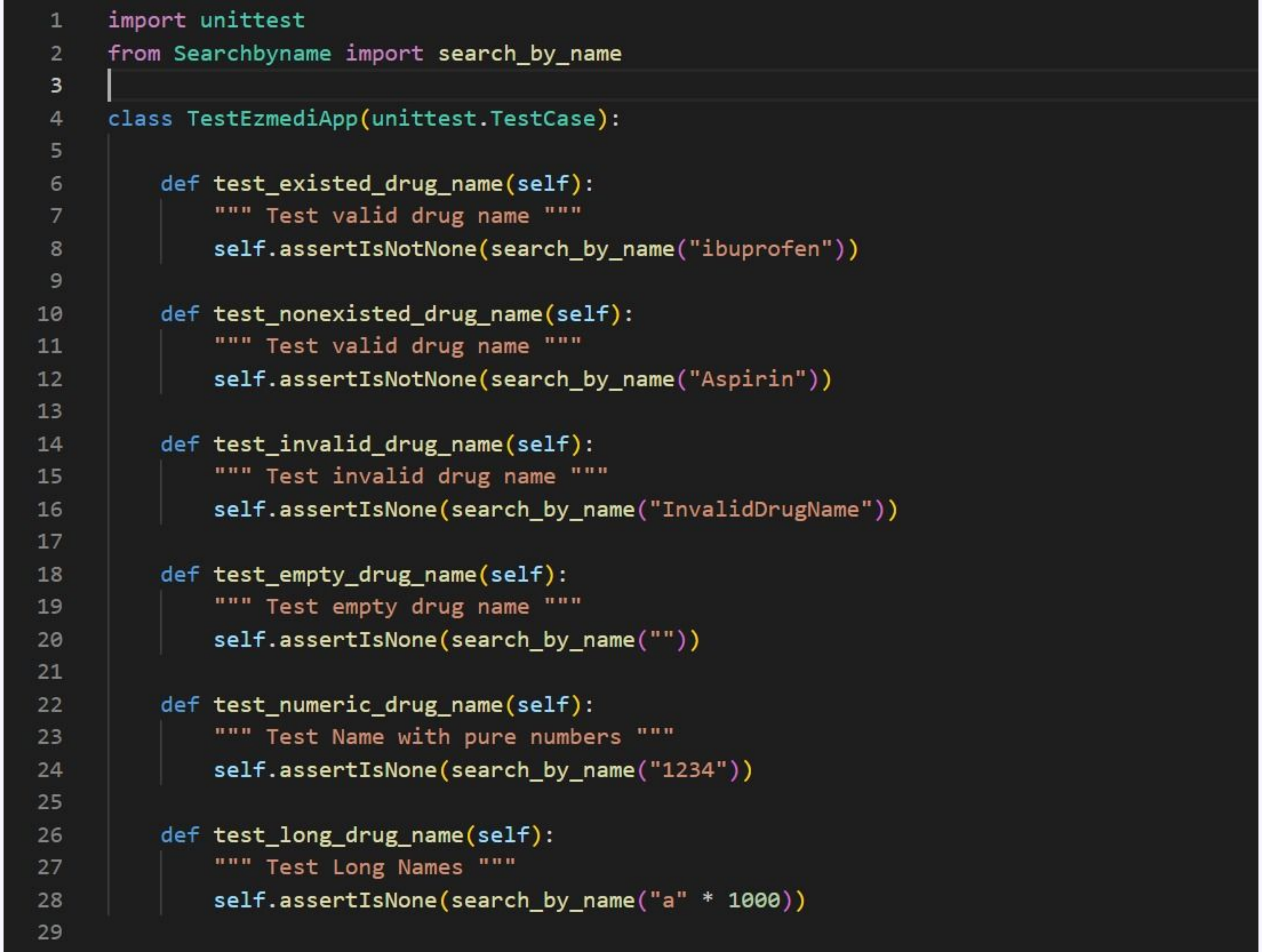


EZMediTests
- EZMediTests.swift
- Info.plist

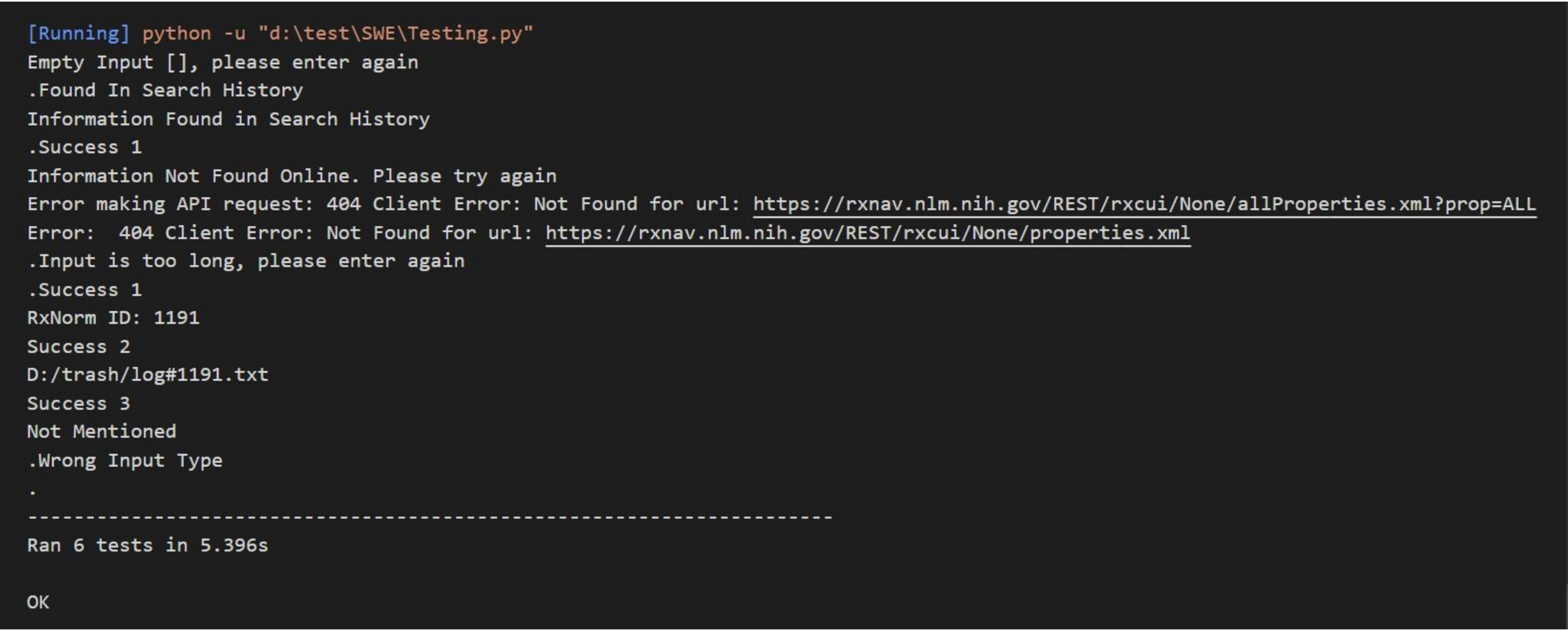
EZMediUITests
- EZMediUITests.swift
- Info.plist



```swift
final class ProfileViewTest: XCTestCase {

    func test_ProfileViewModel_isUserCurrentlyLoggedOut_shouldBeFlase() {
        //given
        //when
        let vm = ProfileViewModel()
        //then
        XCTAssertFalse(vm.isUserCurrentlyLoggedOut)
    }

    func test_ProfileViewModel_isUserCurrentlyLoggedOut_shouldBeTrue() {
        let vm = ProfileViewModel()
        XCTAssertTrue(vm.isUserCurrentlyLoggedOut)
    }

    func test_ProfileViewModel_medicineLib_shouldAdditem() {
        //given
        let vm = ProfileViewModel()

        //when
        vm.fetchCurrentUser()
        let perviousMedicineLib = vm.user?.medicineLibrary

        //then
        XCTAssertEqual(perviousMedicineLib?.isEmpty,false)
```



```python
import unittest
from Searchbyname import search_by_name

class TestEzmediApp(unittest.TestCase):

    def test_existed_drug_name(self):
        """ Test valid drug name """
        self.assertIsNotNone(search_by_name("ibuprofen"))

    def test_nonexisted_drug_name(self):
        """ Test valid drug name """
        self.assertIsNotNone(search_by_name("Aspirin"))

    def test_invalid_drug_name(self):
        """ Test invalid drug name """
        self.assertIsNone(search_by_name("InvalidDrugName"))

    def test_empty_drug_name(self):
        """ Test empty drug name """
        self.assertIsNone(search_by_name(""))

    def test_numeric_drug_name(self):
        """ Test Name with pure numbers """
        self.assertIsNone(search_by_name("1234"))

    def test_long_drug_name(self):
        """ Test Long Names """
        self.assertIsNone(search_by_name("a" * 1000))

```



```
[Running] python -u "d:\test\SWE\Testing.py"
Empty Input [], please enter again
.Found In Search History
Information Found in Search History
.Success 1
Information Not Found Online. Please try again
Error making API request: 404 Client Error: Not Found for url: https://rxnav.nlm.nih.gov/REST/rxcui/None/allProperties.xml?prop=ALL
Error:  404 Client Error: Not Found for url: https://rxnav.nlm.nih.gov/REST/rxcui/None/properties.xml
.Input is too long, please enter again
.Success 1
RxNorm ID: 1191
Success 2
D:/trash/log#1191.txt
Success 3
Not Mentioned
.Wrong Input Type
.
----------------------------------------------------------------------
Ran 6 tests in 5.396s

OK
```

Project Demo

Conclusion

Achievements/Future Work

Database

Create and develop Reminder class

RxNorm API successfully Fetching, connect python and swift

Functionality

Link NDC code to UPC-A barcode

Reminder within different range

Testing

Firebase emulator

Load Testing

Security Testing

Thank you

Questions?

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon** and infographics & images by **Freepik**