

KADIR HAS UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING



DESIGN AND IMPLEMENTATION OF A MOBILE PRESCRIPTION SYSTEM
WITH PATIENT-HEALTHCARE PROFESSIONAL INTERACTION

GRADUATE THESIS

ÇAĞDAŞ EGEMEN ÖZTÜRK

May, 2015

ÇAĞDAŞ EGEMEN ÖZTÜRK

M.S. Thesis

2015

DESIGN AND IMPLEMENTATION OF A MOBILE PRESCRIPTION SYSTEM
WITH PATIENT-HEALTHCARE PROFESSIONAL INTERACTION

ÇAĞDAŞ EGEMEN ÖZTÜRK

B.S., Computer Engineering, Kadir Has University, 2012

Submitted to the Graduate School of Science and Engineering
in partial fulfillment of the requirements for the degree of
Master of Science
in
Computer Engineering

KADIR HAS UNIVERSITY

May, 2015

KADIR HAS UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

DESIGN AND IMPLEMENTATION OF A MOBILE PRESCRIPTION SYSTEM
WITH PATIENT-HEALTHCARE PROFESSIONAL INTERACTION

ÇAĞDAŞ EGEMEN ÖZTÜRK

APPROVED BY:

Asst. Prof. Dr. Tamer Dağ Kadir Has University
(Thesis Supervisor)



Assoc. Prof. Dr. Zeki Bozkuş Kadir Has University



Asst. Prof. Dr. Tansal Güçlüoğlu Yıldız Technical University



APPROVAL DATE: May 13th, 2015

“I, ÇAĞDAŞ EGEMEN ÖZTÜRK, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.”



ÇAĞDAŞ EGEMEN ÖZTÜRK

ABSTRACT

DESIGN AND IMPLEMENTATION OF A MOBILE PRESCRIPTION SYSTEM WITH PATIENT-HEALTHCARE PROFESSIONAL INTERACTION

ÇAĞDAŞ EGEMEN ÖZTÜRK

Master of Science in Computer Engineering

Advisor: Asst. Prof. Dr. Tamer DAĞ

May, 2015

In this thesis, a mobile prescription reminder and scheduling system application with patient-healthcare professional interaction is designed and implemented. By using the application, different types of users are able to create and manage prescriptions and are reminded to take the medication based on the prescription data. Besides, users are able to reach the prospectuses of the drugs. One of the most important functionality of the application is to provide patient-healthcare professional interaction. By using the application, healthcare professionals can assign prescriptions to their patients and they can also monitor their medicine compliance. Since the medicine compliance is very important to have an effective treatment, the main goal of this application is to help people to take their medicines on time and find the prospectus information of the drugs easily by using their mobile phones.

Keywords: Mobile Application, iOS, Software Development, Healthcare Professional, Objective-C, Medicine Compliance, Gamification

ÖZET

HASTA VE SAĞLIK GÖREVLİSİ ETKİLEŞİMİ ÖZELLİĞİ BULUNAN MOBİL REÇETE HATIRLATMA VE PLANLAMA UYGULAMASI TASARIMI VE GELİŞTİRİMİ

ÇAĞDAŞ EGEMEN ÖZTÜRK

Bilgisayar Mühendisliği, Yüksek Lisans

Danışman: Yard. Doç. Dr. Tamer DAĞ

Mayıs, 2015

Bu tezde, hasta ve sağlık görevlisi etkileşimi özelliği bulunan mobil bir reçete hatırlatma ve planlama uygulaması tasarlanmış ve geliştirilmiştir. Farklı tipte kullanıcılar bu uygulamayı kullanarak reçete yaratıp düzenleyebilir ve reçetenin içeriğine göre tedavi zamanları hakkında uyarılır. Buna ek olarak kullanıcılar ilaçların prospektüs bilgilerine ulaşabilirler. Uygulamanın en önemli özelliklerinden birisi hasta ile sağlık görevlisi arasında etkileşimi sağlamasıdır. Sağlık görevlileri bu uygulamayı kullanarak hastalara reçete yazabilir ve onların ilaç kullanım düzenlerini takip edebilir. Düzenli olarak ilaç kullanmak verimli bir tedavi için çok önemli olduğundan, bu uygulamanın ana hedefi insanların ilaçlarını zamanında almalarına yardım etmek ve onların mobil telefonlarını kullanarak ilaçların prospektüs bilgilerine kolayca ulaşabilmelerini sağlamaktır.

Anahtar Kelimeler: Mobil uygulama, iOS, Yazılım Geliştirme, Sağlık Görevlisi, Objective-C, İlacı Düzenli Kullanmak, Oyunlaştırma

Acknowledgements

Foremost, I would like to thank to my advisor Asst. Prof. Tamer Dağ for the continuous support of my M.S. study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I learned a lot from him and I would like to owe my deepest gratitude to this wonderful supervisor.

I also want to express my appreciation for my girlfriend Ezgi Keskiner for her encouragement and constant support.

Finally, I am grateful to my family and all friends who supported me both during my studies and in writing my thesis. This journey would not have been possible without them.

Table of Contents

Abstract	i
Özet	ii
Acknowledgements	iii
Table of Contents	iv
List of Tables	vi
List of Figures	vii
List of Abbreviations	ix
1 Introduction	1
2 Introduction to Mobile Applications	4
3 Types of Mobile Applications	6
3.1 Mobile Web Applications	6
3.2 Mobile Native Applications	7
3.3 Hybrid Applications	7
4 Mobile Application Development Lifecycle	8
4.1 Inception	8
4.2 Design	9
4.3 Development	9
4.4 Testing and Stabilization	9
4.5 Deployment	10
5 Project Medicine Mobile Application Lifecycle	11
5.1 Inception	11
5.2 Design	13
5.3 Development	13
5.4 Testing and Stabilization	14
5.5 Deployment	14
5.6 Developing Mobile Applications for Apple Devices	14
5.6.1 Development	16
5.6.2 Ad-Hoc	16
5.6.3 Distribution	17

6	Project Medicine Application Design and Implementation	18
6.1	Database Design	18
6.2	Using PHP Web Service to do Remote Database Operations	20
6.2.1	A Brief explanation of the working mechanism of SSL ...	22
6.3	Screens and Functionalities of Project Medicine	22
6.3.1	User Perspectives	22
6.3.2	Landing Page of the Project Medicine	23
6.3.2.1	Registration	23
6.3.2.2	User Login	25
6.3.2.3	Find prospectus information of a drug.....	26
6.3.3	“Project Medicine” Main Menu	28
6.3.3.1	Find prospectus information of a drug	28
6.3.3.2	Create a New Prescription	29
6.3.3.3	Today’s Drugs	32
6.3.3.4	My Prescriptions	34
6.3.3.5	Settings	35
6.3.4	Differences between the Normal User and the Healthcare Professional	36
6.3.4.1	Create a New Prescription	37
6.3.4.2	My Patients	38
6.3.5	Gamification	39
6.3.5.1	Game Mechanic: Achievements	40
7	Conclusion and Recommendations for Future Work	42
	References	43
	Curriculum Vitae	46

List of Tables

Table 1.1 Comparison of existing applications.....	2
--	---

List of Figures

Figure 6.1 Database diagram	20
Figure 6.2 System architecture	21
Figure 6.3 Welcome screen	24
Figure 6.4 Registration screen.....	24
Figure 6.5 Login system flowchart.....	25
Figure 6.6 Login screen	26
Figure 6.7 Forgot password screen	26
Figure 6.8 Drug list screen	27
Figure 6.9 Drug detail screen	27
Figure 6.10 Patient main menu screen	28
Figure 6.11 Create a prescription screen (Patient)	29
Figure 6.12 Prescription item detail screen.....	29
Figure 6.13 System flowchart	30
Figure 6.14 Local push notification alert on lock screen	31
Figure 6.15 Local push notification badge on the application icon	31

Figure 6.16 Today’s drugs screen 33

Figure 6.17 Prescription list screen 34

Figure 6.18 Settings screen 35

Figure 6.19 Healthcare-professional main menu screen 36

Figure 6.20 Create a prescription screen (Healthcare-professional) 37

Figure 6.21 Patient list screen 38

Figure 6.22 Patient detail screen 38

Figure 6.23 Achievement earned screen 40

List of Abbreviations

WHO	World Health Organization
TUIK	Turkish Statistical Institute
PC	Personal Computer
GPS	Global Positioning System
OS	Operating System
iOS	iPhone Operating System
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
RFID	Radio-Frequency Identification
UX	User Experience
UI	User Interface
SQL	Structured Query Language
PHP	PHP: Hypertext Preprocessor
SDK	Software Development Kit
OSX	Macintosh Operating System X
UDID	Unique Device Identifier
API	Application Program Interface
ISV	Independent Software Vendor
JSON	JavaScript Object Notation
SSL	Secure Sockets Layer
MD5	Message Digest 5 Algorithm
APNS	Apple Push Notification Service
phCode	Patient – Healthcare Professional Code

Chapter 1

Introduction

In the last decade, smartphones have started to occupy a significant place in our daily lives. Their huge ecosystem brings us various mobile applications that make the smartphones unique and user-centric. Mobile healthcare applications take an important place among this ecosystem. There are lots of mobile healthcare applications with a motivation to make a patient's life easier.

One of the most important problems that can be seen during the treatment process is poor medication compliance. Patients generally resist to take their medicines as directed by their doctors [1].

Especially for the patients with chronic diseases, medication compliance is very important. Following the prescription indications such as sequence, dosage and time can make the difference between a fast recovery of a patient and a long, costly and even possible hospitalization or death [2].

Even though most of the patients are aware of the importance of the medicine compliance, surprisingly, number of patients who has poor medicine compliance is extremely high. According to the World Health Organization (WHO) Report Adherence to long-term therapy for chronic illnesses in developed countries averages 50%. In developing countries, the rates are even lower. It is undeniable that many patients experience difficulty in following treatment recommendations [3].

The main motivation of the application "Project Medicine" is to provide users to have an application that assists them to have their treatment on a regular basis. Applications in the marketplaces generally provide limited functionality. It is easy to find various applications which provide prospectus information about drugs. Additionally, it also easy to find applications which serves as a drug reminder. However, there is a lack of applications which

serves as both a drug database and a drug reminder system. All of the drug reminder applications in the market works like in this principle: the user creates some reminders and get reminded when the medicine time comes. In this thesis, it is aimed to maximize patient and healthcare professional interaction. An environment is created where the healthcare professionals can assign prescriptions to the users and monitor their medicine compliance. Additionally, patients can easily reach the healthcare professionals through the application to share any details related to their prescriptions.

The common features of some of the mobile applications which are available at the Apple Application Marketplace (AppStore) are shown in Table 1.1. By using these applications a user can set a reminder for a drug, and when the reminder is triggered the user is warned by a sound or a notification. Additionally, a user can use some of those applications to see prospectus information of the drug. The table below shows the common features of these apps.

Mobile Application	Features					
	Prospectus Information	Drug Reminder	Store Prescription	Online Infrastructure	Record Medicine Usage	Online Patient - Healthcare Professional Interaction
My Pills Box [5]	No	Yes	No	No	Yes	No
Medscape [6]	Yes	No	No	Yes	No	No
RxMediaPharma [7]	Yes	No	No	Yes	No	No
Pill Reminder [8]	No	Yes	No	No	Yes	No
Foster [9]	No	Yes	No	No	Yes	No
MedCoach [4]	Yes	Yes	No	Yes	Yes	No
Med Reminder [10]	No	Yes	No	No	Yes	No
Pill Monitor [11]	No	Yes	No	No	Yes	No
Pill Alert [12]	No	Yes	No	Partially	Yes	No

Table 1.1: Comparison of existing applications

However, none of these applications provide an environment to leverage patient-healthcare professional communication. It is possible to see some applications that let a user to record their medicine usage activities or provide an online infrastructure that enables a user to reach his/her details from any device. On the other hand, there is only one application that has both functions but it does not support any kind of interaction with the Healthcare Professional [4].

“Project Medicine” makes it possible for a patient to find prospectus information of a drug, besides, it provides an infrastructure where both patients and healthcare professionals can create/assign prescriptions online. “Project Medicine” records the medicine usage behaviors of

the users so that healthcare professionals can monitor and communicate with the patients if necessary.

By using this application it is aimed to encourage people to take their medicines. Since the users can be monitored by the healthcare professionals, a healthier communication environment is created between the patient and the healthcare professional.

The outline for the rest of my thesis is as follows: The second chapter gives an introduction for mobile applications. A brief summary is given about smartphones and mobile applications. Explanation of different types of mobile applications follows in the third chapter. The fourth chapter gives information about a generic mobile application lifecycle. The fifth chapter focuses on the application development lifecycle customized for the “Project Medicine” mobile application. In the sixth chapter, design and implementation process is covered. All the functionalities and the screens of the “Project Medicine” are explained in detail. Finally, the seventh chapter concludes the thesis and makes recommendations for further research.

Chapter 2

Introduction to Mobile Applications

In Today's world, smartphones have an important place in our lives. In the most of the developed countries, the majority of the population are using smartphones.

According to data from Ericsson, around 55% of all mobile phones sold in the second quarter of 2013 were smartphones, compared to 50% in the first quarter and only 30% in 2012 as a whole. By the end of 2013, there were just under 1.5 billion smartphones in use, of which almost half were in Asia Pacific [13].

As of January 2014, 58% of American adults have a smartphone and 67% of cell owners find themselves checking their phone for messages, alerts, or calls even when they don't notice their phone ringing or vibrating [14].

People use smartphones and tablets for several reasons, including taking photos, listening to music and browsing the internet etc. According to the recent research of Turkish Statistical Institute (TUIK), in the first quarter of 2014, 58% of people were used their mobile phones and smartphones to connect to the internet outside their home and office [15]. Additionally, Gartner, which is a famous information technology research and advisory company, predicts that by 2018, more than 50 percent of users will use a tablet or smartphone first for all online activities [16].

One of the most important components and usage reasons of smartphones are the mobile applications which gives extended functionality to the smartphones.

A mobile application is a software application designed and developed for a mobile device such as a smartphone and a tablet. Unlike software applications we are used to see on our PCs, mobile applications are designed to use the capabilities of the mobile device such as

accelerometer, camera and GPS sensor. By using those functionalities, mobile applications can provide an extensive amount of use-cases for the end users.

Mobile applications have lots of positive effects in our lives. We can use mobile phones to play games, take photos, use social media, and use planning programs or even study. Basically, if we use them correctly, mobile applications can make our lives easier.

A recent study conducted by Flurry Analytics showed that apps currently command 86% of the time that the average US mobile consumer spends on their smartphone. The stats show that mobile applications are dominating users' time spent on devices. [17].

A smartphone has its own operating system that is used to manage the smartphone's software and hardware. Apple's mobile operating system iOS and Android are the most popular ones among several others. Indeed, there are also other mobile operating systems such as Blackberry OS and Symbian OS. Since the application covered in this thesis is implemented for the Apple devices, iOS which is the operating system of Apple mobile devices is described.

iOS is the operating system that runs on iPad, iPhone, and iPod touch devices. The operating system manages the device hardware and provides the technologies required to implement native apps. The operating system also ships with various system apps, such as Phone, Mail, and Safari that provide standard system services to the user [18].

It is possible to implement a mobile application to iOS platform by using several ways such as Native, Hybrid and Web applications.

Chapter 3

Types of Mobile Applications

In this chapter, mobile application types are explained. Since it is important to select the correct approach before starting the implementation, positive-negative sides and the technical capabilities of different types of mobile applications are deeply covered in this chapter.

There are several types of mobile applications. Native Mobile Applications, Mobile Web Applications and Hybrid Mobile Applications [30]. It is not fair to say one of them is the best hence, all of them has some advantages and disadvantages. By following the considerations mentioned below “Project Medicine” is implemented as a native application.

3.1 Mobile Web Applications

Since all the platforms have their own development ecosystem, when you try to deploy a mobile application for all of the platforms, it is necessary to implement the applications specifically for each platform. However, if you build a Mobile Web Application, you do not need to do that. Since a mobile web application is basically a web application which is optimized for the mobile devices, they can be opened by only using the web browser of the mobile device. Therefore, they are platform independent and they are no different from traditional web applications thus using the technologies such as HTML, CSS and JavaScript are enough to build a mobile web application.

There are some advantages and disadvantages of Mobile Web Applications. Mobile Web Applications are cross-platform so it is not needed to develop the application for each and every platform. However, it is still needed to consider different screen sizes, different browser versions etc. It is relatively cheaper to maintain the application, since it has cross-platform support, making the change only in one codebase should be enough. On the other hand, mobile devices' browsers are different browsers than we are using in our PCs. Therefore, it is possible to have some performance issues when using a Mobile Web Application. In addition, it is very

limited to use mobile devices' physical components such as accelerometer, GPS sensor or RFID.

3.2 Mobile Native Applications

Native Applications are built specifically for the device. For example, if you have a native iOS application, this application can only run in Apple mobile devices. Native Applications have the opportunity to use all the allowed device specific hardware and software functionalities of the device. Since it is optimized to use a specific device, it is expected to have maximum performance, better user experience, ability to reach allowed hardware such as GPS, accelerometer and camera. On the other hand, it has some disadvantages such as requirement of implementing the application for different platforms. If the application is intended to have cross-platform support, you have to implement native application for the different platforms from scratch. However, its advantages are really important that's why it is the most preferred way.

3.3 Hybrid Applications

Hybrid Applications are some kind of combination of Mobile Web Applications and Mobile Native Applications. In Hybrid Applications there is a native container and in this container there is a Mobile Web Application. This native container acts like a bridge between the mobile device and the Web application so unlike the normal Web Applications, Hybrid Applications can use the device functionality which is not available to use via browser. Of course, it is needed to implement the container part natively for each platform but it is not needed to implement the web part because it is reusable. However, it is hard to maintain a Hybrid Application and its user experience and smoothness is not as good as Native Applications.

By considering this issues it is decided to choose implementing a Native Application. The reason is that it is aimed to maximize the user-experience and user friendliness. Native Application ecosystem gives the most flexibility to do this.

Chapter 4

Mobile Application Development Lifecycle

In this chapter, a widely used mobile application development lifecycle and its components are explained. It is also pointed out the mobile application specific considerations while describing the mobile application development lifecycle.

There are several software development methodologies and software development lifecycles. As a common point, these software development methodologies and software development lifecycles try to create different phases and require to perform specific actions in these phases. However, they differ in defining the phases, the actions in them and the strategy. Although the mobile application development is not very different and a common application development lifecycle can be applicable, some modifications are needed due to the nature of a mobile application.

In this part it will be mentioned about a widely accepted mobile application development lifecycle. Not only it will be mentioned about the process steps such as Inspiration, Design, Development, Stabilization, Deployment and Maintenance, but it will also be mentioned about the necessary considerations when building a mobile application. Since mobile applications are slightly different from web applications or desktop applications, there will be some extra considerations while implementing a mobile application.

We can divide Mobile Application Development Lifecycle into 5 pieces. Inception, Design, Development, Stabilization and Deployment [28].

4.1 Inception

It is basically the ideation part of the process where the ideas and concepts are formed. Infrastructure of the application is constructed in this stage. Idea of the app is analyzed and defined in this phase. Some technical and usability studies should also take place in this section.

In this section all the functionalities and requirements of the application are defined. An iterative process is constructed with the requirement provider.

4.2 Design

Design is the one of the most important phases of the mobile application development lifecycle. Even though you implement the application whose functional abilities are perfect and working as promised, if it provides a bad user experience, end-user may easily give up using the mobile application. Basically, a good design study makes the mobile application complete.

When designing the User Experience, it is important to consider design guidelines of the platforms. For example, Apple provides a Human-Interface Guideline for iOS. By considering the suggestions on this guideline, it is provided a familiar user interface to the end-users. For example, if you put a back button to an Android application, it is strange for the user because an Android user tends to navigate back by using the physical back button of the mobile device.

After finalizing the UX design, it is proceeded to design the User Interface. This step is the realization step of what is studied in the UX design. In the UI Design phase, colors, graphics, basically all the appearance is designed.

4.3 Development

After Inception and Design phases, development phase takes place. In this phase, the planned and designed ideas turn into implementation. Application code is written in this stage and it is aimed to create an application that satisfies all the functions and requirements as defined. Application code should be clean, understandable and have a good quality.

4.4 Testing and Stabilization

After having some progress in the development phase, Quality Assurance starts to work to identify the broken functionalities of the application. Bugs are needed to be fixed in this phase. If necessary and possible, having a beta release to have a real user feedback can be a good option.

In this phase, it is aimed to create an application which is bug free, no problems in terms of usability and without any performance issue. It is best to consider this step at the beginning of the development process. Because it may be hard to stabilize the application at the end of the development step. In this stabilization phase it can be mentioned about the below four steps:

- **Prototype:** The application has the core functionality but there are lots of bugs.
- **Alpha:** Core functionality is fully implemented but not tested completely. It is normal to have major bugs in this stage.
- **Beta:** The application has most of the functionality. Some of the bugs are already fixed in this stage but user feedback is highly important to be aware of the bugs and the broken functionality.
- **Release Candidate:** The application is completed and fully tested. It is suitable to release the application.

It is nice to start testing as early as possible. Since it is harder to change or modify a functionality, UI Element or the architecture when they are completely finished. Having the feedback as early as possible and taking the action respectively can create a huge advantage.

4.5 Deployment

Deployment is the final phase of the Mobile Application Development Lifecycle. In this phase, the application is deployed to the production environment. Deployment process changes platform to platform. However, the idea behind is relatively similar. There are specific application marketplaces for the platforms and the developer can use these marketplaces to distribute their applications.

For example, there is AppStore for iOS applications, Windows Store for Windows Phone Apps and Google Play Store for Android Applications.

Chapter 5

Project Medicine Mobile Application Lifecycle

Project Medicine has been implemented for the iOS platform as a mobile application for the related devices. It is aimed to create an online platform which lets users to search and find prescription details of the drugs that can be found in Turkey and promises a prescription reminder and scheduler system with patient-healthcare professional interaction.

As mentioned in chapter 4, there is no huge differences between developing a desktop application or a mobile application in terms of development lifecycle. Of course, there are some different considerations needed to be taken care of differently but the main concepts are very similar.

In this chapter, the mobile application development lifecycle of the “Project Medicine” is explained. According to the standard development lifecycle process which is mentioned in chapter 4, following structure has been performed.

5.1 Inception

The idea is creating a mobile application for iOS platform which leverages patient-healthcare professional interaction to provide a better medicine compliance. It is also aimed that user should see detailed information about the medicine through the application without bounding to the small prospectus paper that comes with the drug box.

Technical Aspects and Infrastructure Components;

- Prospectus information of the drugs will be collected.
- The application will have client-server architecture.
- Objective - C will be the programming language on the client side.

- SQLite will be used as a local database. The data that needed to be stored in the local device will be stored in the mobile device's SQLite database.
- PHP (Hypertext Preprocessor) will be used as a server side scripting language. All the web services and the server side logic will be handled with PHP.
- MySQL will be used as a server side database. All the data that needed to be stored in the server side will be stored in MySQL database.

Analyzing and defining the application requirements;

Registration process: The user gives his/her personal details to sign up. A user can sign up with two different roles; as a "Normal User" or as a "Healthcare Professional". Normal user is the user who can basically use the application without having any administrative capability such as assigning drugs to other users or monitoring them. On the other hand, a healthcare professional has all the functions a normal user has, additionally, he has administrative capabilities.

Login process: The user inputs his username and password to login the system. If he does not remember his username or password he can use his email address for password recovery.

Search for a drug: The user can enter the name of the drug so that he can see all the information about the drug (prospectus information). It should not be necessary to login to the system to use this functionality. User can see detailed drug information quickly without logon.

Recording the medicine usage: The user sees the upcoming treatment information he/she is going to have at that specific date. In addition, it is needed to record the medicine usage behavior of the user, user should be able to inform the system when he takes the medicine.

Creating prescriptions: A user should be able to create prescriptions. User should be able to name a prescription and add drugs. He can add more than one drug to the prescription and when adding those drugs he should be able to see prospectus information to be sure that he is adding the correct drug to the prescription. After adding a drug, it should be possible to edit it.

There should also be an option to create and assign prescriptions to other users. This should be something only the healthcare professional capable of. Healthcare professional should also be

able give priority to the prescriptions. For example, if there is a critical item in a prescription, the healthcare professional may want to act immediately if the patient does not comply that specific treatment.

Seeing Existing prescriptions: Users should be able to see their prescription details. They should be able to modify them if necessary. Additionally, since the drug names may be confusing for the users, when a user want to see the details of a prescription, he/she should be able to see a picture that helps the user to recognize the drug.

See Patients: Healthcare professionals should be able to see a list of their patients. They should be able to see personal details, prescriptions and medication compliance information of their patients.

Settings: Users should be able to change their personal details such as email, password and phone number. Besides, users should be able to manage their "healthcare professional-patient code" here that will help to create a secure communication between the patient and the healthcare professional.

5.2 Design

By considering the requirements, flowcharts for the application is created. After the creation process of the flowcharts, navigation of the application is studied. Suggestions from the Human-Interface Guideline for iOS has been considered to maximize the end user experience. In addition, it is also aimed to have a real user feedback in this phase. All the flowcharts and the navigation flow are shared with the real users to collect some feedback and make necessary changes before starting to implement the navigation infrastructure and UI artifacts.

5.3 Development

After finalizing the design phase, the development phase has started. All the logic, implementation and functional tests took place in this phase. With the help of GIT distributed source code version control system, which is designed to handle everything from small to very large projects with speed and efficiency [19], the source code maintainability is successfully established.

5.4 Testing and Stabilization

This phase started to take place at the very beginning of the development phase. After the end of the each core functionality, the application is tested by the users to have feedback. By considering the feedbacks from the users, necessary changes have been performed and again the application is tested by the users. This process is performed iteratively until having the user acceptance.

At the end of each iteration, it is aimed to have a bug free, highly usable and efficient versions. At last, a release candidate version is created.

5.5 Deployment

Since the Project Medicine is a mobile application for the iOS platform, Apple Application Marketplace (AppStore) should be used to deploy and distribute the application.

5.6 Developing Mobile Applications for Apple Devices

Besides the main development lifecycle mentioned in the previous section an extra development lifecycle specific for the Apple Mobile Application development took place in parallel. Every mobile platform has its own internal development tools and processes. They have their own Software Development Kits, Integrated Development Environments and toolboxes. “Project Medicine” is implemented for Apple devices and some parts of its development process differ from the other platforms. In this chapter, Apple platform specific development process and its components are explained in detail.

To develop an iOS application the developer must have a Mac computer which contains OS X operating system. After that, the only thing needed is to download and install XCode application from the AppStore. XCode is the Integrated Development Environment of Apple and it comes with the iOS SDK which is a must have to develop an iOS mobile application. By having those, the developer simply has a source editor, interface editor, tools and frameworks for iOS development.

There are several languages that can be used to develop an iOS application such as Objective-C and Swift. In the thesis, the application is implemented by using Objective-C. Objective-C is an object oriented programming language which is built on top of the C programming language. The developer can use Xcode to implement an iOS mobile application with Objective-C.

To put the application on the AppStore marketplace, the application must be digitally signed. This sign means that the application is created by a registered developer or a company. To sign the application, developer must have an Apple iOS Developer account.

To build and distribute an application for iOS several components are needed: A developer, UDID, AppID, Application Source File, Signing Certificate and Provisioning Profile [29].

- Developer: The one who creates the application.
- UDID: Unique Identifier of the Apple Device. It is needed because you have to register the device ID so that you can deploy your application to the device.
- AppID: Unique Identifier of the application. In the Apple application marketplace (AppStore) every application have a unique identifier.
- Application Source File: These are the components of the application such as codes and images that make the application.
- Signing Certificate: When a developer enrolls to the Apple Developer Program, he creates a Developer Certificate. It is needed to sign the application. There are two types of Developer Certificates; Development Certificate and Distribution Certificate. Development Certificate is for development and testing phase. On the other hand, distribution certificate is needed when the developer puts the application to the AppStore.
- Provisioning Profile: Provisioning Profiles are one of the most important elements of the development process. It enables the device to be used for testing. Provisioning Profile is the element which makes the application usable on the iOS device. To run the application code on a device, at first a development Provisioning Profile must be installed on the device. This Provisioning Profile simply contains the Development Certificate, UDIDs (Provisioned Devices), AppID, Creation Date, Version and some other information.

There are different types of Provisioning Profiles such as Development, Ad-Hoc and Distribution [29]. Development type is used when the developer needs to run and test the application on a specific device. Ad-Hoc type is used to make the application like it is downloaded from the AppStore. Basically, it enables to test what will happen if the application is downloaded from the AppStore. Last but not least, Distribution type is used to sign the application to be sent to the AppStore.

To sum up, there are basically three steps of development process which are needed to take place to develop a mobile application for iOS platform: Development, Ad-Hoc and Distribution.

5.6.1 Development

In the Development Phase, the application is implemented and tested. The developer can use iOS simulators which are provided by Xcode to load to application to the virtual devices so that the developer can test the functionality and the design of the application. However, although simulators are useful to see and test the application, it is a necessity to load the application to a real device such as an iPhone or an iPad. To load the application to a real device not only source files are enough but also Developer Certificate and Provisioning Profiles are needed. Basically, the application must be signed by the developer. For the first step, using the Application Source Files, Developer's Signing Certificate and the Development Provisioning Profile are enough to build and test the application on a real device.

5.6.2 Ad-Hoc

After the Development Phase, Ad-Hoc phase takes place. In the Ad-Hoc phase Ad-Hoc Provisioning Profile is used and it is expected that the application would be like if it is downloaded from the AppStore. In the Ad-Hoc Phase the Application Source Files, Company/Organization's Signing Certificate and the Ad-Hoc Provisioning Profile are used to see what the application would be like if downloaded from the AppStore.

5.6.3 Distribution

If everything is okay for the Ad-Hoc Phase, Distribution Phase takes place. The application again built in this phase by using related certificates and profiles and shipped to the AppStore. If the application passes the review of the Apple, it becomes available at the AppStore to the all Apple Device users. In the Distribution phase the Application Source Files, Company Provisioning Profile and the Distribution Certificate is used.

Chapter 6

Project Medicine Application Design and Implementation

In this chapter, design and implementation process of the “Project Medicine” is explained. First, database management systems used in the application are explained. Next, the server side implementation process and components are described. Then, the client side design and implementation process is explained.

As mentioned in chapter 2 there are several platforms to implement a mobile application for handheld devices. “Project Medicine” is implemented for Apple’s iOS platform. Apple uses iOS Operating system for its mobile devices so the application should be specifically implemented for the popular and stable iOS Operating System.

6.1 Database Design

One of the key elements of “Project Medicine” application is its database. Since its functionality highly depends on the medicine information, prescription records and patient-healthcare professional communication, it is necessary to build an effective and easily maintainable database.

In the application, there are lots of data that needed to be stored such as prospectuses, user information and medicine information etc. There are two different database management systems used in this application. One of them is the local database of the phone. One of the options to manage the data in iOS is to use SQLite database management system. SQLite can be used for low-level database work. iOS basically contains the SQLite library which is a lightweight relational database engine. It uses a procedural, SQL-focused API to manipulate the data tables directly. Since using the local database promises an always available, fast, secure and network connection independent database, some of the functionalities, which does not contain user specific information and requires relatively better performance, is handled by using the local database of the mobile device.

It is aimed to implement Project Medicine device independent. It means, even though you do not have your phone, it should be possible to use your credentials with any phone and see your details. To make this happen, all the user specific data is stored in a remote server. So if the user has the internet connection, he/she can use any device that has iOS to see his details. However, for some cases, making requests to a remote server is costly and does not offer good performance. For example, in the application there is a functionality to list all the drugs (10K pieces). It is also possible to filter and quickly find the required drug information. If those drugs would be in a remote server, each and every time the application would make a request to download all the drugs from the server. Instead, this drug list was put into the local SQLite database for the sake of simplicity. Of course, when the user presses the drug item to see the details, a connection is opened between the application and the remote server to get the prescription details of the application.

On the other hand, the replication of the drug list is also stored in the remote database. At each launch of the application, version of the remote drug list is checked, if there is an update the local database is synchronized with the remote database. In conclusion, although some of the things are stored in the local database, it is periodically synchronized with the remote server to have the most updated version.

The other database management system which is used at the remote server side is MySQL. MySQL is one of the most popular open source databases in the world. Whether you are a fast growing web property, technology ISV or large enterprise, MySQL can cost-effectively help you deliver high performance, scalable database applications [20]. Since it is an open source, robust, highly suits for small and medium applications and most importantly has a very large and collaborative community, it is decided to use MySQL at the remote server side. Figure 6.1 presents the database diagram for the application. As mentioned before all the application and user data are stored in the remote database and the data is reached through web services by the client.

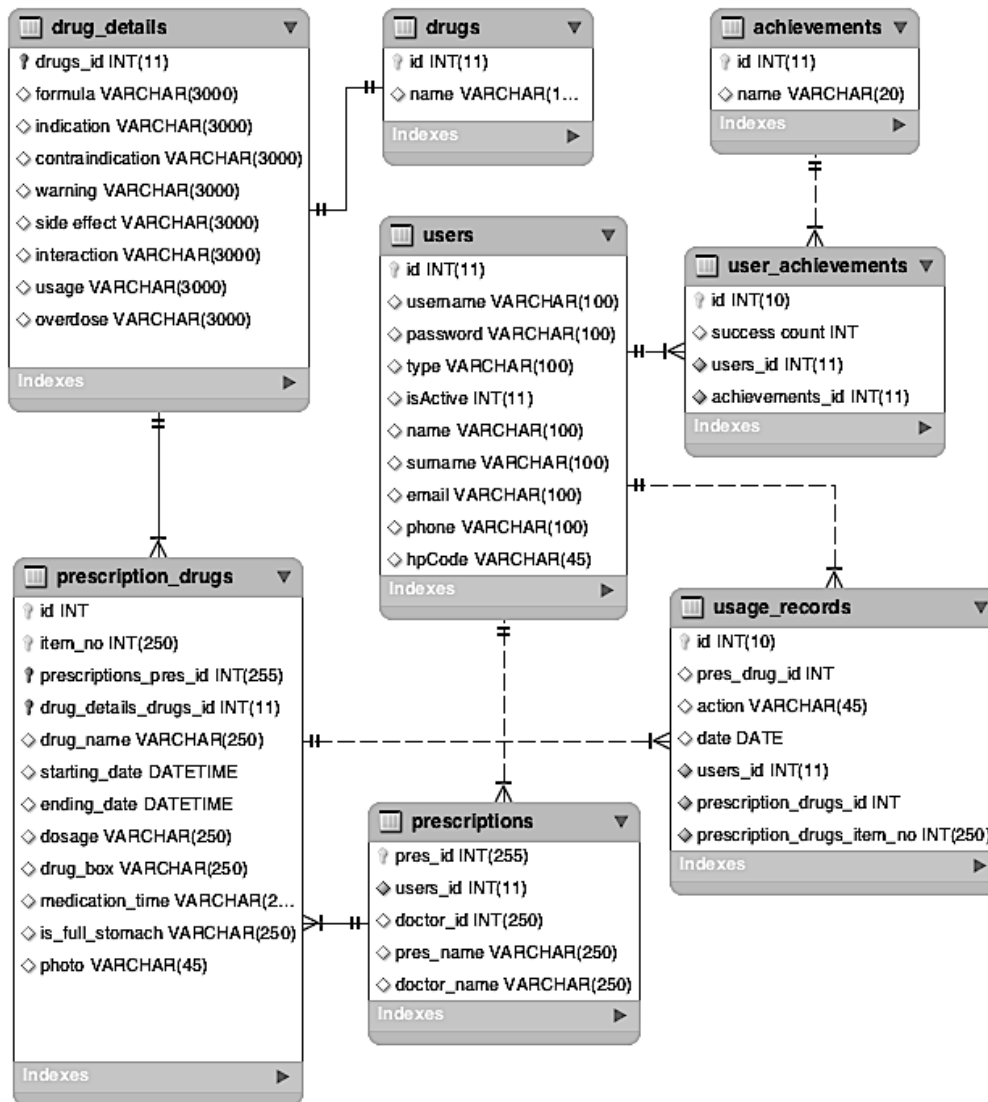


Figure 6.1: Database diagram

6.2 Using PHP Web Service to do Remote Database operations

Since almost all the user specific data is stored at the remote MySQL database, it should be possible to read and manipulate the data from the “Project Medicine” application. However, although it is technically possible to directly connect to the MySQL database from the iPhone application, it is not the best practice. Instead, PHP scripts that contain web services are used as an abstraction layer. The system architecture is illustrated in Figure 6.2. “Project Medicine” application basically sends the request to the PHP web service. Then, the PHP script connects to the database so that it can query the database for the requested data and returns the result to the iPhone application in JSON (JavaScript Object Notation) format. The data which is retrieved in JSON format can be easily parsed and converted to native Objective-C objects,

then used in the client side. On the other hand, it is also possible to do “insert”, “update” and “delete” operations. For example, when a user wants to create a prescription, prescription details are sent by using the related web service method. The PHP script at the server side uses the data which is sent with the request and executes the necessary SQL (Structured Query Language) statements to manipulate the database tables.

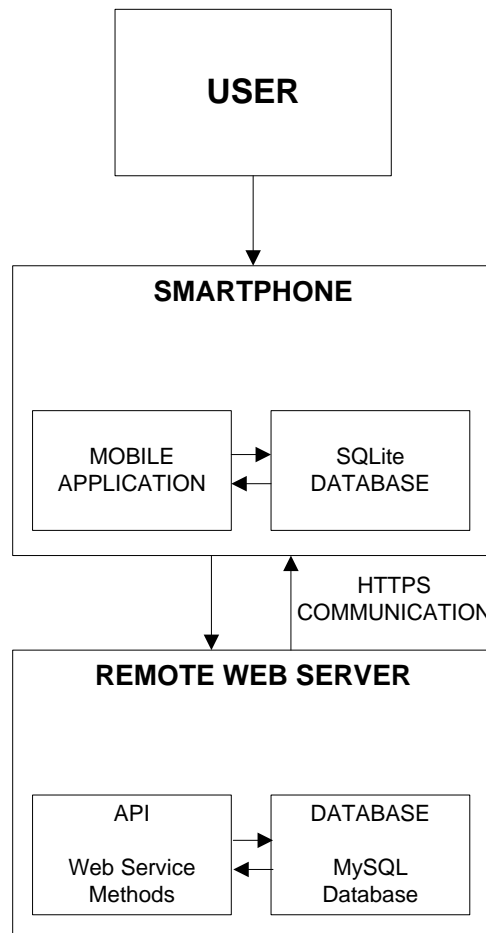


Figure 6.2: System architecture

In conclusion, in the server side there are PHP scripts that contain several web service methods. The request is sent to a PHP script from the iPhone application and the PHP web services are used to access the database to be able to execute the necessary operations. Then, the JSON response received from the web service is consumed at the client side.

The communication between the client and the server is secured with Secure Socket Layer (SSL). SSL makes possible to establish an encrypted link between the server and the client. It ensures that all the information passed between those points will remain secure. It is also used frequently for the systems, which has client server architecture. In Project Medicine, it is also

needed to create connection between the server and the client and it is needed to pass private information all the time. Therefore, secure connection between the Project Medicine mobile application and the server was established by maintaining the SSL.

6.2.1 A Brief explanation of the working mechanism of SSL

At first, SSL is activated on the web server. After the activation of SSL on the web server, it is created two cryptographic keys: a Public Key and a Private Key. SSL encryption works with the usage of public and private key pairs. The point is if the data is encrypted by the key, only its pair key can decrypt it. Typically, the public key encrypts the data and the private key decrypts it. Basically when the client sends a message, it encrypts the message with the public key before sending it to the server. After the server receives the message, it uses the private to decrypt it. Public key can be shared with any client, and the message is encrypted with it so even an attacker sniffs the network and takes the message, it is really hard to decrypt it without having the private key. Therefore, only the server should know the private key.

6.3 Screens and Functionalities of Project Medicine

In this chapter, detailed information about the “Project Medicine” functionalities is given. By considering the different types of users, use-cases, screen designs and the functionalities of the Project Medicine are described in detail. First, different user types and their capabilities are explained. Then, the technical and the functional aspects at the each and every screen of the application are given in detail.

6.3.1 User Perspectives

Two different types of users can use the application. One type is the normal user such as patients, parents and relatives. The other type is the healthcare professional such as surgeons, nurses and dentists. The major difference between these types is that a healthcare professional can assign prescriptions to normal users or other healthcare professionals and monitor their medicine compliance. On the other hand, a normal user should not be able to assign prescriptions to any other user or monitor their medicine usage.

6.3.2 Landing Page of the Project Medicine

The “Project Medicine” mobile application is designed and implemented for the iOS platform. All the functionalities mentioned below are implemented and tested successfully for various Apple mobile devices including iPhone 4, iPhone 5s, iPhone 6 and iPad 4. The login system flowchart is shown in Figure 6.5

6.3.2.1 Registration

To use all the functionalities of the application, users should be registered. Both types of users can register from the same page. Users should provide details such as name, surname, e-mail, phone number, username and password. Only the Username, email and password fields are mandatory as the username and password are needed to login to the system. E-mail is also a mandatory field for the password recovery. Besides, username is the unique identifier of the user so it should not have been taken by another user. However, it is also important to fill other fields because they are the elements which will definitely leverage the communication between the users. The welcome screen and the registration screen are shown in Figure 6.3 and Figure 6.4 respectively.



Figure 6.3: Welcome screen

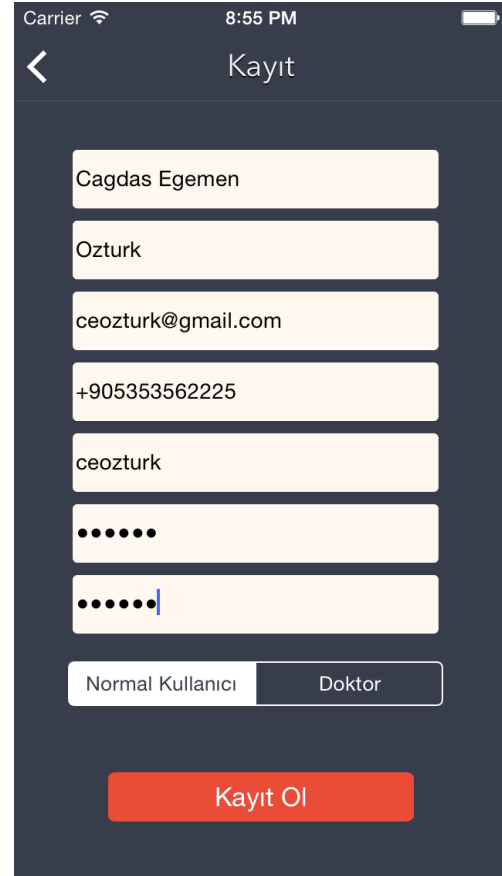


Figure 6.4: Registration screen

For the sake of personal information privacy of the user, it is decided to hash user's password with a hash algorithm. Hash algorithms turn any amount of data into a fixed-length "fingerprint" that cannot be reversed [21]. After researching some existing hash algorithms, it is decided to use MD5 (Message-Digest algorithm 5) algorithm. MD5 is a well-known cryptographic hash function with a 128-bit resulting hash value. MD5 is widely used in security-related applications, and is also frequently used to check the integrity of files [22].

The idea is to protect user's privacy. By hashing the password, the server does not have the real password but only the hashed value of it. MD5 promises a one way encryption, which basically means it is not possible to get back what the hashed message was.

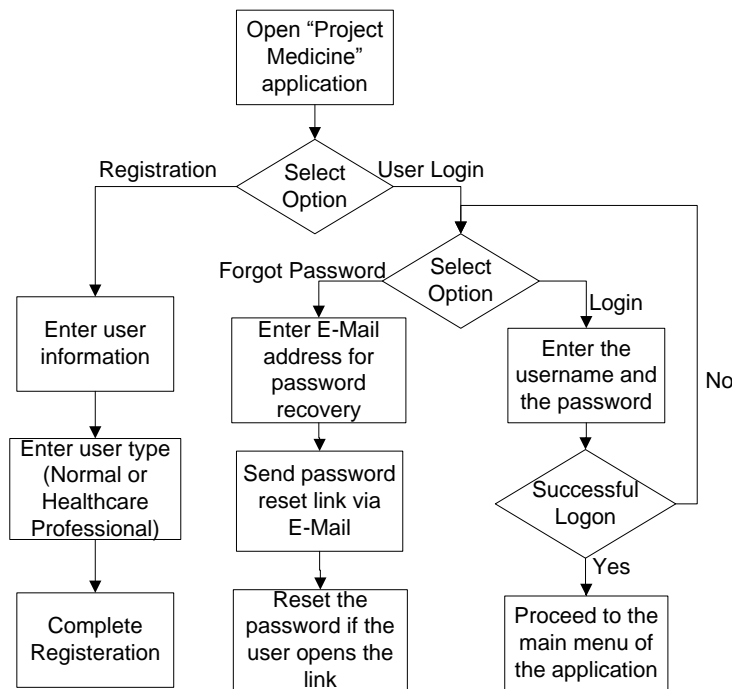


Figure 6.5: Login system flowchart

After having a feedback from a security expert, it is decided to use salted password hashing that is considered as a more secure way to hash a message. The point is when two users uses the same password their hashed results will be the same. Which results if an attacker maintains a lookup table that contains pre-computed hashes of the different passwords, he may extract the message. To create the uniqueness hashes can be randomized with a “salt” by appending a string to the message [21]. In project medicine a random string is used as the salt. Basically, the salt is appended to the password and then hashed and stored in the database. Since the attacker will not be aware of the salt, he will not be able to pre-compute a look-up table.

6.3.2.2 User Login

From the main screen user can press the user login button and enter the username and password to login to the system. If the user does not remember his password, he can use “password recovery” button and enter his E-mail so that he can recover his account. Since the server does not have the real password but a hashed value, it is impossible to send the password to the user. Instead, a link to reset the password is sent to the user via E-Mail so that the user can create a new the password and login to the system. The login screen and the forgot password screen are shown in Figure 6.6 and Figure 6.7 respectively.



Figure 6.6: Login screen



Figure 6.7: Forgot password screen

6.3.2.3 Find prospectus information of a drug

For the sake of simplicity, you do not need to login to the system to find some information about a drug. Therefore, it is possible to launch the application and press to the “Drugs” button to navigate to the drug list easily. On the other hand, it is also possible to search drugs after logon. The drug list screen and the drug detail screen are shown in Figure 6.8 and Figure 6.9 respectively.



Figure 6.8: Drug list screen

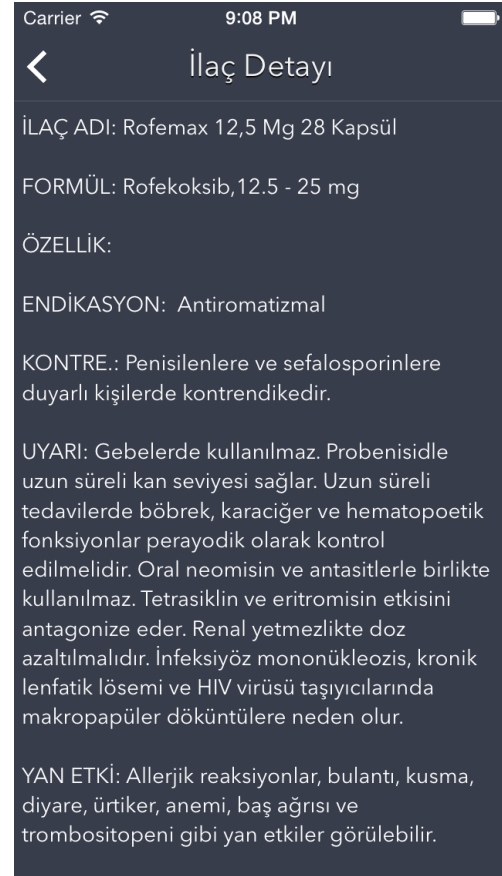


Figure 6.9: Drug detail screen

In this section, user can see a search bar and a list of drugs [31]. The list of drugs come from the local SQLite database for the sake of performance improvement. User can use the search bar to find a specific drug. Since the drugs are stored in the local database, the performance of the search result is very satisfactory. When the user finds the drug he can press on it to see further details such as formula, indications, contraindications, warnings, side effects, usage and so on. User can see the all those information about the drug hence, this information comes from the remote server. By using the web services, drug id is sent to the server and as a result, information about the drug has been gathered. Detailed information about the drugs is not stored in the local database because in a case like that the application would require too much space on the device.

6.3.3 “Project Medicine” Main Menu

After logging in as a normal user, the application welcomes the user with a normal user main menu screen. This screen has five components: “Find a Drug”, “Today’s Drugs”, “My Prescriptions”, “Create a New Prescription” and “Settings”. The patient main menu screen and the system flowchart are shown in Figure 6.10 and Figure 6.13 respectively.



Figure 6.10: Patient main menu screen

6.3.3.1 Find prospectus information of a drug

It is completely identical with the functionality as described before. User can basically see a search bar and a list of drugs to explore specific drugs.

6.3.3.2 Create a New Prescription

Creating a New Prescription is one of the key functionalities of the application. By using this functionality user can create prescriptions. Each prescription should contain at least one drug and the user gives the usage details for the drug(s) stored in the prescription.

After joining this screen, user can see a text field which represents the prescription name, add drugs button which leads the user to another screen to select a drug and finally “save” button which is used to send the prescriptions to the remote server. The create a prescription screen (Patient) and the prescription item detail screen are shown in Figure 6.11 and Figure 6.12 respectively.

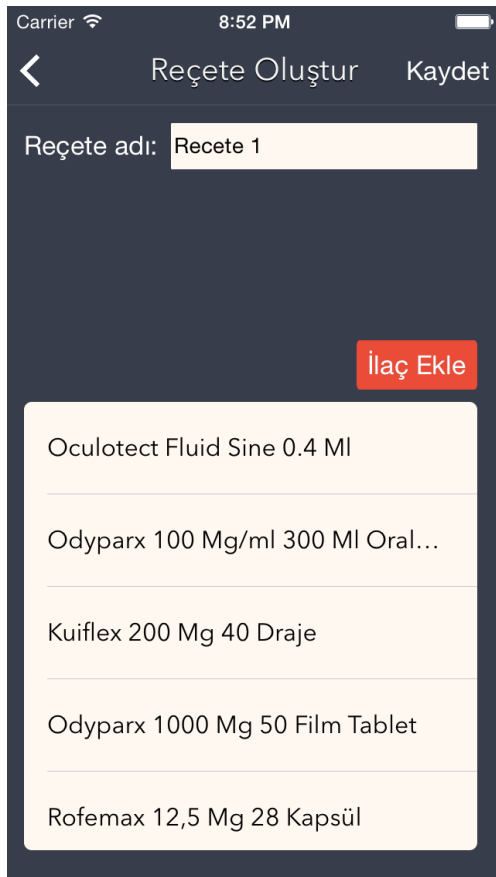


Figure 6.11: Create a prescription screen (Patient)



Figure 6.12: Prescription item detail screen

When the user presses the “Add Drug” button, the same drug list mentioned in the drug search functionality is appeared with only one difference; an add button next to each item. When the user finds and selects a drug, he presses to the add button and goes back to the “Create a New

Prescription” screen with a pop-up. In this popup, selected drug name is automatically comes and user only needs to fill the other fields such as “Full Stomach?”, “Quantity in the box(optional)”, “Dosage”, “Start Time”, “End Time (Optional)” and “Medication Interval”. The user can also take the pictures of the pills or the drug boxes and add these pictures to the prescription so that he/she can easily recognize them later.

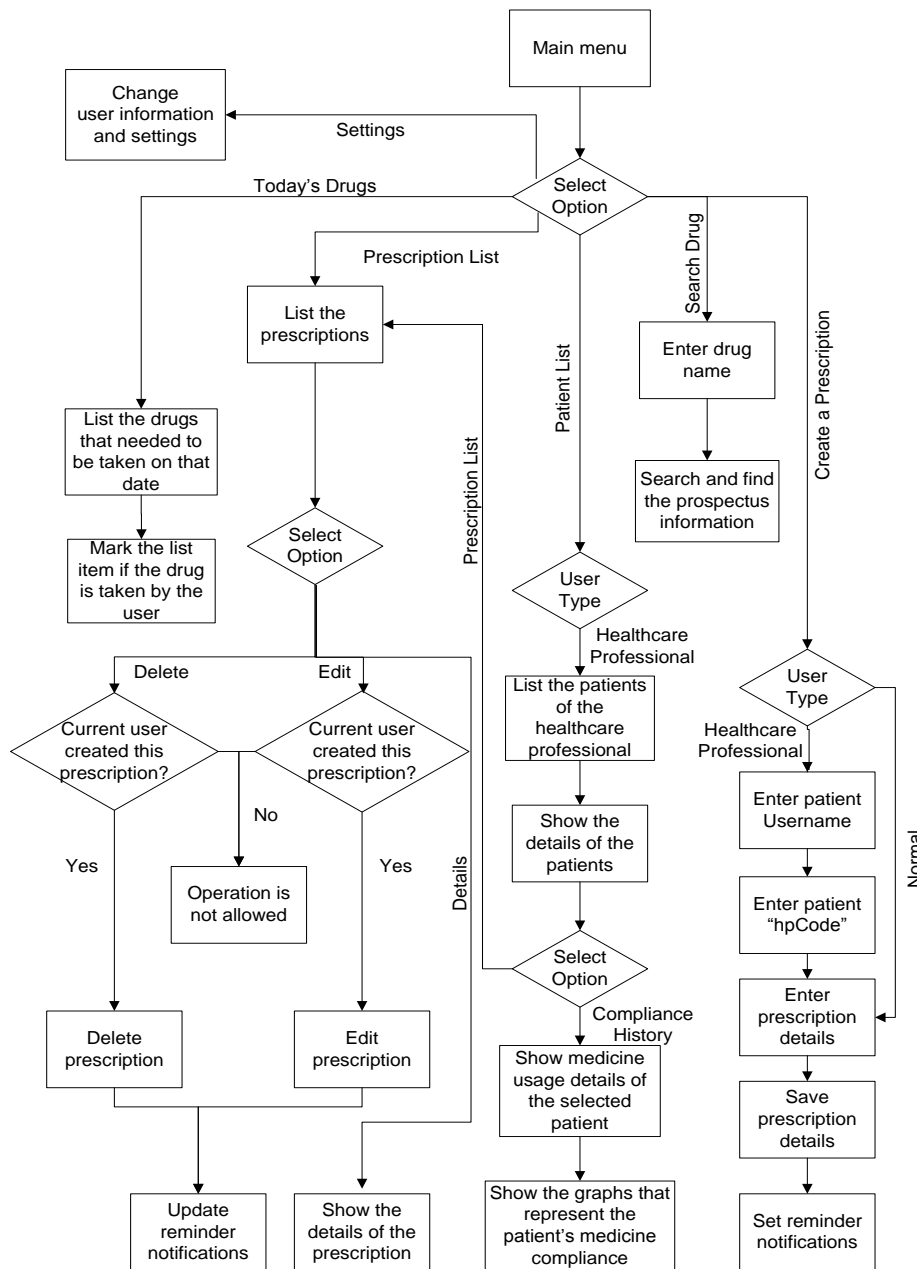


Figure 6.13: System flowchart

After adding one or more drugs user can see a drug list in the Prescription Creation View and he can modify it before submitting. If everything is okay user can press the “Save” button to send the prescription configuration to the server. Now the prescription is saved for this specific user. Additionally, local notifications are also set to remind medication times to the user. The local push notification alert on lock screen and the local push notification badge on the application icon are illustrated in Figure 6.14 and Figure 6.15 respectively.

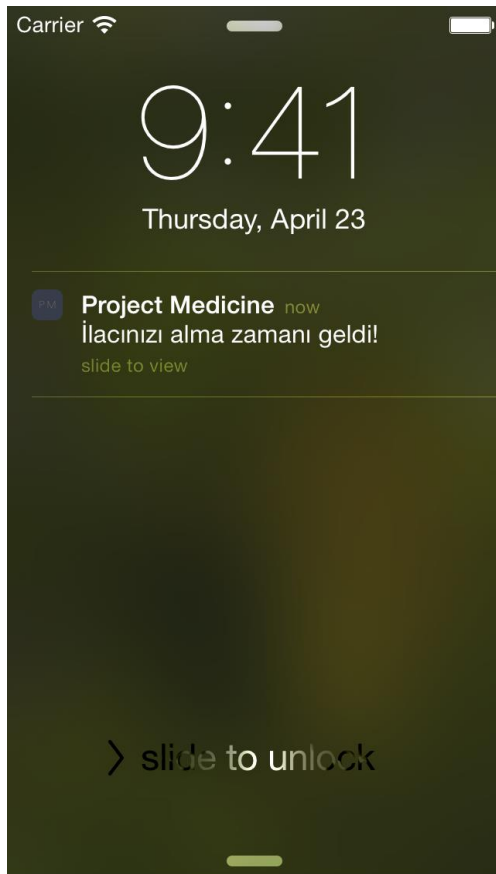


Figure 6.14: Local push notification alert on lock screen

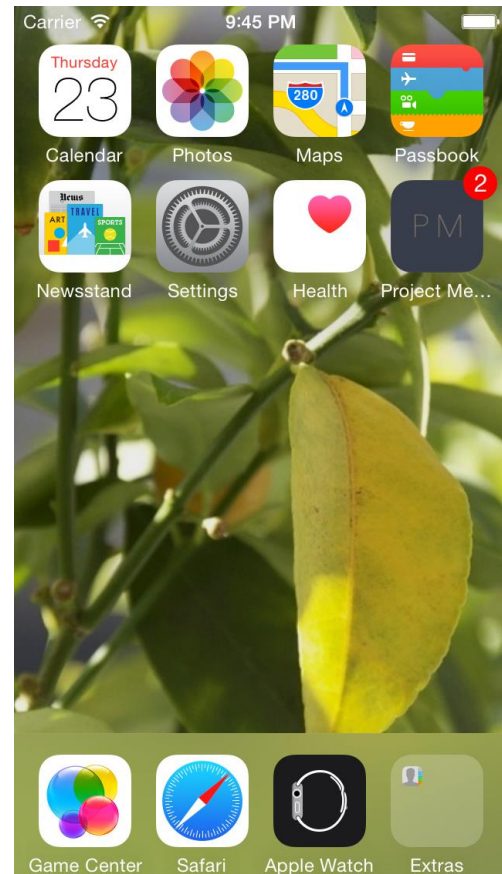


Figure 6.15: Local push notification badge on the application icon

For the sake of reliability, local notifications are used in the application. The other option was the Remote Notifications (Push Notifications). Local and remote notifications are used to notify the users by enabling the application that is not running in the foreground to let its users know the application has information for the user [23]. Although, their functionality serves similarly, their working principle is totally different. Push Notifications uses Apple Push Notification Service (APNS). APNS is responsible to transport and route a remote notification from a given provider to a given device. It means that if there is no internet connection it is not possible to receive remote push notification. Although internet connection is needed to use

majority of the functionalities of the application, it is not acceptable not receiving the medication time reminder when the user does not have an internet connection. Additionally, push notifications are working with fire and forget mechanism. It means that there is no guarantee that push notifications will actually be delivered, even if the APNS server accepted them [23] which makes the push notifications unreliable.

As a result of this, local notifications are used in the application. Local notifications are basically scheduled and sent by the app itself so they are reliable. Other advantage is even the user is not logged in or does not have an internet connection, since the local notifications are already set during the prescription creation process, user gets the medication reminder notification.

6.3.3.3 Today's Drugs

Today's Drugs functionality has two important roles. One of them is to show users what drugs will be taken on that day, the other one is to let users to record their medicine taking activity.

In Today's Drugs screen the user simply sees a list of drug names and time. It serves as an informative screen so that the user can see what kind of medicines will be taken on that day. On the other hand, there is a checkbox next to each item which represents the usage status of the item. The user must select this checkbox after using the related medicine so that his medicine usage behavior can be tracked. The only constraint is that the user must select the checkbox during the day because after midnight, Today's Drugs list will be refreshed for the next day. The today's drugs screen is shown in Figure 6.16.

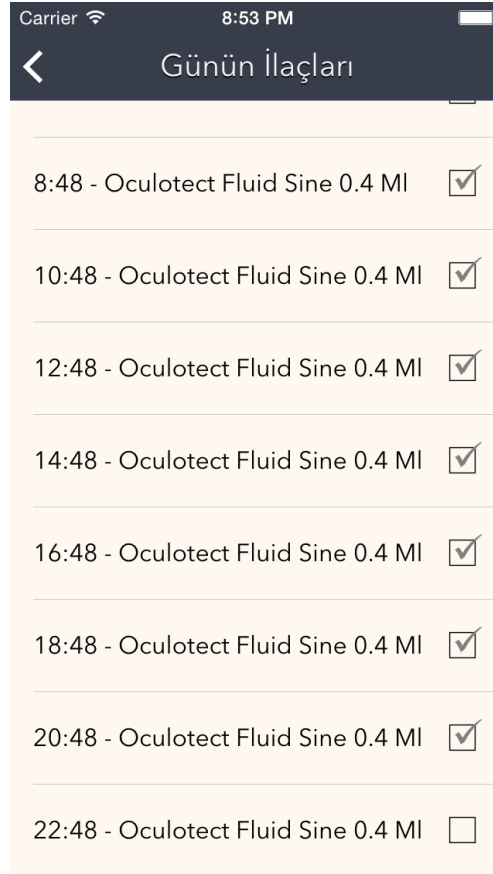


Figure 6.16: Today's drugs screen

At first, it is considered that the user should select the check box in ten minutes after the medication time to measure medicine compliance more accurately. However, it is realized that even though the user take the medicine he may not be appropriate in some situations to open the application and select the item. For example, user can be in a business meeting or can be out of GSM service. In a situation like that, even though the user used the medicine, he may not be available in ten minutes to select the checkbox. For this reason, it is thought that it should be also reasonable to let user to select the checkbox before the day ends.

It is one of the most important parts of the application. If the user does not select the checkbox after using the medicine, it is impossible to have an insight about the user's medicine compliance.

6.3.3.4 My Prescriptions

It is important to see currently active prescription items in one place. My Prescriptions button navigates the user to the page where the user can see all the active, both user created and healthcare professional created, prescription names in a list. The prescription list screen is shown in Figure 6.17. It is also possible to see the details of the prescription. Since it may be needed to see the details or to change the configuration of the prescription.

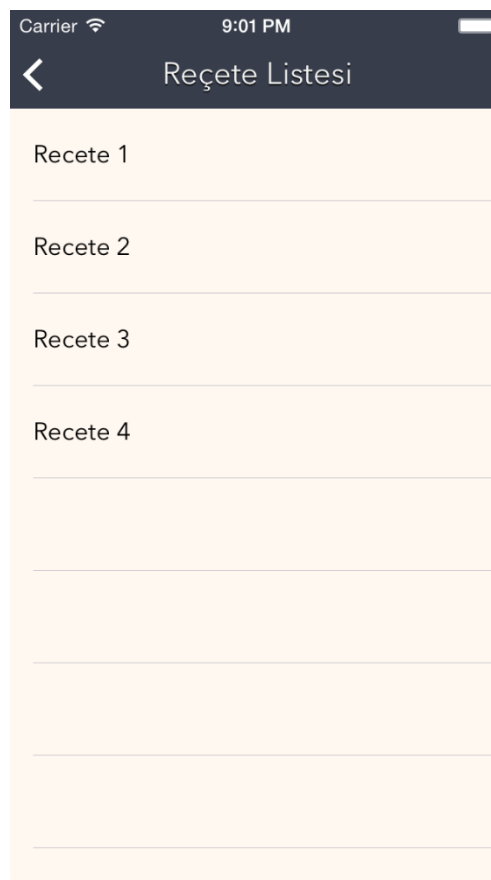


Figure 6.17: Prescription list screen

When the user presses a prescription item to see the details he is faced with a screen very similar to the new prescription creation screen. The user can add new drugs to the prescription or edit the existing ones. However, these modifications are limited. A user can only change the prescriptions which are created by him. If a prescription is assigned by a healthcare professional, the user cannot change the configuration of the prescription.

6.3.3.5 Settings

It is needed that the user should be able to change some of his details such as email, phone number and password. The user can also turn on/off the notification service from the Settings menu. The settings screen is shown in Figure 6.18. Last but not least, user can see and change the patient-healthcare professional code which will be referred as phCode.

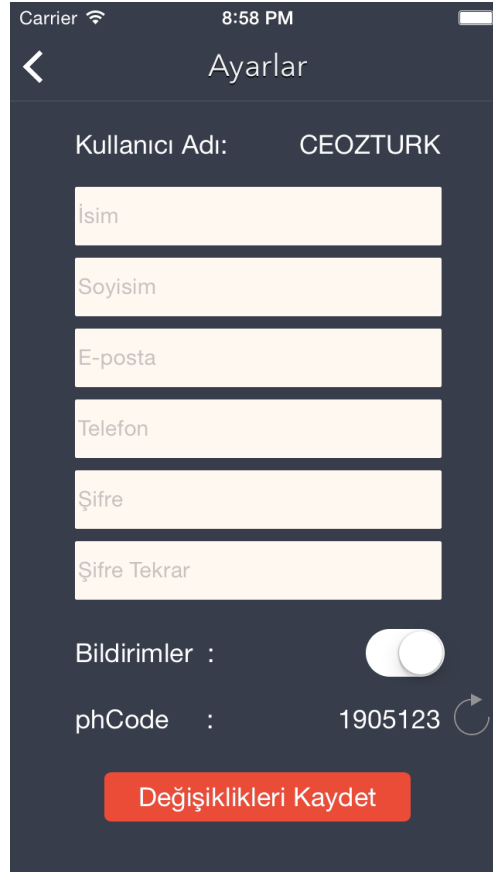


Figure 6.18: Settings screen

phCode gives the user extra uniqueness. At first, it was enough to assign a prescription to a patient by just using the patient's username. Basically, the user tells his username to the healthcare professional and when the healthcare professional wants to assign a drug to the user he enters the username of the patient and assigns the prescription. However, after having feedback from healthcare professionals, it is realized that it is likely to face some problems here. For example, if somebody knows the username of the patient, he can easily create a healthcare professional account and spam the user with unwanted prescriptions. Therefore, phCode functionality is added. When the healthcare professional wants to assign a prescription to a patient not only he must enter the username of the user but also he must enter the phCode

of the user. The critical thing is it can be also exposed and spammed the same way it was mentioned. However, there is a functionality to change the phCode in the settings menu. User can easily update his phCode by pressing the refresh button. Now the phCode is changed and if a healthcare professional wants to assign a prescription to the patient, he should request the new phCode from the patient.

6.3.4 Differences between the Normal User and the Healthcare Professional

The main differences between the normal user and the healthcare professional are the ability to assign prescriptions to other users and being able to monitor their medicine usage records and compliance. Healthcare Professional can also use the application as a normal user. He can create his own prescriptions and get notified about his medicine times. The healthcare professional main menu screen is illustrated in Figure 6.19. In this section, different screens and functionalities that the healthcare professional have will be explained.



Figure 6.19: Healthcare-professional main menu screen

6.3.4.1 Create a New Prescription

Create a new prescription functionality is almost identical with the normal user. Except, there are two extra input fields for the username and the phCode. The create a prescription screen (Healthcare-professional) is shown in Figure 6.20. If the healthcare professional fills those fields properly the created prescription automatically created for the input user. However, if the healthcare professional does not fill those inputs, the prescription will be automatically assigned to himself.

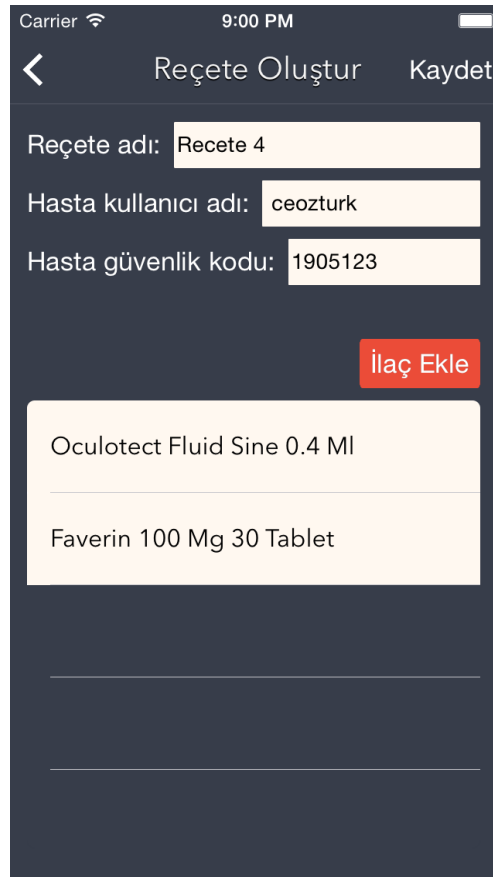


Figure 6.20: Create a prescription screen (Healthcare-professional)

In addition, the healthcare professional can give priority to the prescriptions from one to ten. If there is a critical item in a prescription the healthcare professional can give a higher priority to that prescription. As a result of this, when a patient does not comply that specific treatment, the healthcare professional gets a notification which also indicates the priority level of the prescription.

6.3.4.2 My Patients

My Patients button leads the user to the screen which is only available for the healthcare professionals. In the first screen the healthcare professional sees a list of his patients. When the healthcare professional assigns a prescription to a patient, the patient automatically marked as the patient of the related healthcare professional.

Healthcare professional can find and select a specific patient to see further details. The patient list screen and the patient detail screen are shown in Figure 6.21 and Figure 6.22 respectively. When he/she selects a patient, he/she is navigated to a screen where personal information of the patient can be seen such as name, surname, phone number and email. Besides, the healthcare professional is faced with two other buttons named, “Prescriptions” and “Medicine Compliance Monitor”.

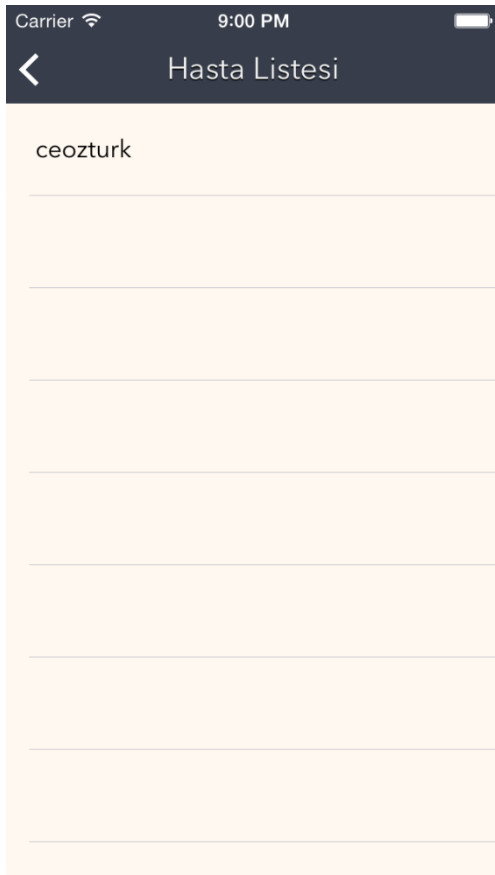


Figure 6.21 Patient list screen



Figure 6.22 Patient detail screen

If the healthcare professional presses to the Prescriptions button, he/she can see the prescription list of the patient. The important thing is he/she not only see the prescriptions assigned by him/her but also see the prescriptions assigned by the patient himself or another healthcare professional. To see what other medicine is used by the patient may give an insight to the healthcare professional. He/she may use this information to decide what treatment should be more effective.

The healthcare professional can press the prescription items to see the details of the prescriptions. However, he/she can only modify the prescriptions that was assigned by himself/herself.

Other button which is mentioned before is the Medicine Compliance Monitor button. Based on the daily medicine taking information, informative charts are created in this section. The healthcare professional can see the medicine compliance behavior of the patient by using this functionality.

6.3.5 Gamification

Gamification is the concept of applying game-design thinking to nongame applications to make them more fun and engaging [24].

The research company Gartner predicts that in 2015, a gamified service for consumer goods marketing and customer retention will become as important as Facebook, eBay, or Amazon, and more than 70% of Global 2000 organizations will have at least one gamified application [25].

Gamification concept uses several components that is seen in the games such as award mechanism and competitive play. It integrates these components to make nongame applications more interactive and desirable.

“Project Medicine” is not a game application and it aims to encourage people to take their medicines on time. It is thought that using gamification concept can motivate users and help them to embrace the application and use it regularly and effectively. There are several game mechanics to integrate gamification concept to an application. The five most commonly used

mechanics in gamification are: Points, Achievements (Badges), Levels, Leaderboards, and Challenges [26]. Achievement mechanic is integrated into “Project Medicine”.

6.3.5.1 Game Mechanic: Achievements

Achievements are a virtual or physical representation of having accomplished something. Achievements are a way to give players a way to brag about what they've done indirectly as well as add challenge and character to a game [27]. It is very common to use badges in this game mechanic. Since the badges represent the success of an achievement, when a user successfully completes a predefined task (achievement), they “unlock” the related achievement and get a badge. The achievement earned screen is shown in Figure 6.23.

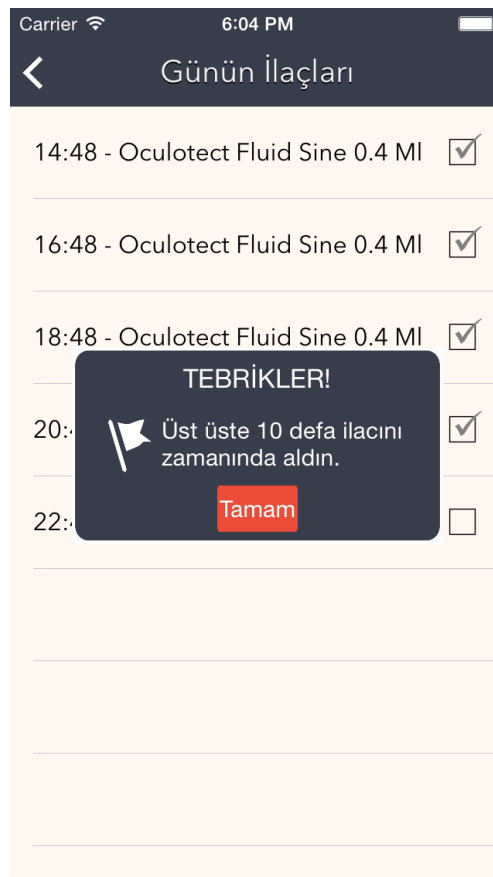


Figure 6.23 Achievement earned screen

Some of the achievements defined in the “Project Medicine” are shown below. They are defined as locked and hidden. When a user fulfills the requirements of an achievement he/she earns the related badge:

- Three medicines in a row on time
- Ten medicines in a row on time
- Twenty medicines in a row on time
- Successfully created a Prescription
- First touch with a Healthcare Professional
- First touch with a Patient
- Finished a treatment with an excellent adherence

Chapter 7

Conclusion and Recommendations for Future Work

In Today's world, smartphones have a massive place in our lives. Apart from using them for communication, people use its applications to play games, access news and e-mails, read books, find a location etc. Smartphones serve as a handheld computer and its capabilities make people's life much easier.

Taking medicines on time and as directed is really important for an efficient treatment. Especially for the chronic illnesses, medicine compliance plays a key role in the treatment process. As mentioned in the introduction, number of people who has poor medicine compliance is extremely high.

In this thesis, it is proposed a mobile healthcare application named "Project Medicine". "Project Medicine" is a mobile prescription reminder and scheduling system. Additionally, it provides an online infrastructure where the patients and healthcare professionals can interact with each other. "Project Medicine" lets users to find prospectus information of drugs, create and manage prescriptions and get notifications for the medication schedules. More importantly, healthcare professionals can also assign prescriptions to the users, monitor their medicine compliance and communicate with them. By creating such environment, "Project Medicine" encourages people to take their medicines on time and have closer communication with the healthcare professionals.

In future works, when creating prescriptions, the application can check the possible interactions with the existing drugs and notify the patients and healthcare professionals. Additionally, there are currently several hardware solutions that collects various data from the user including blood pressure, sleep pattern, heart rate and body temperature. This data can be imported to the Project Medicine mobile application so that the healthcare-professional can monitor the patient in much more detail.

References

- [1] B. Kocurek, “Promoting Medication Adherence in Older Adults and the Rest of Us”, *Diabetes Spectrum*, vol. 22, no. 2, pp. 80–84, 2009.
- [2] J. M. Silva, A. Mouttham, and A. E. Saddik, “UbiMeds: A Mobile Application to Improve Accessibility and Support Medication Adherence”, *Proceedings of the 1st ACM SIGMM international workshop on Media studies and implementations that help improving access to disabled users - MSIADU '09*, 2009.
- [3] E. Sabate and W. H. O. Staff, *Adherence to long-term therapies: evidence for action*. Geneva: World Health Organization, 2003.
- [4] *MedCoach Medication Reminder* [Online]. Available: <https://itunes.apple.com/us/app/medcoach-medication-reminder/id443065594?mt=8>
- [5] *My Pills Box* [Online]. Available: <https://itunes.apple.com/us/app/my-pills-box/id964008217?mt=8>
- [6] *Medscape* [Online]. Available: <https://itunes.apple.com/tr/app/medscape/id321367289?mt=8>
- [7] *RxMediaPharma* [Online]. Available: <https://itunes.apple.com/tr/app/rxmediapharma/id568941038?l=tr>
- [8] *Pill Reminder by Drugs.com* [Online]. Available: <https://itunes.apple.com/us/app/pill-reminder-by-drugs.com/id453359236?mt=8>
- [9] *Foster - Pill Reminder* [Online]. Available: <https://itunes.apple.com/us/app/foster-pill-reminder/id554168976?mt=8>
- [10] *Med Reminder* [Online]. Available: <https://itunes.apple.com/us/app/med-reminder/id421104320?mt=8>

- [11] *Pill Monitor – Medication Reminders and Logs* [Online]. Available: <https://itunes.apple.com/us/app/pill-monitor-medication-reminders/id477666512?mt=8>
- [12] *Pill Alert – Medication, Prescription Reminder and Tracker* [Online]. Available: <https://itunes.apple.com/us/app/pill-alert-medication-prescription/id889581369?mt=8>
- [13] GSMA, “The Mobile Economy 2014”, 2014.
- [14] *Mobile Technology Fact Sheet* [Online]. Available: <http://www.pewinternet.org/fact-sheets/mobile-technology-fact-sheet>
- [15] Türkiye İstatistik Kurumu. *Hanehalkı Bilişim Teknolojileri Kullanım Araştırması, 2014* [Online]. Available: <http://www.tuik.gov.tr/PreHaberBultenleri.do?id=16198>
- [16] Gartner. *Gartner Says By 2018, More Than 50 Percent of Users Will Use a Tablet or Smartphone First for All Online Activities* [Online]. Available: <http://www.gartner.com/newsroom/id/2939217>
- [17] A. Heaton. (2014, July 8). *The latest statistics on mobile app usage to inform your mobile marketing strategy* [Online]. Available: <http://www.smartinsights.com/mobile-marketing/app-marketing/mobile-app-statistics>
- [18] *About the iOS Technologies* (2014, Sep. 17) [Online]. Available: <https://developer.apple.com/library/ios/documentation/Miscellaneous/Conceptual/iPhoneOSTechOverview/Introduction/Introduction.html>
- [19] *Git* [Online]. Available: <http://git-scm.com>
- [20] *MySQL: The world’s most popular open source database* [Online]. Available: <http://www.mysql.com>
- [21] Defuse Security. *Secure Salted Password Hashing - How to do it Properly* [Online]. Available: <https://crackstation.net/hashing-security.htm>

- [22] *What is MD5 (Message Digest number 5)?* [Online]. Available: <http://www.accuhash.com/what-is-md5.html>
- [23] *Local and Remote Notifications in Depth* (2015, Mar. 09) [Online]. Available: <https://developer.apple.com/library/mac/documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/Chapters/WhatAreRemoteNotif.html>
- [24] *Gamification* [Online]. Available: <https://www.gamification.org>
- [25] B. Burke. (2014, Apr. 04). *Gartner Redefines Gamification* [Online]. Available: http://blogs.gartner.com/brian_burke/2014/04/04/gartner-redefines-gamification
- [26] Z. Fitz-Walter. (2013, Jan. 24). *A brief history of gamification* [Online]. Available: <http://zefcan.com/2013/01/a-brief-history-of-gamification>
- [27] *Game Mechanic: Achievements* [Online]. Available: https://badgeville.com/wiki/Game_Mechanics/Achievements
- [28] *Introduction to the Mobile Software Development Lifecycle*. [Online]. Available: http://developer.xamarin.com/guides/cross-platform/getting_started/introduction_to_mobile_sdlic
- [29] A. Holguin. (2011, Apr. 17). *Apple iOS Development Process Summary / Explanation* [Online]. Available: <http://swfhead.com/blog/?p=1185>
- [30] *Mobile Web Apps vs. Mobile Native Apps: How to Make the Right Choice* [Online]. Available: http://www.lionbridge.com/files/2012/11/Lionbridge-WP_MobileApps2.pdf
- [31] *Türkiye İlaç ve Tıbbi Cihaz Kurumu* [Online]. Available: <http://www.titck.gov.tr>

Curriculum Vitae

Çağdaş Egemen Öztürk was born in 1988 at Şişli, İstanbul. He received his BS degree in Computer Engineering in 2012 from Kadir Has University. Since 2013 he has been working as a software developer at SAP Development Center Turkey.