DEVELOPING A PLATFORM-INDEPENDENT SMARTPHONE APPLICATION TO SUPPORT VOLUNTEER EFFORTS

by

DANIEL S. ROSSITER

THESIS
Presented to the Faculty of the
Honors College
The University of Texas at San Antonio
In Partial Fulfillment
Of the Requirements
For the Degree of

BACHELOR OF SCIENCE IN COMPUTER SCIENCE
WITH HIGHEST HONORS IN THE HONORS COLLEGE

THE UNIVERSITY OF TEXAS AT SAN ANTONIO
COLLEGE OF SCIENCES
DEPARTMENT OF COMPUTER SCIENCE
MAY 2013

DEVELOPING A PLATFORM-INDEPENDENT SMARTPHONE APPLICATION TO SUPPORT VOLUNTEER EFFORTS

PREPARED BY:

Daniel S. Rossiter

APPROVED BY:

Kay Robbins, Ph.D., Thesis Advisor

Steve Robbins, Ph.D., Thesis Reader

Turgay Korkmaz, Ph.D., Thesis Reader

Accepted:

Richard Diem, Ph.D., Dean of the Honors College

Received by the Honors College:

ACKNOWLEDGEMENTS

During the process of developing this thesis I have received support from so many wonderful people, not least of whom have been the members of my thesis committee, including my advisor, Dr. Kay Robbins, and my readers, Dr. Steve Robbins and Dr. Turgay Korkmaz. The members of my committee have all taken time out of their already packed schedules to review numerous versions of my Thesis Proposal and later my final Thesis, make recommendations on how to improve the application, and just generally been willing to provide guidance where needed, and for all of this I am eternally grateful.

In addition to my thesis committee, I must thank the leadership of the UTSA Ambassadors organization, including Dr. Pat Graham, Sally Crouser, and Joey Whitehouse, all of whom have been tremendous assets throughout this process, helping me to coordinate whatever it was that I needed from the organization in order to make this thesis a success.

The UTSA Ambassadors application would not have been nearly as appealing had it not been for the hard work of two individuals that worked with me to develop graphics that have been used. Salina Cram developed the application loading screen which is displayed when the application is first opened and Brian Smith developed the four logos which are displayed on the application home page.

Finally, the person who I either owe the most blame or the most thanks (I have yet to decide which), is Harrison Pierce. The idea for the Ambassador application was a direct result of a meeting that we had together, deep in the recesses of the Computer Science department, back in the fall of 2011. Without his persistence, I never would have thought of, let alone attempted, this thesis project.

May 2013

ABSTRACT

DEVELOPING A PLATFORM-INDEPENDENT SMARTPHONE APPLICATION TO SUPPORT VOLUNTEER EFFORTS

Daniel S. Rossiter, B.S.
The University of Texas at San Antonio, 2013

Supervising Professor: Kay Robbins, Ph.D.

The University of Texas at San Antonio (UTSA) Ambassadors is a group of student leaders who volunteer their time to support the university in a number of different capacities. The purpose of the platform-independent smartphone application developed in this thesis is to facilitate their work. The application will be used by the organization's members to provide a more convenient method of keeping accurate records for the number of hours worked by each member on a monthly and annual basis.

Prior to this project, the organization had a website, but nothing that fully harnessed the power of smartphones to provide enhanced functionality on mobile devices. Through interfacing with the pre-existing website's application programming interface (API), this thesis creates a native smartphone application for Android- and iOS-powered devices that allows Ambassadors to interact with data housed on the website. Through use of the Rhodes development framework, both native applications were written using a single source code, with a very limited amount of code specific to one platform or the other, allowing for enhanced performance on both platforms, based on the features available for the device.

During the latter half of development, select Ambassadors were given the opportunity to test aspects of the application and give feedback, which steered the development process. Allowing Ambassadors to interact with the application prior to the full release allowed not only

for the adding of additional features based on what users felt was needed, but also made it possible to quickly discover bugs that inevitably appeared throughout development.

TABLE OF CONTENTS

ABSTRACT ..... **IV**

LIST OF FIGURES ..... **VII**

CHAPTER 1: INTRODUCTION **1**

CHAPTER 2: USE CASES AND DESIGNING THE INTERFACE **6**

2.1 USE CASE: USER LOGIN 7
2.2 USE CASE: MODIFYING USER PROFILE 8
2.3 USE CASE: REGISTER FOR EVENT 9
3.1 THE RHODES DEVELOPMENT FRAMEWORK 12
3.2 DEFINING FUNCTIONAL REQUIREMENTS 13
3.3 INTERFACING WITH THE UTSA AMBASSADORS WEBSITE 14
3.4 MAINTAINING THE LOCAL APPLICATION DATABASE 15
3.5 USER FEEDBACK 16
3.5.1 DESIGN USABILITY AND ESTHETIC APPEAL FEEDBACK 16
3.5.2 NECESSARY FUNCTIONALITY FEEDBACK 17

CHAPTER 4: DISCUSSION AND FUTURE WORK **18**

4.1 APPLICATION FEATURES 18
4.2 CONTINUING RESEARCH 18
4.3 SUBMISSION TO ANDROID AND APPLE DISTRIBUTION PLATFORMS 19

REFERENCES **21**

LIST OF FIGURES

Figure 1 UTSA Ambassadors Site: Event Index 2

Figure 2 UTSA Ambassadors Application: Non-Esthetic Home 4

Figure 3 UTSA Ambassadors Application: Esthetic Home 4

Figure 4 UTSA Ambassadors Application Mockup: Event Index 6

Figure 5 UTSA Ambassadors Application: Event Index 6

Figure 6 UTSA Ambassadors Application Mockup: User Profile 8

Figure 7 UTSA Ambassadors Application Mockup: Event Show 9

Figure 8 UTSA Ambassadors Application: Event Show 9

Figure 9 Event Registration Activity Diagram 10

Figure 10 User Authentication Activity Diagram 15

CHAPTER 1: INTRODUCTION

The UTSA Ambassadors is a university-sponsored registered student organization at The University of Texas at San Antonio. The organization was founded in 1988 with the intent of being the student face of the university, giving tours to prospective students, meeting with various politicians and potential donors who wish to see the university, and generally serving the university in whatever capacities are needed. In the 2011-2012 school year the UTSA Ambassadors contributed over 10,000 hours of service to the university, making the organization one of the most active sponsored organizations on campus. Each of these 10,000 hours had to be tracked by the organization in order to justify funding from the university and to allow individual Ambassadors to document their service contribution. Historically, these records have been maintained by manually signing in on a piece of paper at each event attended. In the late 90s a website was created to maintain these records more efficiently, and this past year the website received a much-needed overhaul. The overhaul solved many long-standing issues, including making it usable on mobile devices and making the backend much cleaner, thus more maintainable. However, even the new website is lacking in some ways that simply cannot be solved by means of a mere website. Figure 1 demonstrates one limitation of the website: the event listing on the website, which is not optimum for mobile devices. Because of the limited screen dimensions provided by a mobile device, the data displayed on a given page must be thinned to the bare essentials. Short of creating an independent mobile-only site, this is not practical since removing the detail would work well on a mobile device, but would leave the screen utterly bare on larger desktop screens.

A growing trend for organizations that wish to surpass the limitations of a simple website is to create a native smartphone application. While websites are limited in what they are able to

access in a smartphone's arsenal of many useful features, native applications can, by their very nature, access the full artillery, enabling the application developer to provide functionality never

Title	Date	Time	Members needed	Hours	Location	Instructions	Ait
Diploma Dash - work Children's area not running in dash	March 02, 2013	6:30 am - 8:30 am	4	2.0	In Front of Parking Garage by Main Building	Dress warm. You will be applying tattoos to kids. NOT racing.	✔
Diploma Dash - running in dash. Team 2	March 02, 2013	7:30 am - 8:30 am	5	0.0	In Front of Parking Garage by Main Building	Need 2 teams of 5 to run the dash. This is team 2. No credit for running the dash.	✔
Diploma Dash - running in dash. Team 1	March 02, 2013	7:30 am - 8:30 am	3	0.0	In Front of Parking Garage by Main Building	Need 2 teams of 5 to run the dash. This is team 1. No credit for running the dash.	✔

Figure 1: The index of upcoming events on the second version of the UTSA Ambassador website.

dreamt of by a website developer. This thesis harnesses a number of these extra features, including modifying the user's in-phone address book, calendar, and calling capabilities, each of which will be described in detail. It is for these added benefits, among others, that it was deemed beneficial for the Ambassador organization to have a smartphone application, and thus it is the topic of this Honors Thesis.

In order to make this application available to as many Ambassadors as possible, the decision was made to use a development framework that supports platform-independent application development. Using a development framework allows for a single source code to generate a native application within any of the smartphone Operating Systems supported by the chosen framework. This ability to write an application source code once for multiple native applications is very powerful, especially when either money or time is a limited asset (as they often are).

Since the smartphone market is dominated by Android devices, followed closely by iOS, the choice was made to limit support for this application to those two platforms. According to The Nielsen Company, the market share of Android devices for persons aged 18 to 34, which is the target audience of this application, was 47 per cent nationwide and the iOS share an additional 29 per cent. In combination, the application should support approximately 76 per cent of smartphone users (Nielsen 2011). A survey of current Ambassadors showed similar numbers. Of the 31 Ambassadors who responded to a survey conducted through the Ambassador Facebook group, 19 members (61 per cent) chose iPhone and 9 members (29 per cent) chose Android. The remaining 10 per cent was split with 1 vote for BlackBerry and 2 votes for a non-smartphone. The 31 votes cast represent approximately half of the membership and I believe provides a representative sample of the organization as a whole.

In designing the proposed platform-independent smartphone application, consideration was given to three key components: necessary functionality, design usability, and esthetic appeal; in other words, what the application does, how the actions are triggered, and how it looks while doing these actions. Because this application is being designed as an alternative for a pre-existing website, it is relatively clear what the user must be able to do in order for the application to be an asset to the organization: it must provide most of the functionality available through the website. However, because of the nature of a mobile platform, which inherently provides less processing power and memory than the average desktop or laptop computer, the *way* in which the functional requisites are achieved must be carefully considered in order to make best use of the platform strengths, while bypassing its weaknesses.

In the case of design usability, this is again influenced by the differences between a mobile platform and the standard computer interface. Specifically, the difference in screen

dimensions between the platforms requires special thought in order to create an intuitive, usable interface. For a smartphone application to be designed well, it must reduce page content significantly in order to accommodate this limitation. The way in which the end user interacts with the application must be completely re-imagined when implemented in a mobile application form (Fling 2009).



Figure 2: The first version of the application home page was fully-functional, but it lacked esthetic appeal.

Finally, esthetic appeal must also be considered. Although this could be included within design usability, it is really a separate issue. Esthetic appeal does not so much affect the usability of the application as it does the desire for a person to use it. As an example of this difference, Figures 2 and 3 show two designs for the home page of the application, both of which are equally functional, but only one of which has esthetic appeal. In designing the UTSA Ambassadors application, all three of these aspects have been considered and implemented in order to produce a quality product.



Figure 3: The second version of the application home page is far more esthetically appealing.

In the following chapters, we will discuss how

these challenges have been addressed, as well as the challenges encountered during the process. In Chapter 2 we will explore the use cases developed alongside the design choices that they resulted in. In Chapter 3 we will discuss the application development methodology and implementation, including the development framework chosen, its specifications, the programming choices made while using it and the proposed beta testing process for the application. Finally, Chapter 4 will describe the outcome of work done during this Thesis.

CHAPTER 2: USE CASES AND DESIGNING THE INTERFACE



Figure 4: The proposed index design for events within the application.

When developing a functional and esthetically-pleasing design for a mobile application, the developer is faced with a small fraction of the screen dimensions available on a standard computer screen. This fact was at the forefront of the design proposed for the UTSA Ambassadors application. Figure 4 shows a mockup of the proposed event index and Figure 5 shows a screen capture of the actual event index for the application. It presents a very different look than the index seen on the website (Figure 1). While the website provides the user with almost all of the information about a given event, the application only provides the most important details (event name and date-time range). If the user is interested in obtaining additional information about a specific event, he or she must click on an event to view more details. The realization of this proposed design can be seen in the screen capture of the application seen in Figure 5. This design concept was again implemented with the member index, which only provides a listing of first and last name as well as an option for members to be filtered based on the committee they are a member of within the organization.



Figure 5: The final version of the application event index.

The design for the UTSA Ambassadors application required first re-imagining the implementation for each of the necessary functions for the end user. To facilitate this process, use cases were considered for each of the primary user interactions that must be handled by the application. The designs produced through these use cases indicate which aspects of the application must be most accessible, while showing which features are less regularly used, and thus can be given a lower level of visibility within the interface.

2.1 Use Case: User Login

When users first access the application, they will be presented with a login screen. On this page, a user will enter his or her email address and password in the fields provided. The application will send the credentials to the server for verification. Once verified, the user will be directed to the application index with a number of possible actions available. During this process, the following alternate courses might occur:

- If the user reaches the login page and submits his or her credentials with either the email or the password field empty, this action is caught by the HTML5 frontend, which prevents processing of the data by the backend and displays an error. At this point, the user can either fill in the required fields and submit again or click the “Back” button. The email, if initially entered, will be retained as written for resubmission, but, for security reasons, the password previously entered will not be retained.
- If the server responds with an error message after the user submits credentials, the user will be directed back to the login page with the server response as an error message displayed at the top of the screen. This alternate path would include a case where the user submits invalid credentials, but may include other server errors as well. The email will be retained as written for resubmission.

2.2 Use Case: Modifying User Profile

When a user is viewing his or her profile, which may be accessed through the "My Profile" link in the application index or through finding his or her name in the Ambassador index, the user clicks the "Edit" button at the top of their profile. On the edit page, the user makes any desired changes then clicks "Submit." The changes are delivered to the server, and the user is directed back to their profile, a mockup of which can be seen in Figure 6 with the submitted changes now visible. A message is displayed at the top of the page indicating the success. During this process, the following alternate courses might occur:

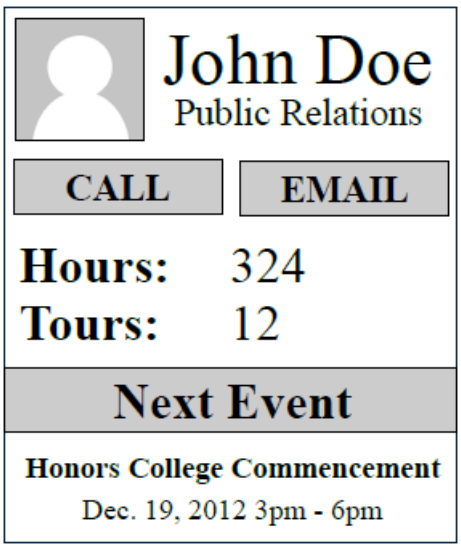


Figure 6: The proposed design for a user profile.

- If the user reaches the edit page and chooses not to change any of fields, instead clicking the "Cancel" button, he or she will be directed to the previous page accessed with no changes made.
- If the user reaches the edit page and enters an invalid date in the birthdate field or does not enter a first name, last name, or email address, this action is caught by the HTML5 frontend, which prevents processing of the data by the backend and displays an error. The user can either fill in the required fields or click the "Cancel" button. Any edits submitted will still be present for resubmission.
- If the server responds with an error message after the user submits changes, the user will be directed back to the edit page with an error message displayed at the top of the screen describing the error returned. Any edits submitted will still be present for resubmission.

2.3 Use Case: Register for Event

Starting from the event index, a logged in user will click the desired event, which will take him or her to the show view for that event. On the show, the user will click the "Register" button, which will take the user to a confirmation page; a mockup of this display can be seen in Figure 7 and the final product in Figure 8. Once the user confirms the selection, he or she will be redirected back to the event show, now with the "Register" button changed to a "Swap" button and a message at the top alerting the user that the registration was successful. The option for Ambassadors to swap an event registration allows them to relist an event that they have previously committed to in the hopes that another Ambassador will take on the responsibility, but the original Ambassador remains committed to the event unless someone else chooses to take the event. This process, including the following alternate courses, is shown in Figure 9.



Figure 7: The proposed design for an event profile.

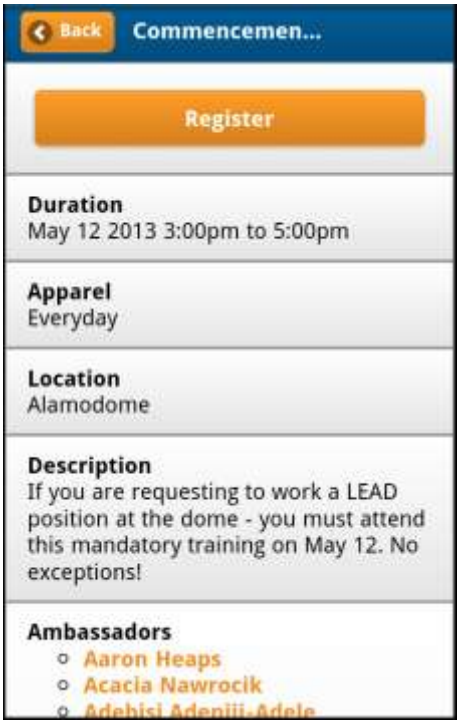


Figure 8: A screen capture of the event registration screen.

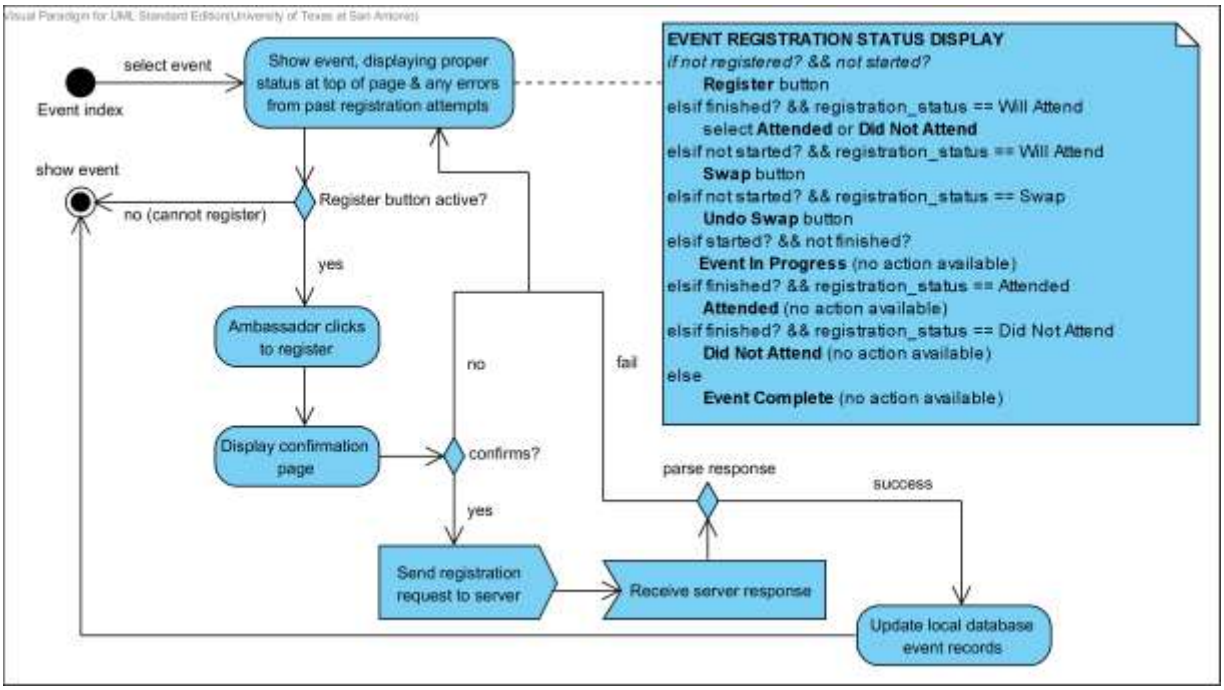


Figure 9: The process of registering for an event is described in the above diagram, taking into account the various alternate courses that may be encountered.

- If the user reaches the confirmation page, then chooses not to confirm their selection, he or she will be redirected back to the event show view.
- If the open spots for a given event have filled since the last time data was synced to the application then, after the user has confirmed his or her selection, he or she will be redirected to the event show view with an error at the top of the page notifying him or her of the registration failure and the "Register" button is no longer present.
- If a server error occurs that prevents the registration from being processed, the user will be redirected to the event show view with an error at the top of the page notifying the user of the failure and the "Register" button will still be visible.

2.4 Use Case: Complete Event

When a user is viewing an event that he or she previously registered for after the registration period is complete, the user will be presented with the option to select either "Attended" or "Did Not Attend" based on whether he or she was actually at the event. Once the

user makes his or her selection, he or she will be directed back to the event page, this time with a confirmation message verifying that his or her selection has been recorded.

- If a server error occurs that prevents the selection from being processed, the user will be redirected to the event show view with an error at the top of the page notifying the user of the failure and the two select options will still be visible.

CHAPTER 3: METHODS AND APPROACH

When developing a platform-independent application, the first consideration is determining which framework to use. A framework is necessary in order to abstract away differences between the ever-evolving platforms for which the application is being designed to work with. These differences range from how to interact with mobile device feature APIs such as the camera and the address book to the programming language that the platform expects applications to be written in. While the Android platform expects applications written in Java, iOS applications must be written in Objective-C.

There are a number of cross-platform development frameworks available. With these frameworks, an overarching trend is to manage cross-compatibility by handling the presentation of content by way of an embedded browser. This approach allows the developer to handle the design in web-based languages which are interpreted similarly across all mobile devices. HTML, CSS, and JavaScript make cross-compatibility easier to achieve and provide familiar languages for developers who have previously done website development. In the end, the Rhodes framework, which is one framework that utilizes this method to achieve cross-platform compatibility, was chosen for this thesis.

3.1 The Rhodes Development Framework

The Rhodes framework uses the embedded browser approach with a Ruby on Rails (Rails) backend, to create native applications for various mobile devices, including the Android and iOS devices, which are the focus of this thesis. The framework also includes jQuery Mobile, which has shipped with the Rhodes framework since version 3.1. In addition to the benefits of browser-based cross-compatibility, the Rhodes framework provides interfaces to many mobile device features, including access to the phone book, calendar, and camera on a given device.

Through the various Rhodes wrapper classes, which abstract away the platform-specific implementations, developers can hook into the native functionality for any of the supported devices, making it possible to create powerful applications that utilize the full potential of what a mobile device has to offer.

It is important to note that, although the Rhodes framework runs on Rails, it does not support the full language. One of the most notable limitations is the lack of any support for Ruby Gems, a feature that Ruby and Rails developers often heavily rely on to quickly and easily implement common tasks. In addition to a lack of Gem support, the Rhodes framework also strips away some of the common helper functions that come standard in the full Rails environment. A notable example of this is the Rhodes Object Relational Mapper (ORM), Rhom, which provides limited features when compared to Rails' ActiveRecord, which is a far more intuitive, full-featured ORM. The first limitation discovered when beginning development with Rhom was the lack of dynamic find_by functions, which simplify database query syntax for the programmer. A reason for these and other missing conveniences is the necessity of making the application as lightweight as possible (Rhomobile a).

3.2 Defining Functional Requirements

Once a development framework was selected, the next step was to use the framework language in order to interface with the UTSA Ambassadors website, allowing the end users to authenticate themselves, then view and modify data stored in the server database. A subset of these remote data from the server are retrieved and stored locally on the mobile device in order to ensure that the application runs at an acceptable speed, even in areas where bandwidth may be limited, which is a common occurrence with mobile devices that may find them in a dead or slow zone of cellular service.

When determining what exactly the application needs to *do*, the first step taken was to look at what the website offered to its users. Through this analysis, five functions available to general users were deemed necessary in order for a fully-functional user experience to exist. These functions are the abilities to view all registered users and their directory information, edit one's own user profile, sign up to work a specific event and sign off after completing said event, view the ambassadors with the most hours worked and the most tours given in the present school year, and view general organization information. The website also provides administrator accounts with the ability to do additional tasks, but, given the limited scope of usability for these functions they were excluded from the application design. Once aware of these requisite functionalities, the design for the application's local database become instantly clear. At minimum, the database must support storing information about Ambassador, Event, and EventRegistration objects in order to support these requisite functionalitites.

In addition to supporting functionality already available to website users, in order to be truly valuable, the application should harness the unique opportunities offered by the smartphone platform. This was achieved through allowing users to automate the adding of Ambassadors in the directory to their contact list as well as allowing events to be automatically added to the user's calendar.

3.3 Interfacing with the UTSA Ambassadors Website

Not coincidentally, the UTSA Ambassadors website is also written in Rails, making it slightly simpler to interface the application with the website given the language commonality. The programming language commonality aside, there were a number of challenges to overcome in successfully interfacing the two. At the end of the first semester of thesis work, there was still plenty left to be done, but following the second semester of thesis work, the website has been successfully interfaced.

The website provides a JSON-based API for accessing the majority of the data available on the frontend, making population of the application as well as synchronizing new or modified data back to the website relatively simple. Since JSON is natively understood by the Rhodes

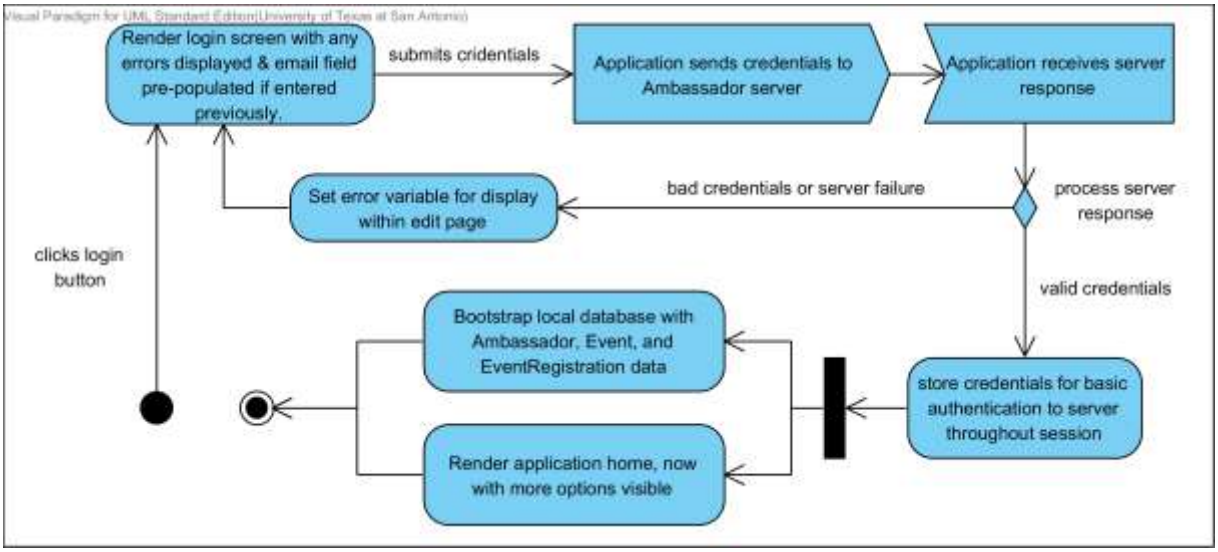


Figure 10: Activity diagram depicting the process of user authentication with UTSA Ambassadors server.

framework, no additional parsing is required before using any returned data from the server (Rhomobile b). Using the website's API, the application bootstraps its local database upon user login, pulling down all of the Ambassador information as well as all upcoming event information. Additionally, event registration data for the authenticated user are retrieved, allowing the active user to see and modify which events he or she will be attending. Since event registrations are extensive and would cripple the application if all pulled simultaneously, additional registrations relating to other users are only retrieved from the website as needed. The bootstrap process is depicted in Figure 10.

3.4 Maintaining the Local Application Database

Once the initial bootstrap of the local database is complete, as described above, the application retrieves additional data as needed. For example, each user profile must display the most recent events registered for by the profile owner. Since the initial bootstrap only retrieves

the authenticated user's event registrations, viewing any profile other than the users' own requires the application to retrieve registration data for the profile owner prior to rendering the page. Once data for a given page has been retrieved, these data are stored locally to ensure that future views render as quickly as possible. Each time that the application is re-accessed, local data are checked for validity and updated as needed.

3.5 User Feedback

Near the end of the final thesis semester, various members of the UTSA Ambassador leadership were provided a demonstration of the application. Following the demonstration, the participants provided feedback on aspects of the application that they liked and changes they would like to see before the final release. In the first round of demonstrations, a total of seven Ambassador Coordinators were asked to provide feedback on the application in its alpha stage of development.

3.5.1 Design Usability and Esthetic Appeal Feedback

For the user interface, there was a high focus on the events section of the application, followed by the user profiles and the home page of the application. While some suggestions made had been planned, but not implemented at the time of the demonstration, others had never been considered previously.

One of the new suggestions given during these demonstrations was to remove the About Us page and have the startup page go straight to the Login screen. Prior to this change, the start page had a single link to the About Us, which was removed as users who go to the trouble of downloading the application are likely already aware what the UTSA Ambassadors organization is.

Additional recommendations given which aligned with pre-existing plans were to include on the event index both what type of event credit would be earned and the date-time for the event

and to create a quadrant design for the home page which would show four tiles, including “My Profile,” “My Events,” “Directory,” and “Events.”

3.5.2 Necessary Functionality Feedback

In reviewing the necessary functionality, universally the demonstration participants wanted to see the ability to sync the Ambassador directory into their phone contact list and have events that are signed up for automatically added to their phone calendar. These features were planned, but not implemented at the time of the demonstration. Additionally, a suggestion was given to include an “hours worked” count for the logged in user, which had been postponed due to issues retrieving this data through the site’s API, but was later resolved.

CHAPTER 4: DISCUSSION AND FUTURE WORK

Upon completion of this thesis, the application will be available for future generations of UTSA Ambassadors to use. In addition to the obvious requirement that the current website stays live and continues to provide the API in its current state, some additional requirements must be met in order for this project to be a success.

4.1 Application Features

The application in its current state meets most of the specifications described in the initial Thesis Proposal submitted in the Fall of 2012. These specifications state that the user must be able to view all registered users, edit one's own user profile, sign up for and sign off on a given event after completion, and view general organization information. The only feature included in the proposal that has not been met in this initial release of the application is viewing the Ambassadors with the most hours worked and the most tours given for the given year.

4.2 Continuing Research

As with any project, challenges were faced during the development of this thesis. Most of these challenges were overcome, but the way in which they were overcome was not always in the best way and room for continued work on the application exists.

As mentioned above, the inclusion of a high score for hours worked and tours given was not possible in this initial release. The reason for this was a lack of an API which would return these numbers pre-calculated. Because of this limitation, the only way to calculate a high score table would be to download all event registrations for all users, then individually pull event information for each event worked by any Ambassador one-by-one from the API, resulting in thousands of HTTP requests which is completely infeasible. The solution for this issue is to modify the server, creating API access to pre-calculated high score values.

Another aspect of the application that does not function in the best way possible is the gathering of events worked by the authenticated user. Because of the limitations of the Ambassador site API, any events that occurred in the past can only be retrieved one-at-a-time. For any users who have been with the organization for a year or more, this would likely total to hundreds of HTTP requests and create similar issues to those described with the high score table above. The resolution for this issue will again require tweaking the server to allow for bulk retrieval of events in the past based on some criteria.

Finally, the application currently purges its local database every time that the user restarts the application, which creates a very computationally- and network-expensive start-up sequence that could be improved significantly. Ideally, the application should retrieve an up-to-date Event, EventRegistration, and Ambassador listing on startup, then check each returned value against the pre-existing entries in the local database and only change the database when necessary. The application should also garbage collect some of the old database entries that are no longer relevant, such as removing Ambassadors no longer listed on the site and removing any Event information that is no longer needed. These changes to the application itself should only be implemented once the server API has been modified so that unnecessary work is avoided as much as possible.

Following thesis submission, the plan is to implement as many of these tweaks prior to the Fall 2013 semester so that the UTSA Ambassadors will have the best functioning application possible when the main school semesters begin. This being said, the application will be fully functional when it arrives in Google Play and the App Store, just not yet at its full potential.

4.3 Submission to Android and Apple Distribution Platforms

At the time of writing this, the UTSA Ambassadors application is scheduled to be submitted to both Google Play, which is the Android repository for application disbursement, as

well as to the App Store, which is Apple's equivalent. The UTSA Ambassadors have asked for a delay in submission in order for the organization to handle some internal discussions. Making the tool accessible from a well-known, easily-accessible source will not only enable future Ambassadors to quickly find and start using this tool, but will also provide the organization with added publicity.

REFERENCES

Fling, B. (2009). *Mobile design and development, practical concepts and techniques for creating mobile sites and web apps*. (pp. 66-67). Sebastopol, California: O'Reilly Media, Inc. Retrieved from http://lib.utsa.edu

Nielsen Company (2011). The Mobile Report. *The State of the Media*, (Q3), 8-9. Retrieved from http://www.nielsen.com/content/dam/corporate/us/en/reports-downloads/2011-Reports/state-of-mobile-Q3-2011.pdf

Rhomobile. (n.d.). *Extending the Rhodes framework*. Retrieved from http://docs.rhomobile.com/rhodes/extensions

Rhomobile. (n.d.). *Rho::AsyncHttp API*. Retrieved from http://docs.rhomobile.com/rhodesapi/asynchttp-api