Project Part A: Application Proposal

Due: October 13, 2016 11:59pm

Objectives

In this project, you will use the skills acquired through lectures and labs to propose a new application. It can be any kinds of applications (e.g., web applications, mobile apps), it can be implemented in any languages (e.g., Java, Python) and using any frameworks (e.g., AngularJS, Ruby on Rails). Our goal is to help you gain insights into the software development process by working on this project. First, you will start by identifying the requirements from users' perspectives. Then, you will design a solution and create a working prototype of the solution. Finally, you may identify key issues around and take the prototype into a useful product.

Team Formation

For a quarter-long project, you will form a team of five to six students. The team should be formed before Part A is due. If your team needs a different size, please consult the instructor by the beginning of the third week. The team size cannot be changed in the middle of the project. However, the instructor can make an exception under extremely necessary circumstances, so please come and talk with us if exceptional situations arise.

Propose Your Idea

After your team is formed, you need to come up with an idea for your application that is both appealing and feasible to pursue. The feasibility of your project is crucial, because in Part B and Part C, your team needs to implement the proposed idea and produce a working and testable application. Therefore, when proposing your application in Part A, the team must investigate whether the application is feasible to implement and whether the team has the capability to finish the product. So please make sure to propose requirements such that the implementation could be completed in time and you are able to a live demo during the final presentation. It is important to find a good balance between creativity and feasibility!

Your team must set up the project repository on Github and include the TA of your lab section as a watcher (read permission only). You need to write a report to propose the idea and justify its feasibility. You also need to make mock-up designs to demonstrate the idea and include screenshots in the report. You need to prepare for a presentation to pitch the idea. Each team will

be scheduled for a 7-minute presentation and a 3-minute Q&A on the lab session of Week 3. Both the report and the slides must be submitted via CCLE.

Checklist

- Form a team and find an application idea of interest
- Set up a project repository on Github
- Make mock-up designs (e.g., screen snapshots)
- Write a proposal report
- Create slides and prepare for the presentation

Please submit presentation slides and report electronically on CCLE before the due date. The submission is per team, and there is no need to submit individual reports per person.

Report Structure

Your report should include the following sections.

Project Name:
Team Members:
Repository URL:

Motivation

You must include a description on why the proposed application is needed by the users, what the problem is, and what your vision is for creating the application.

Feature Description and Requirements

You must include a description of the feature and its requirements in English. You can start with two to four user stories and usage scenarios. You must also include a UML use case diagram to specify your functional requirements. You may also include state chart diagrams to specify behavioral requirements. You may specify other non-functional requirements such as performance, persistence, etc. in English as necessary. Please include mock-up screen snapshots of the proposed application and describe the associated user interaction scenarios in detail, so that the teaching staff can understand the proposed features and assess the difficulty level of the proposed implementation (e.g., similar to how user manuals are written.)

Feasibility

You must include a detailed description of how it is feasible to implement the proposed application. Here are several common factors that could lead to software development failures:

- too many features
- lack of communication between team members
- procrastination
- third party APIs/services that may not be reliable to use
- unfamiliar with APIs

You must describe your preliminary design by including UML class diagrams showing your design classes and their associations (relationships). If your team is leveraging existing libraries or technologies to implement the proposed feature, please include the specific descriptions of those libraries and technologies to be used and how they will be integrated into your application.

Capability

You must include specific, detailed descriptions of how your team is capable of implementing the proposed application. Please include relevant prior internship projects or class experiences to substantiate and justify your claims about each individual's capability and his or her estimated role in the team.

In your report, you must specify each member's responsibility for the Part A project assignment.

Sample Report

KnowMyCandidate
Part A

Fall 2015
UCLA CS130

October 13th, 2015

INTRODUCTION

We propose to build a mobile application for the iOS platform called KnowMyCandidate that would provide comprehensive information about the political campaign candidates. In light of the upcoming 2016 U.S. Presidential Elections, we believe this app will be perfectly timed and handy for voters. Our goal is to let users learn more about the candidates, their platforms, popular issues, and upcoming campaign or election events in a personalized and intuitive fashion. In this proposal, we will explain the motivation behind the development of this app and identify the problems that we seek to resolve. We will also provide an in-depth explanation of the design of our app using use case diagrams, class diagrams, state charts, and mock designs of the user interface. Finally, we will examine the feasibility of developing such an application in an eight-to-ten week frame and identify each team member's strengths and their potential contributions.

MOTIVATION

While tabloids and news outlets are catching up to the mobile age by moving to the Internet and wireless devices, U.S. politics and campaign strategies are lagging behind the rapid speed at which technology is developing. The low turnout on election days demonstrates the need for new ways to encourage people to vote and become excited about politics. In particular, youth vote (voters under 30) is the lowest amongst all age groups. Only 49% of the youth population (18-30 yrs) voted in 2012[1]. What is better than encouraging young voters with their phones? KnowMyCandidate is an iOS app designed to help people identify candidates who support issues they care about or wish to learn more about. Through a personalized matching process and daily questions, KnowMyCandidate aggregates data about the user's views and background. Then, through a newsfeed, issue polls, calendar, as well as candidate profiles, the app keeps the user in the know and educates them on important issues. We want to help people voice their opinions and make informed decisions about the future they wish to see.

There are no strong players in this market yet. Given the upcoming election in 2016, voters will undoubtedly benefit from KnowMyCandidate. Some news sources such as FOX News and CNBC have dedicated pages on their website and mobile app for campaign and candidates. However, those can be overwhelming and are not personalized. There are a few other websites that have attempted at some similar products. Nevertheless, some of them present information in a disorganized manner, and the others do not have a mobile client. These shortcomings are exemplified by two such websites, namely https://www.isidewith.com/, and http://www.ontheissues.org. ISideWith presents a vast amount of information nicely and concisely. However, they do not have a mobile app currently. In addition, http://www.ontheissues.org offers a gigantic database of all candidates in the past few years, as well as where they stand on key issues and their quotes reflecting their standpoints. However, this website too does not have a mobile app version of their product. In addition, the website itself is not very user-friendly, and nor does it have striking design.

What we are trying to achieve with KnowMyCandidate is to design a simple yet striking mobile app that will not only help users stay informed about their candidates and issues that are relevant to them, but also present information in a concise, reliable, and beautiful manner. We believe that by making a well-designed and user-friendly mobile app, users will be more

encouraged to actively participate in the Presidential Election process because they know the candidates well and understand the issues relevant to them better.

FEATURE DESCRIPTION AND REQUIREMENTS

We designed the features of this project and the specific requirements of each feature with the purview of providing a comprehensive yet clean and simplistic user experience; we selected only those features that we felt would be critical for the user.

The user will be able to create an account with the system by logging in through Facebook, at which point, if it is their first time using the application, they will be presented with a short survey to gauge their views and beliefs about certain key issues and select the importance they place on each issue. We will then use that information to present to them a set of presidential candidates whose platforms most directly align with their beliefs and preferences. Next, we will provide the user the ability to select candidates they wish to follow or simply, by default, follow the candidates that they matched with from the survey. Once this is done, we will redirect the user to their homepage.

The homepage will include a countdown to election dates and a news feed that will have news articles, twitter and facebook feeds, and other information regarding the candidates they are following. The news feed will also contain daily pop-up survey questions to get more information in a continuous manner from the user about their beliefs without them having to go through large tedious questionnaires. Furthermore, users will have the ability to comment or like news articles on the news feed and filter articles by candidate, issues, news source, or party.

Another important feature will be a profile page for each candidate that provides a holistic summary of the candidate. It will include aspects such as their position on key issues, information about their political and personal background, news related to them, links to their campaign sites and twitter feeds, and polls that show the number of people that support the candidate.

In addition, we will provide a calendar that shows a schedule of upcoming debates, speeches, interviews, events, and other key dates for the candidates that the user is following. The user will have the ability to filter this schedule by candidate.

Finally, we will have a feature that allow users to change their positions or beliefs on issues and re-determine which candidates' platforms best match their views.

The user stories, use-case diagram, state diagram, class diagram, and mock screenshots below explain the features and functional requirements of the application in more detail.

User Stories and Use cases

An example user of our app is someone who is not familiar with the presidential candidates but is interested in learning. He or she may feel overwhelmed by the number of candidates and the wealth of information about them, so our app helps him or her get started with the onboarding survey. The survey would match this user to relevant candidates and then show the candidates'

key information. The user would be able to quickly learn about candidates and also start to follow news about them.

Another example user of our app is someone who wants to easily follow news about specific presidential candidates. Our app allows this type of user to browse relevant information without needing to search through news sites and multiple sources. By viewing a news feed tailored to the user's preferences, the user can more easily focus on candidates and not miss important updates.

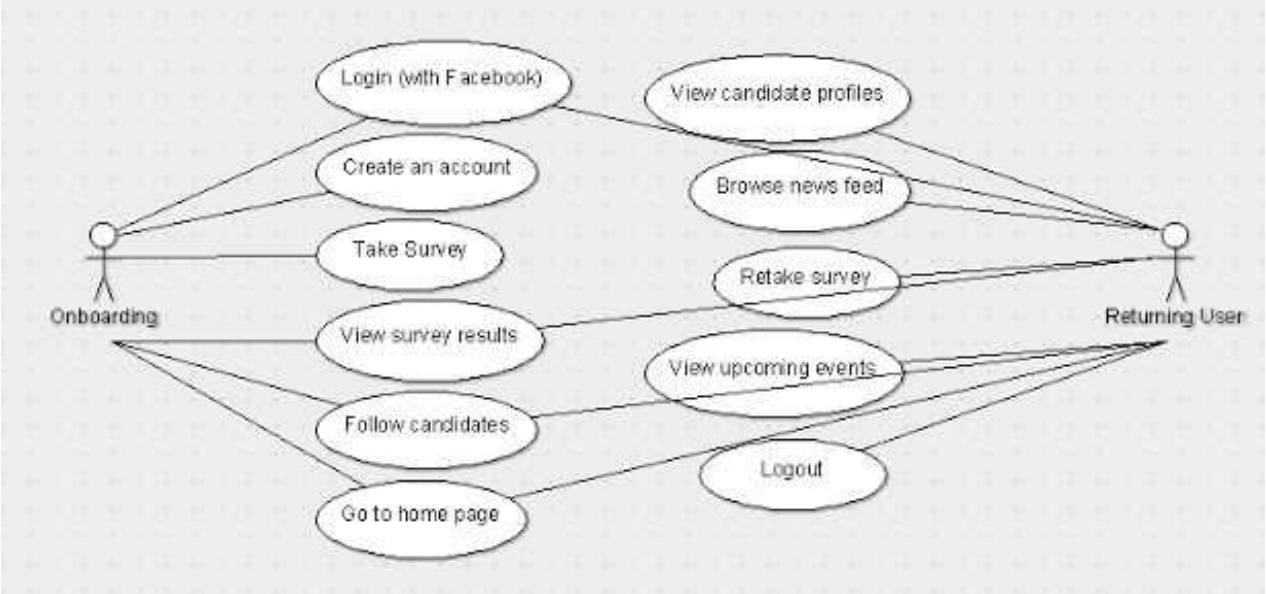


Figure 1: Use Case Diagram for onboarding user and returning user

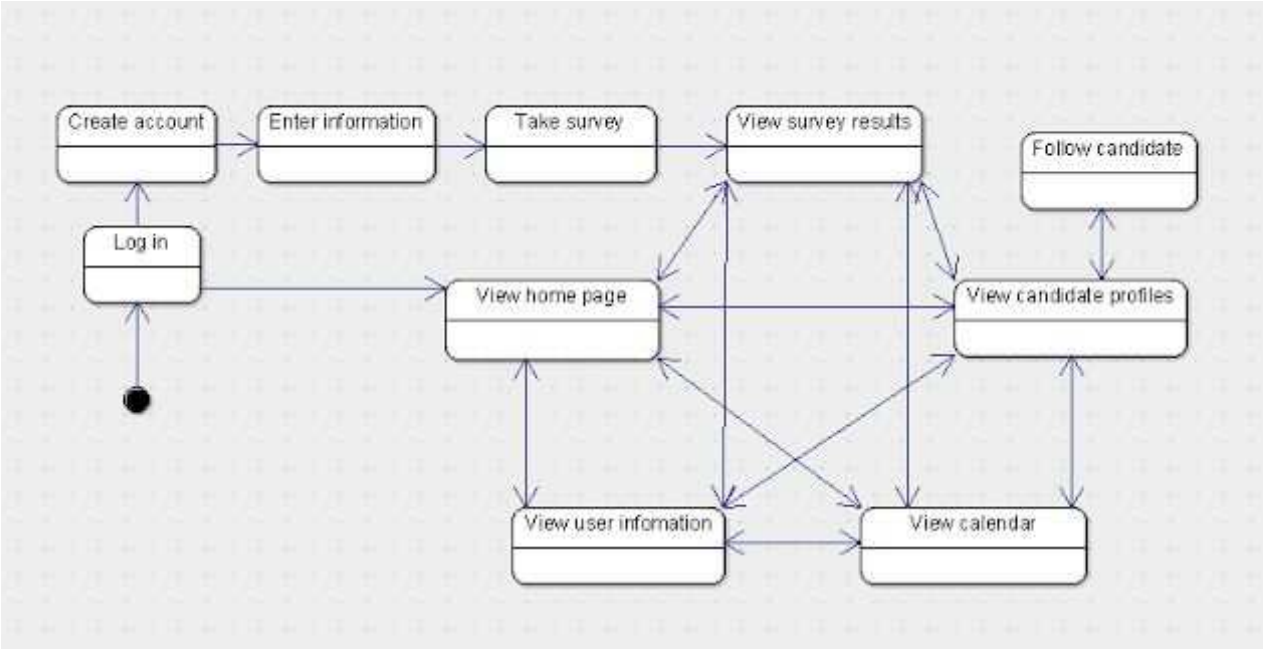


Figure 2: State Chart Diagram

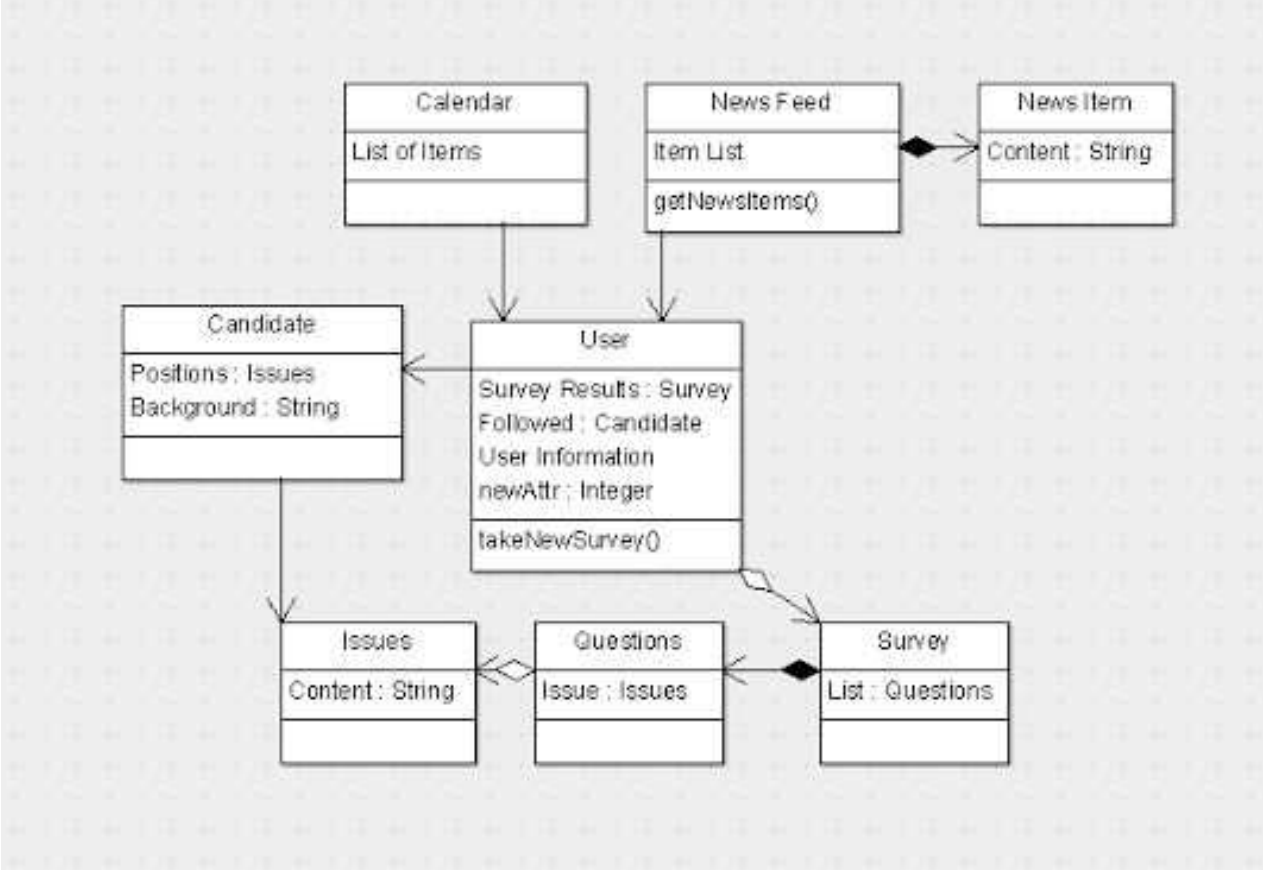
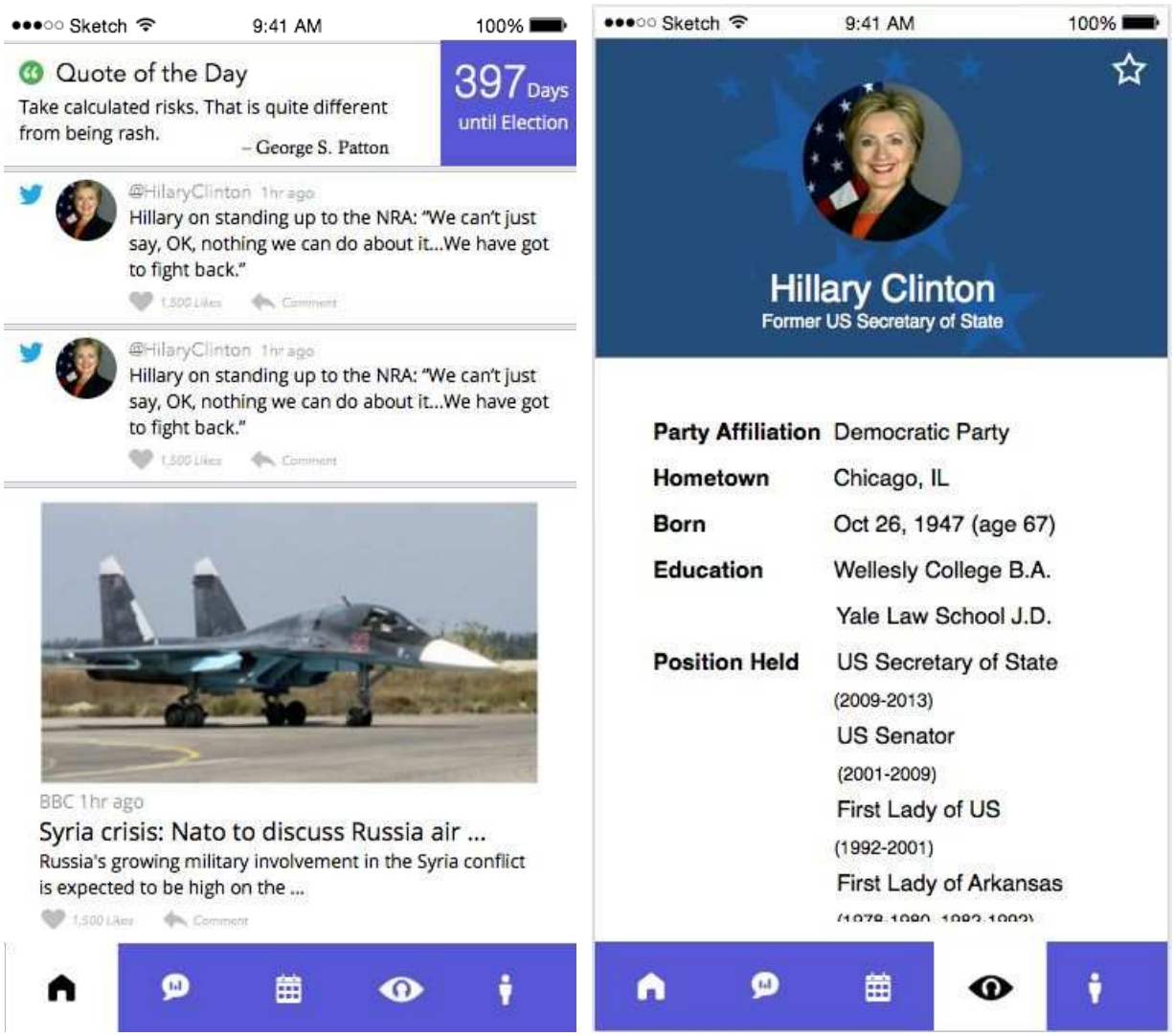


Figure 3: Class Diagram of KnowMyCandidate



Figures 4&5 : Design Mock-up for Home Tab and Candidate Profile Tab

To navigate through our features, the user simply has to select one of our five tabs in our user interface. The left-most tab is homepage, in which the user can see a countdown to the election date, social media updates about the candidates, and relating news in which the user has indicated interest.

The second tab is the survey results. It allows users to view their results from the survey and the summary of the general opinion on specific issues based on the survey data we collect. We select the summary based on the issues in which the user has indicated interest. Also, the users will have the freedom to retake the survey in the case when their views have changed and we will provide them with new candidate matches. The third tab is calendar which will give the user information about upcoming events and debates relating to their candidates of interest. The fourth tab is for candidate profiles. The user will have the option to follow the candidate by tabbing the star button on the upper right corner. Lastly, the fifth tab is a user profile which allows users to change their candidates and identify the issues they want to follow.

Non functional requirements

We expect KnowMyCandidate to be available at all times to everyone that has the app. In our initial stage, since we expect a small number of users, we can get away with the rate limits our API's have. For instance, we expect 5000 transactions per month from the Bing Search API to be sufficient to support our usage rate. Its processing and response time should be within reasonable range. It should not be CPU intensive and it should not lag the user's phone.

FEASIBILITY

After doing a thorough investigation of how we want to implement our application, we believe our project is feasible during an eight-week period. In this section, we will give a brief description of tools and techniques that we plan to use in order to address several main challenges that we confront so far.

One of the biggest technical challenges is obtaining reliable and accurate information about the candidates. This includes the candidate's profile information, the candidate's position on different political issues, latest news about the candidate, and upcoming events such as debates and voting dates.

To obtain the latest news about the candidates in the newsfeed, we will use the Bing Search API, the Twitter API, and the Facebook API. The Bing Search API is free for applications with fewer than 5000 transactions per month which is sufficient for this project. We can query for the candidate's name and sort the returned news result list by ranking and date. The Twitter API allows us to get the latest tweet for a given candidate. The Facebook API allows us to get the feed for the candidate's Facebook page.

To obtain the candidate's profile page, the candidate's position on different political topics, and the list of events, we will use web scraping through Python. For the candidate's profile page, we will scrape Wikipedia and Inside Gov. For the candidate's position, we will scrape the information from Inside Gov which has the candidate's positions on different issues in tabular form as shown in Figure 6. We will scrape the 2016 Presidential Debate Schedule for the events.

Topic	Position
Abortion is a woman's unrestricted right	Strongly Agrees
Legally require hiring women & minorities	Strongly Agrees
Comfortable with same-sex marriage	Strongly Agrees
Keep God in the public sphere	Strongly Disagrees

Figure 6: Screenshot of Website with Candidate Opinion

For the backend cloud storage, we will use Parse which is free for small applications. We will use Django and Python to scrape the website and pass the information to the Parse website. Parse can also schedule background jobs which will be useful for future scraping jobs.

Another technical challenge is the candidate matching algorithm that matches the user with the candidates with the most similar viewpoints to the user. We have all taken algorithms and we are confident we can create a strong matching algorithm.

CAPABILITY

Each of our team members has various relevant experience that enables successful implementation and full stack development of the product.

worked at Google, Yahoo, Salesforce, and NASA. She knows Python, MySQL, and JavaScript. She has built web applications with Amazon Web Services and Google App Engine. She is interested in working full-stack.

has software engineering experience from internships at Amazon, Netapp, and Symantec. He is proficient with Java and good object-oriented practices and thus can help with creating the backend for the app.

has experience in Java and Python and interned at Visa doing big data and machine learning. He is proficient in Python and Spark but he is looking to help with backend and expand his skills.

has prior software engineering experience from working at Yahoo and Glassdoor. He is comfortable with Java, Python and SQL, and has previously worked with web crawlers so he can contribute to scraping the candidates' information off the web and also to the development of the backend of the application.

knows Java and Python and interned at Amazon before. She has taken courses such as Scalable Internet Services and Web Applications and is interested in working within a group to build great applications.

has experience working with Objective-C and developing for iOS environments. He interned at Google for 2 summers developing for Google Maps for iOS and Google Play Games SDK, and is comfortable writing the client-side iOS code.

has experience in web designing and animation programming. She will be mainly working on app design and user interface. She also has experience in Java and Python so she can help out the backend. She is looking forward to learn Objective-C and develop in iOS environments.

has had experience in full stack and will ramp up on mobile development. She will use sketch to design app mockups and support backend programming. Her worked on the frontend at Facebook and backend at Microsoft. She understands the fundamentals of relational database and server-client communication with Python.

| has experience in developing algorithms from several research internships, and thus will contribute to the algorithms used by the application, particularly the method of matching users to candidates.

We are using BamBam for project management and git for source control. With our diverse and complementary skills, we believe KnowMyCandidate will be a sleek and professional iOS experience.

CONCLUSION

In this report, we explained the motivation behind deciding to develop KnowMyCandidate, which we believe fulfills a very important need. Spreading awareness and sharing knowledge about the upcoming elections and candidates is crucial in shaping the future of the country, and in turn, the people who make that decision. This, coupled with the fact that there is no other mobile app in the market that presents this information in a concise and useful manner, is why we believe KnowMyCandidate is an essential product that people would like to use. In addition, after examining the feasibility of developing this app and identifying the team's aggregate expertise, knowledge, and potential, we have concluded that this app would not only be possible to develop within the duration of this course, but would also be an exciting project that our team can learn from. After receiving approval for this proposal, our team will start with the design and programming of this mobile app.

Sources:

[1] "Voter Turnout Demographics - United States Elections Project." *Voter Turnout Demographics - United States Elections Project*. N.p., n.d. Web. 07 Oct. 2015. <http://www.electproject.org/home/voter-turnout/demographics>.