

Designing Apps Interoperable and Functional on Multiple Mobile Platforms using Google Environment

D.Inupakutika, C.Basutkar, S.Kaghyan, D.Akopian, P.Chalela, A.G.Ramirez, A.McAlister

Abstract

The use of mobile and web tools in health care has greatly improved interactions between doctors, patients and healthcare professionals in the past few years. According to the University of Texas Health Science Center at San Antonio (UTHSCSA) almost 75% of the 296,980 women in the United States that are diagnosed with breast cancer will have hormone receptor-positive breast cancers. Endocrine hormonal therapy (EHT) is very effective for nearly all women with hormone-receptive positive tumours and is the most widely prescribed one. The dedicated use and adherence to this therapy for 5 years has also shown larger reduction in recurrence [6]. However, even with such proven benefits, the adherence is limited to just 33% of all the women who are prescribed. In such cases, the use of interactive easy-to-use apps would promote and improve adherence [2]. Such apps should enable fast responses to patient queries, guide patients through treatment, help them understand symptoms, motivate them through educational content, and prompt interaction with their peers. In this paper, we describe an approach for accelerating app prototyping using the existing Google Android platform and converting it to a cross-platform web application thereafter. Google Firebase [1] is used as a database server to assist in monitoring and sending notifications to users without compromising the safety and security of patients' data. The proposed system and approach can also be further tailored for similar technology-assisted health promotion and intervention studies. The effectiveness of the approach is evaluated through a randomized controlled study with breast cancer patients conducted by the UTHSCSA research team.

Index Terms – smartphones, web services, cross-platform, mHealth, apps, technology, mobile devices, internet

Introduction

Mobile technology ownership has increased substantially over the past decade. Mobile devices have provided opportunity for innovation in the delivery of health promotion interventions, facilitating access to hard-to-reach groups. Use of mobile technology for clinical purposes (i.e., text messaging, health apps) is rapidly growing [3]-[5]. In general, mobile health (mHealth) [4], [5] nowadays addresses an array of issues: improving the convenience, speed, and accuracy of diagnostic tests; monitoring and managing chronic conditions, medication adherence, smoking cessation, physical activity promotion, appointment keeping and medical test result delivery, improving patient-provider communication, health

information communication, data collection and access to health records. The number of mHealth apps published on the two leading platforms, iOS, and Android, has over 325,000 mHealth apps available since 2017 [22]. mHealth services typically involve backend architectures and/or teams of health professionals that monitor and consult interested individuals [5].

According to The University of Texas Health Science Center San Antonio, about 296,980 women in the United States are diagnosed with breast cancer each year and approximately 75% of them have hormone receptor-positive breast cancers. Endocrine hormonal therapy (EHT) is effective and appropriate for nearly all women who have hormone receptor-positive tumors. Patients who used EHT for about 5 years have shown 50% better results in survival rates and reduction in recurrence. The UTHSCSA research team in collaboration with the Software Communication and Navigation Systems Laboratory at UT San Antonio, is conducting a randomized controlled trial to test the feasibility and effectiveness of a bilingual, culturally tailored, personalized, interactive mobile app in combination with patient navigation to improve adherence to oral endocrine hormone therapy among breast cancer patients. [6].

Considering that breast cancer is most common among women over age of 40, and they are especially affected by usability problems, for instance not being able to perform tasks of setting reminders for a doctor appointment or medication, on switching to a new smartphone it takes time for them to get used to the new user interface. This population is at risk to be excluded from the benefits of using mHealth applications or services [7]. Usability issues are often caused by mobile platform restrictions, for instance application's fragmentation, availability of similar widgets, user interface designs and limitation in screen sizes. These restrictions force developers to build application code for specific and each target platform: Android, iOS etc., separately resulting in increased development cycles and resources. Eventually users have to adapt to varied user interfaces and navigation [8]. Since smartphones are easily accessible by 80 percent of the mobile phone users, the systematic and constructive approach to this problem is to design an android application and then convert it to a web service. Our research envisions to reduce the impact of fragmentation that mobile technologies suffer, which stems from the diversity in devices (ranging from tablet, to smartphone), operating system platforms (Android, iOS, Windows Phone, and Symbian), and diversity in user preferences and needs (i.e., accessibility).

This paper demonstrates a design approach for

multiplatform applications, which use the development tools from Google, and a user-centric method for design validation. Google Firebase has a feature to send notifications to users without having to manually code it in the project. Monitoring personnel, referred to as Patient Navigator (PN) monitors patients subscribed to the program, and uses a Firebase admin account to send calendar events, reminders and notifications to users. The home page of the application includes buttons and icons redirecting the patient to various activities. Facebook is integrated for peer modeling and building a network of support. The application is published on Google Play Store with a single .apk file. As part of this research, the application is also converted to a cross-platform web service for broader deployment.

Proposed System

A. System Architecture: Cross-platform Application Concept

The core component in adherence to EHT is to increase patients' **self-efficacy and motivation**, by providing education via the mobile phone app on general information about hormone receptor positive breast cancer, importance

of EHT adherence and potential side-effects, in addition to information on how to prevent or treat common side effects, self-monitoring/management of symptoms, copying and self-care skills, etc. The acquisition of skills and performance of new actions is facilitated by **peer modeling**, in which participants are able to observe other breast cancer patients like themselves modeling those skills and performing the promoted behavior. The app will provide regular medication reminders, motivational messages, and will facilitate a direct communication with the medical team, as well as help them build a support network. The optimal solution to this was integrating the built-in Google calendar app into the android application, which allowed for educating and motivating patients by presenting similar experiences from fellow patients as well as health care professionals. Since the phone uses Google account based authentication [9], it is easier to synchronize the events and send reminders from the PN. When the user signs in for the first time, she is prompted to choose a Google account from the account picker. This Gmail ID for the chosen account is recorded on the firebase database. PN uses this ID to send relevant information to the patient, such as emails, reminders, and notifications. Google authentication methods are used in the project to make the patient data secure and HIPAA compliant [10].

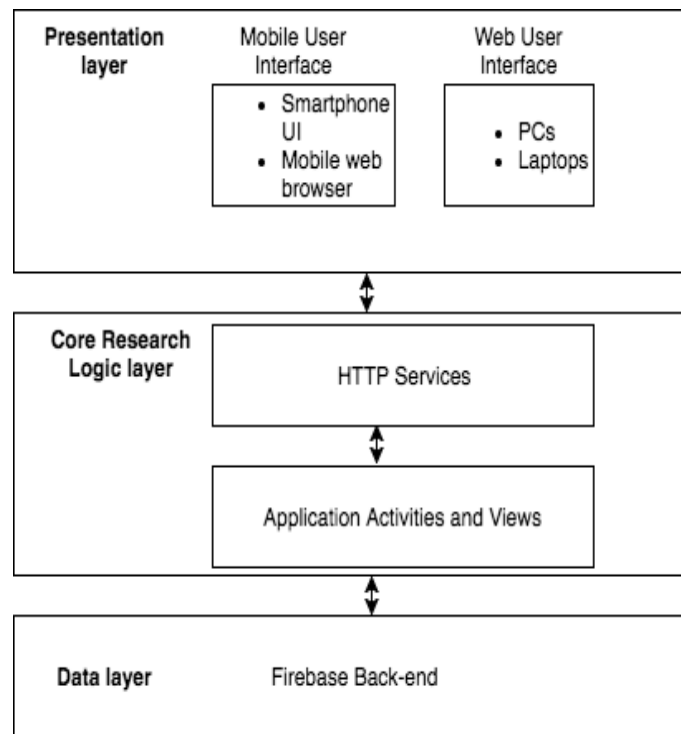


Fig. 1. Design of a cross-platform application

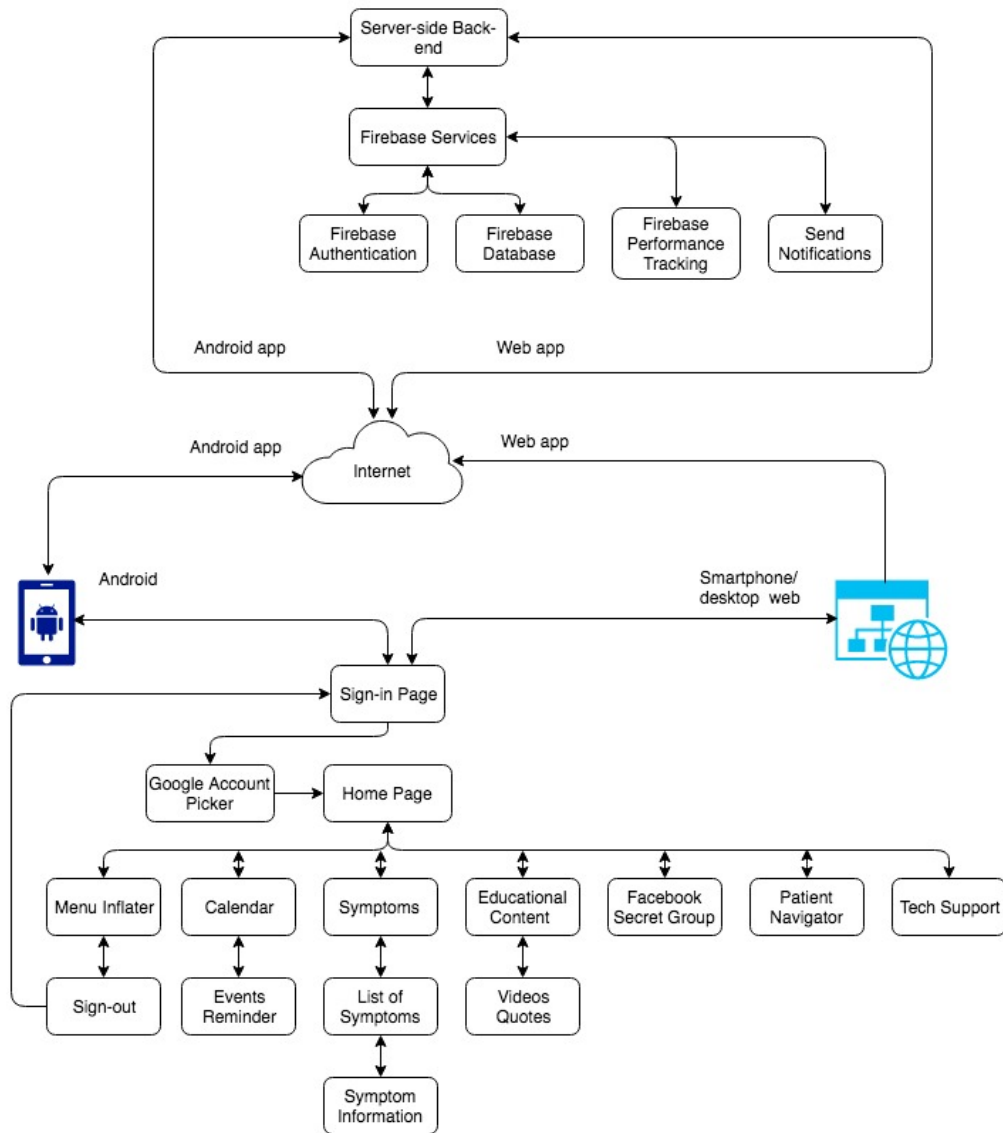


Fig. 2. Detailed Concept with Application Modules and Client-Server Interaction

B. System Features: End-to-End Architecture and Concept Workflow

By making use of Firebase, all user data can be saved on the cloud, making it safe and secure. All components can be contained in a single app and connected through buttons on a landing home page. Fig. 1 shows the cross-platform application concept for rapid design. Fig. 2 shows the application workflow and its interaction with the service [11]. Fig.3 describes the web portal dashboard for PNs, in which they can perform CRUD (create, read, update and delete) operations on patient records. This functionality has proven to facilitate mHealth data collection for further analysis and processing, which are desirable attributes in health intervention research.

System Design

There are two ways to access the application; one is by downloading the app from the Google play store. Another way to install it is by having the user open the .apk file, which can be obtained through a trusted source. An additional web-service has been developed to provide this .apk file. After the user successfully installs the application on the device, the app prompts the user to login using a Google account. If a default account is not present, the app displays a picker, which allows to either select an existing account, login to another account or to create a new Google account. Upon successful login, a home page is displayed containing access to various parts of the application, in the

form of buttons. Every button points the user to a different activity. The application needs to be connected to the Internet to allow the user to login and to submit any record changes, but it can also store temporary data in the application's cache until the device connects to the Internet and synchronizes the data. Authentication, database, performance tracking, and notifications are utilized from Firebase. The user and the app related data are all saved on the firebase database. Fig. 2 represents the complete flow of the project, including the web service.

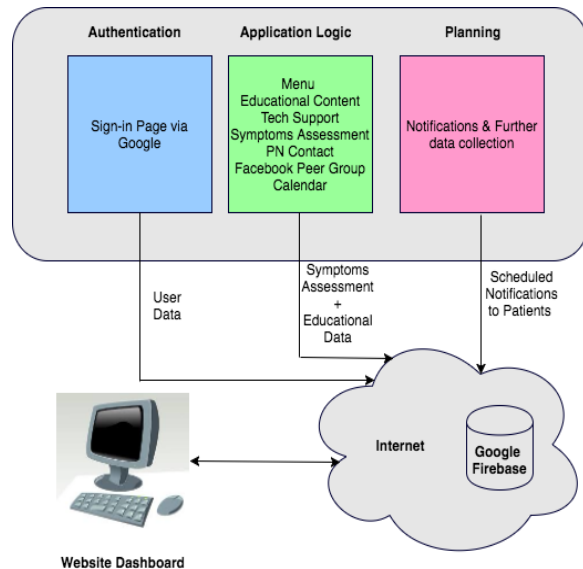


Fig. 3. Dynamic usage of application

Methods

Phase 1: Developing a Native Android Mobile Application

As a first step to our systematic approach of designing interoperable cross-platform app, we developed a native android app. For the app development, we needed to understand how to leverage the capabilities of mobile technology to incorporate traditional intervention elements. Android application is made up of Android activities [12]. Each activity includes one or more features of the application. All the activities interact with the user, so the activity Java class creates a window where we can place our User Interface (UI) defined by an XML file. A set of activities along with services, broadcast receivers and content providers, make an android application. Hence, the core components of the project were developed as different activities. Although the activities work together to form a cohesive user experience, each activity is independent from others. The *Intent* class is used to maneuver to another activity; this also allows few variables to be passed to the other activity.

Application Structure

Android application is developed using an open-source build automation tool called Gradle [13] for adaptable and fast automation. The application consists of various activities corresponding to the application's functionality. It has following features:

- Clinical use and assistance in diagnosis (through symptoms reporting activity that provides the possibility to check up on symptoms and personal health record access through symptom weekly summary, through contact activity for providing easy way to communicate with the PN if necessary)
- Reminder (scheduling of appointments, help patients manage their appointments and events through calendar activity)
- Easy medical referencing (Included reference guides and other breast cancer specialized medical referencing through Educational content and videos activity)
- Maintaining healthy lifestyle (through all the activities)

All the activities are individually described in the subsections below.

Splash and Sign-in Activities

Android apps take some time to start up, usually when the app is opened for the first time. There is a delay that is unavoidable. The splash activity shows a logo or a visually appealing image. The amount of time the user takes looking at the splash screen is the same time the app takes to start. An intent is used to redirect the user to the sign-in page once the tasks in splash activity are completed. Sign-in integrates standard Google Sign-in into the app.

We retrieve the user's unique ID, along with their name and email ID to be stored in Firebase as shown in Fig. 3. Every time a new user signs in, his/her Unique User ID (UID) is used to create a node in Firebase, under which the email ID and name is stored for reference. Although, the system checks if the user already exists, then skips the part where his details must be entered in database. This is required for keeping track of app access and to allow only authorized users to be part of this health intervention.

Home Activity

Home activity is the primary screen where the user navigates to different features of the app through buttons. Users can access the features through the buttons specific to each feature. Each feature is described in the following sections.

Reminder and Events on Calendar App

The application allows users to keep track of their medication, doctor appointments, and peer group interactions. It also allows users to sync in phone calendar app with the reminders from EHT app (shared Google Calendar) for increasing adherence to their treatment. This

is useful when an assigned doctor shares calendar data with a patient.

Educational Content

The application delivers motivational videos, quotes, and other information to help Hormone Therapy patients keep moving forward with their therapy.

Symptoms

This activity consists of a list of breast cancer symptoms for patients to keep track of. Patients can submit their symptom intensity levels and these get recorded in Firebase, along with patient information and timestamps. This data allows for PN and doctor to be aware of the patient's situation and advise patient accordingly. Each symptom has a sliding progress bar with 5 entries starting from 0 to 4, which map to symptom levels "not at all", "slightly", "moderately", "quite a bit", and "extremely."

Patient navigator (PN)

This is an activity class that displays the information related to PN. The user can contact the PN by email, text message, or phone call.

Contact Support

This is an activity where the user can contact the developer or tech support team in case she finds issues with the app. The activity displays the phone number and email address of the team.

Applications' Backend: Google Firebase

Firebase is an Internet service, which is provided by Google. It includes several products such as Analytics, Real-time Database, and Authentication that can easily be integrated into the backend.

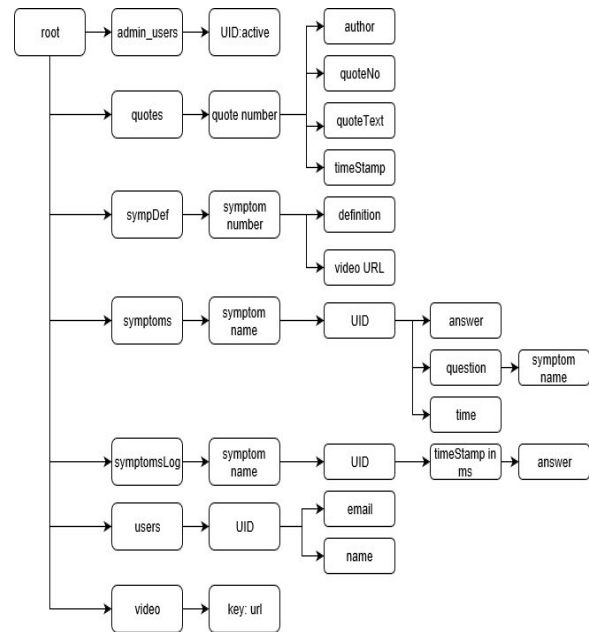


Fig. 4 Firebase Database Structure

Database Structure

The data stored in the Firebase database is in the form of a JSON tree. It can be perceived as a cloud-hosted JSON tree. Unlike SQL, there are no tables with rows. When the data is added to the JSON tree it becomes a node in the existing JSON structure with an associated key. A node can be accessed in source code by using the set of keys that index it in the JSON structure.

Fig. 4 shows the database structure. The top-most node is called the root node. A symptom level submission under a particular symptom name, such as "decreased appetite", for example, is saved under the respective UID and the timestamp at which the most recent submission was done. Fig. 5 shows a symptom level submission, as it looks in the Firebase web console.

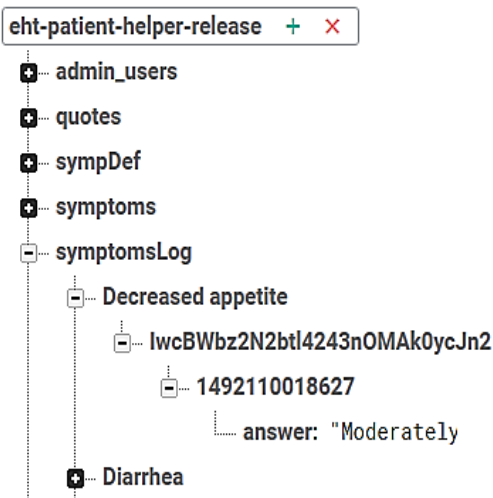


Fig. 5. symptomsLog database snapshot

Facebook secret group for Peer Networking

A closed secret group is made under the Patient Navigator Facebook account. A secret group is where only the participating members added by the patient navigator are allowed access to read and write posts. For the Android application, a Facebook button is placed on the action bar. This button opens the Facebook group on the Facebook app. If a Facebook app does not exist on the user's device, the same group is opened using the device's default web browser. For the web application, the Facebook group is opened in browser.

Publishing on Google Play Store

Publishing is the general process that makes the Android application available to users. When we publish an Android application we perform two main tasks:

- Prepare the application for release: During the preparation step, a release version of the application is built, which users can download and install on their Android-powered devices.
- Release the application to users: During the release step, we publicize, sell, and distribute the release version of the application to users.

Usually, the application is released through an application marketplace, such as Google Play Store. However, an application can also be released by sending its file directly to users or by letting users download it from our website. The publishing process is typically performed after testing of the application in a debug environment is finished. Also, as a best practice, the application should meet all of the release criteria for functionality, performance, and stability before its release. Fig. 6 below shows the EHT Patient Helper Android app on Google Play Store after its release.

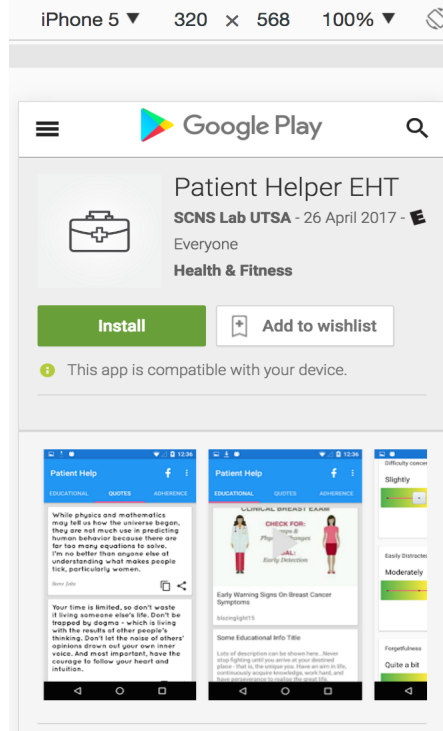


Fig. 6. Android EHT Patient Helper App

Phase 2: Web-based Implementation

Medical information commutes with patients, nurses and doctors and hence, the need for such information to be ubiquitously available is most important to reliable patient care and health-care systems. The EHT mobile application that has been discussed runs on the device once download from the Google Play Store [14]. The application makes use of web views and is hybrid in a way that some features are embedded as web pages. In the healthcare field, the distributed nature of applications finds a standardized technical solution in the web. Hence, we have also developed a web application that runs on a Node.js [15] web server and can be accessed through a web browser like Chrome, Firefox, Safari, or Internet Explorer. All the data is typically saved in Google Firebase (similar to native mobile application). The application is designed and developed for multiple device (mobile devices, laptops and desktops) usage. The purpose of developing our mobile native as well as web application is to provide bedside management to breast cancer patients and Patient Navigator for their daily activities and involvement. With these handheld tools, we can try to suppress the cumbersome steps in their respective processes' of recovery and guidance to keep them as simple and motivating as possible. This implementation also extends the number of platforms, which can run the application, as its web-based nature ensures the application must not necessarily be platform-specific. Hence, this serves our research aims of having motivational and visual user interfaces with the inclusion of context sensitivity in apps adapting app content

into a format that is accessible and usable across multiple platforms for the client.

Development methodology

Our web mobile application is a set of HTML, CSS, JavaScript and other related files that reside on the Node.js server. These web pages are formatted for smartphones and tablets, and they can be accessed through the mobile devices' web browser. For the web implementation, we focused on the most recent and popular way of Responsive Web Design [16], as a way to approach the mobile web presence of our application. Web content in the application can be adapted to make it more accessible to mobile users.

A. Used Technologies

Based on the architecture of the app as described in the Phase 1: Developing a Native Android Mobile Application section, Android mobile application implementation section, the web application was developed using JavaScript as a programming language and the technologies below:

- **Node.js:** Event driven server-side JavaScript environment.
- **EJS:** Embedded JavaScript. A templating language that generates HTML markup with plain JavaScript. It is primarily HTML, but with additional features which allow us to efficiently re-use pieces of our project.
- **Session.js:** For secure and persistent login (keeping users authenticated/logged in).
- **Express:** Node.js web framework.
- **Passport:** Authentication middleware for Node.js. Help in authenticating with different methods.
- **Google's OAuth2 Authentication Strategy:** For creating a login flow for users and using the profile information to provide users with personalized functionality.
- **Google Firebase:** NoSQL-based real-time database that provides a backend as a service. For patient and educational data storage and retrieval.

B. Application Structure

The application source code is hosted on a server, which makes it possible to make updates to the app without the process of submission and approval typically required by the app store. Patients and patient navigator (i.e., nurse) can access the application with his/her Google account. The following are some of the features of the Node.js web application:

- Google login and registration (using OAuth2 with *passport-google-oauth2*)
- Require login for certain sections of the application (i.e. paths to different sections like symptoms, educational etc.)

- Linking Google account to user's application account in Google Firebase through the use of Unique User ID (UID)
- Registering newly signed in Google user to Firebase's *user* database.
- Storing user's symptoms data to Firebase's *symptoms* database for retrieval and analysis purposes
- Retrieving updated educational, quotes and adherence data from Google Firebase

To set up the base application, we set up *npm packages*, *node application*, *configuration files*, *database structure* and *routes* as mentioned in the following section.

Routes

Routes are the application end points and they determine how the application responds to user requests from the client. For our web application, we have the following routes:

Route	Description
/	Sign-in page
/menu	Menu page
/calendar	Calendar of events
/patnav	Patient navigator contacting
/support	Tech support
/symptoms	Symptoms
/educational	Educational and motivational content
/logout	Logout

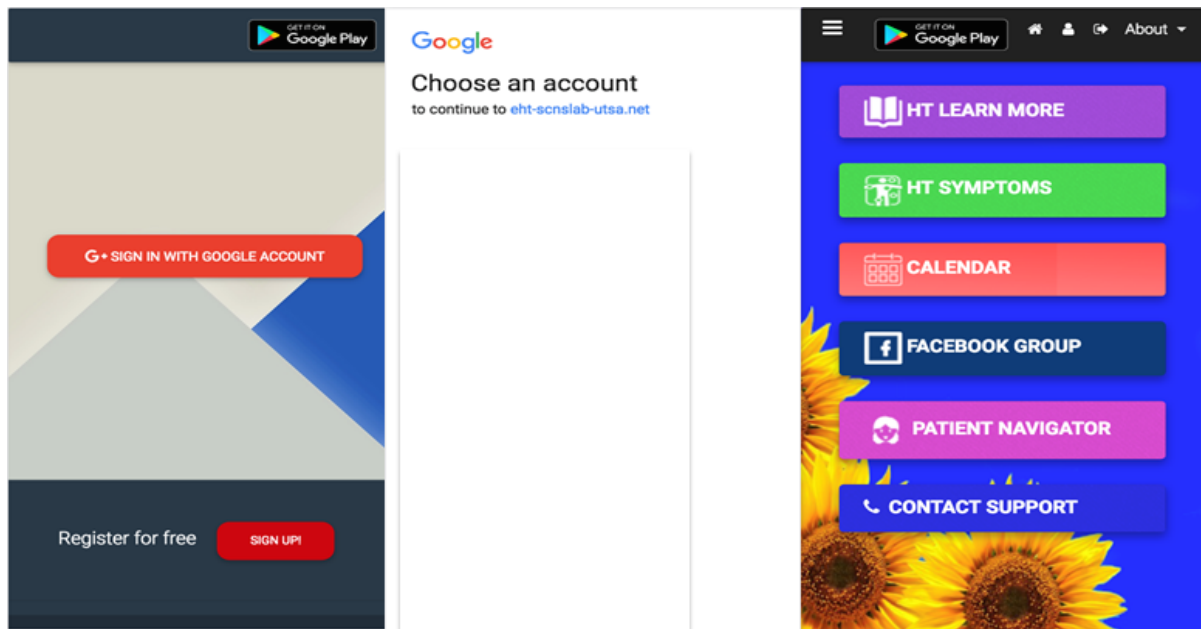
All the routes to different pages of this web application is protected using **route middleware**. A user has to be logged in for accessing all the pages. The user is redirected back to <http://app.eht-scnslab-utsa.net:3000/> if they try to access <http://app.eht-scnslab-utsa.net/menu> or any webpage on the website without logging in.

Every web page of the website has the *Google Play Store button* that links to the originally described Android mobile application.

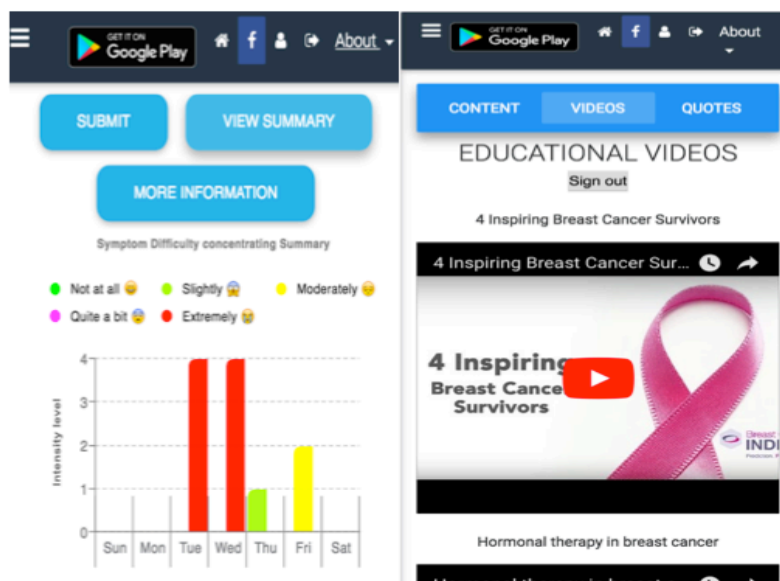
All the web pages mimic the mobile application's activities. However, the symptoms page also allows patients to view their weekly summary for each symptom. It can be viewed by clicking **View summary** button for each symptom. This page also implements **two-factor authentication**. The patients need to sign in using their Google account to get the updated list of the symptoms. It is similar to the **Google OAuth2 authentication for viewing menu** page. Two-factor authentication ensures that the user's Google account is linked to user's Firebase account. A unique UID gets generated whenever the system encounters a new user.

The Interactive EHT Patient Helper Application

The web-based implementation is released as a separate website [17] on Amazon Elastic Compute Cloud Instance with Node.js server. Fig. 7 shows the EHT Patient Helper Android application.



a)



b)

Fig. 7. The Complete interactive EHT Patient Helper Application (left to right): a)1) Screen 1: Application Sign-in Page; 2) Screen 2: Google Authentication with options to add a Google account and to choose from existing accounts; 3) Screen 3: Menu Page with a panel to choose one of the features of the application e.g. Tracking Symptoms, Accessing motivational educational content or videos; b) 4) Screen 4: Educational Page with motivational videos, quotes and adherence

System Evaluation and Validation

In this section we consider native mobile apps as the ones that are built for a specific mobile Operating System (OS) and run at native speed and mobile web apps are those that run on the web and are designed to appear perfectly on any device (smartphones, desktops, laptops and tablets).

A. Cross-platform Development

Developers often ask themselves whether to develop a native app or a web-based app. The fragmented mobile market with numerous choices for mobile devices' environments and platforms makes it difficult for an app to be portable.

In such case, developing a single application with HTML5 and JavaScript is a promising choice. Each one of these platforms has a web browser and hence, can run web apps. Although, web apps can run on any mobile platform, native apps have their own advantages too. Table 1 discusses the advantages of both native and web apps based on research by [18] and own research.

Table 1
Advantages Of Native Mobile And Cross-platform Web Apps

Native Mobile App	Cross-platform Web App
Extensive access to device hardware/ sensors and native mobile platform features (e.g. contact list, file systems or notifications)	Reaches broad community of audience (smartphone as well as desktop/ laptop users)
Better performance	Easy multi-platform development
Apps' visibility via platform's app stores or marketplace such as The App Store or Android Apps on Google Play	Easy upgrades across all platforms

B. Evaluation Environment

For the evaluation and validation of the proposed web application prototype, we used three different platforms: 1) Android OS, Moto G model, Version 5.1. 2) iOS, Apple iPhone 6s model, Version 10.3.2. and 3) MAC OS X, Macbook Pro model, Version 10.9.5. The application was observed on Chrome Web Browser in the three devices. Native Android Mobile Application implementation was tested on Moto G model with Android Version 5.1.

C. Measures

Further, a set of technical and non-technical evaluation parameters described in [19] helped us identify the right type of solution:

- *User Experience*

The golden rule for better end-user performance is that 80-90% of the end-user response time is spent on the frontend. Application launch time is one of the most important performance parameter for any smartphone as far as user experience is concerned.

- *Delivery Platform*

Table 2 highlights some of the key differences that have been observed in terms of method of delivery of native mobile app and cross-platform mobile web app.

Table 2
Differences In Methods Of Delivery Of Apps

Native Apps	Cross-platform Web Apps
Downloaded onto a mobile device	Can be accessed through a devices' browser
Typically installed and run as a standalone application	Does not require any new software to be installed
Users need to manually download or approve the application and install app updates	Upgrades are made to the web server avoiding user intervention
App Stores and Marketplace help users to find the app	It can be hard for users to find the app

D. Results

We integrated application functionalities based on the feedback from the focus groups and in-depth interviews conducted by the UTHSCSA [20] research team with breast cancer patients, oncologists, nurses and patient navigators. Application components include self-reporting of EHT adherence, side effects, symptom tracking, and weekly symptom summaries in the form of graphs, educational and encouraging testimonial videos, emojis, Facebook support group information, and the ability to contact for technical assistance. The UTHSCSA study research team is regularly conducting studies to test and improve these mobile and web applications, with the purpose of improving adherence to EHT.

Conclusion and Future Work

Despite the proven benefits of therapy and medication, non-adherence still exists as a healthcare challenge. EHT adherence is still minimal (less than 80% daily doses taken) and about 33% of women still do not follow their prescribed medication. Smartphone-based and Web-based adherence applications like the ones discussed in this paper are scalable, inexpensive and easily accessible to anyone with mobile devices, either through the Android marketplace, or as websites. The apps discussed in this paper are being tested by the UT Health Science Center at San Antonio [21] in a randomized controlled trial. They are being considered as possible strategies to suggest to non-adherent patients and to incorporate in their medication. The discussed apps already possess the desirable attributes. However, feedback from focus groups and patients participating in the study can still contribute to the process of improving the mobile and web-based versions of the app.

Acknowledgements

The authors would like to thank Dr. Jafet Morales for his critical review, valuable inputs and suggestions. This work was supported in part by Susan G. Komen (Award No. SAB160005) and the Mays Cancer Center (Grant No. P30 CA054174).

References

- [1] Google. *Google Firebase*. [Online]. Available: <https://firebase.google.com/>.
- [2] The University of Texas Health Science Center at San Antonio, Oncologists, oncology nurses, patient navigators and breast cancer patients of the UT Health San Antonio Mays Cancer Center, “*Komen Adherence Study*,” 2017.
- [3] Text4baby. *Maternal and child health*. [Online]. Available: <http://www.text4baby.org>.
- [4] Research2guidance. *mHealth App Developer Economics 2014 Report*. [Online]. <http://www.research2guidance.com/r2g/research2guidance-mHealth-App-Developer-Economics-2014.pdf>
- [5] Istepanian, R.S.H., Laxminarayn, S., Pattichis, C.S. (2006) “M-Health: Emerging Mobile Health Systems”. *Topics in Biomedical Engineering*. Int. Book Series. London, UK: Springer-Verlag
- [6] The University of Texas Health Science Center at San Antonio. (2016). *Improving Adherence to EHT among Breast Cancer Patients*. [Online]. Available: <http://clinicaltrials.gov/ct2/show/NCT02850939>. NLM Identifier: NCT02850939.
- [7] Harte R, Quinlan LR, Glynn L, Rodríguez-Molinero A, Baker PM, Scharf T, ÓLaighin G. “Human-Centered Design Study: Enhancing the Usability of a Mobile Phone App in an Integrated Falls Risk Detection System for Use by Older Adult Users”. *JMIR Mhealth Uhealth*. 2017;5(5):e71.[Online]. Available: <http://mhealth.jmir.org/2017/5/e71.10.2196/mhealth.7046>. PMID: 28559227. PMCID: 5470007
- [8] P. Harbig, I. Harbig, EM. Damsgaard, “Suitability of an electronic reminder device for measuring drug adherence in elderly patients with complex medication,” *J Telemed Telecare*, vol. 18, no. 6, pp. 352–356, 2012.
- [9] A. Nakhimovsky, T. Myers, “Google, Amazon and Beyond,” *Apress. Springer Book Archive*, pp 131-160, 2004.
- [10] R.M. Califf, MD. Lawrence, H. Muhlbaier, “Health Insurance Portability and Accountability Act (HIPAA), Must There Be a Trade- Off Between Privacy and Quality of Health Care, or Can We Advance Both? ,” *Circulation, AHA Journals*, August 25, 2003.
- [11] S. Okuboyejo and E. Omatseyin. “mHealth: Using Mobile Technology to Support Healthcare.” *Online Journal of Public Health Informatics* 5.3 (2014): 233. PMC. Web. 10 July 2017.
- [12] S. Hirsch, A.Pereymer, S. Rajpal, “Systems and Methods for a Mobile Application Development and Development Platform,” *United States Patent*. Patent No.: US 8,261,231 B1. Sep. 4, 2012.
- [13] S. Gunasekaran, V. Bargavi, “Survey on Automation Testing Tools for Mobile Applications,” *International Journal of Advanced Engineering Research and Science (IJAERS)*, Vol. 2, Issue-11, Nov., 2015.
- [14] Software Communications and Navigation Systems Laboratory. Applications Development Group. The University of Texas at San Antonio. (April 2017). *EHT Patient Helper Google PlayStore App*. [Online]. Available: https://play.google.com/store/apps/details?id=edu.eht.PAT_help.
- [15] K. Lei, Y. Ma, Z. Tan, “Performance Comparison and Evaluation of Web Development Technologies in PHP, Python and Node.js,” Jan. 29, 2015.
- [16] J. Bryant, M. Jones, “Responsive Web Design,” *Pro HTML5 Performance*, Springer, pp 37-49, 2012.
- [17] Software Communications and Navigation Systems Laboratory, Applications Development Group. The University of Texas at San Antonio. (June 2017). *EHT Patient Helper*. [Online]. Available: <http://app.eht-scnslab-utsa.net/>.
- [18] A. Holzinger, P. Treitler, W. Slany, “Making Apps Useable on Multiple Different Mobile Platforms: On Interoperability for Business Application Development on Smartphones”, *IFIP International Federation for Information Processing*, 2012.

[19] Klaus Haller, "Mobile Testing," *ACM SIGSOFT Software Engineering Notes*, Vol 38, Nov. , 2013.

[20] Amelie G.Ramirez, Patricia Chalela, Dr. P.H. (February 2015). *Innovative Breast Cancer Survivor Study funded by Susan G. Komen for the cure and directed by the Institute for Health Promotion Research at the University of Texas Health Science Center at San Antonio in partnership with the Mays Cancer Center*. [Online]. Available: <http://www.uthscsa.edu/patient-care/ctrc/news-item/innovative-breast-cancer-survivor-study>.

[21] Amelie G.Ramirez, Patricia Chalela, Dr. P.H. (October 2016). *Researchers to create app to help breast cancer patients stick to therapy*. [Online]. Available: <https://news.uthscsa.edu/researchers-create-app-help-breast-cancer-patients-stick-therapy/>.

[22] Research2Guidance. *325,000 mobile health apps available in 2017 – Android now the leading mHealth platform*. [Online]. Available: <https://research2guidance.com/325000-mobile-health-apps-available-in-2017/>.

Corresponding Author

Devasena Inupakutika is a doctoral student in the Department of Electrical Engineering at the University of Texas at San Antonio with Professor David Akopian in his Software Communications and Navigation systems laboratory. She is pursuing her doctoral studies in Electrical Engineering (Communications concentration) with specialisation in design of messaging solutions and mobile computing. Her research includes web and mobile application development, tech mining, wearable technology and IoT wearable devices.

Email: devasena.inupakutika@my.utsa.edu