Cityzen - Project Plan

Francesco Agosti, Connor Gordon, Hailey Johnson

Project Executive Summary:

Overview:
Cityzen has developed a new way for city governments (such as Raleigh) to incorporate the opinions of constituents who are not typically the most vocal (most importantly, young voters). The Cityzen web application allows local governments to directly release polls to constituents concerning current issues and projects. These polls are distributed locally and users are then able to give input with ease from their computer, without attending events such as town halls or committee meetings. Cityzen's ultimate impact is to create a more cohesive community in which everyone's opinions can be heard and valued, ideally resulting in the creation of better, more useful projects which benefit a larger subset of constituents.

Cityzen Mobile serves to connect users to Cityzen's web service via location-driven push notifications. The mobile app will serve as an interface between the mobile user and the web service, which contains the majority of the functionality. The application's value lies in its ability to actively engage users within a more personalized polling system (receipt of polls and updates are based on the user's GPS location [or preset zip code] and user's interest preferences). This takes the burden of continued engagement off the user by placing the active responsibility of user-service interaction on the service itself.

User Experience:

1. Download Cityzen Mobile
2. Greeted by a splash screen containing the Cityzen logo
3. When the main page loads, log in or sign up on the app
    i. User info is saved by the device until explicitly logged out
4. On subsequent visits to the mobile app, the user will be able to do any of the following:
    i. Open their profile
    ii. View notifications which link to polls on the webpage vie
    iii. View the "about" page on the web
5. When the user is not actively using the application, Cityzen Mobile will run in the background and trigger OS-level notifications that will take users to the application where they will select the poll they wish to vote on.

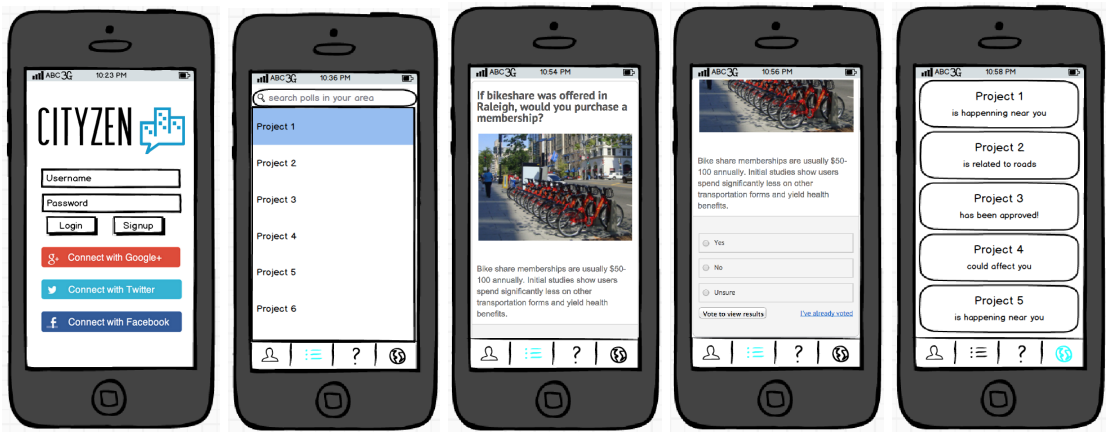
Secondary Goals:

1. Instead of transporting users to the browser to vote, users will be able to directly respond to polls from within the app
2. Users will be able to update their (e.g. parks and rec, sanitation, children, etc.) through the web page, so they only receive polls about matters that they are interested in.

Technical Considerations:

- Development Platform: Cordova/PhoneGap - Since the primary goal of Cityzen Mobile is to reach as many constituents as possible, cross-platform functionality and accessibility will be valued more highly than the native app experience.
- UI Framework: Ionic, a library of customizable UI pages that easily integrated with PhoneGap, will be used to make for a cleaner, more straightforward user experience
- External APIs: This app will integrate with Google, Facebook, and Twitter for the purpose of logins and user storage.
- External costs: All current databases are attached to and/or upheld by Cityzen, and should therefore incur no external costs

GUI draft:



Prioritization and Goal Management:

PRIMARY GOALS:	Subgoals:	Phase
native push notifications	• mobile app sends notification to mobile OS • user can access new poll through notification	P1
login functionality	• facebook login • google login • twitter login • Cityzen account login	P1
user account data	• communication of user data between mobile app and web app • send cookies with user data on requests to web server	P1

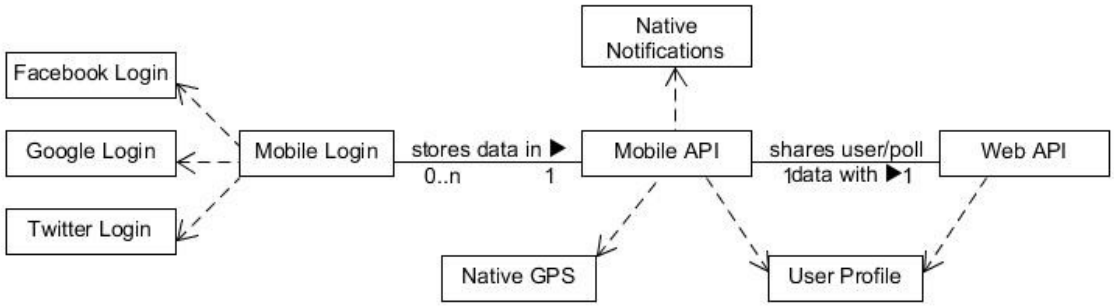
	• preservation of user data in the mobile app until user explicitly logs out	
vote in-app	• user can respond to a poll without accessing the website	P1
GPS-based profile	• Cityzen sends user polls based on mobile GPS data	P1
Default location profile	• Cityzen sends user polls based on the default city choice	P1
User preferences profile	• Cityzen sends user polls based on the profile preferences	P2
View recent polls	• app maintains list of recent polls • user can view voting history	P2
SECONDARY GOALS: (if time allows)		
Integrate linked poll podcast and video information		N/A
View past & inactive polls		N/A

Design:

Priority Goals:

- As of current status, Cityzen is a tool for only Raleigh. However the mobile app will be designed to be easily applied to and implemented by other cities
- The app will be built for forward compatibility of the web site; no core functionality should break or change when the web app is update

Design:



Dependencies:

Our project is currently *dependent* on the following resources and technologies

- Cityzen Web application resources and API (including databases) – to support interactions between the mobile app and the web app
- Cordova, our cross-platform framework – to implement the mobile app across multiple both Android and iOS platforms
- Ionic, an open library of mobile optimized HTML, CSS, and JS components – for flexible and elegant UI and front-end design
- ngCordova, a Cordova plugin containing AngularJS extensions -– to create/send native notifications to users as well as collect their location data.
- External social media login, synchronized using o*neall*
    - Facebook
    - Google
    - Twitter

Concerns:

The following are current questions/concerns that we have about the project

- Login - interaction and synchronization of user data between the mobile application and the web application
    - How can we keep these cross-API communications secure?
    - How can we best preserve user data over time?
- Accessing one poll at a time from within the mobile app:
    - the current Cityzen web application displays all relevant apps on a single page
    - This will need to be updated if we want the mobile app (which will be linking to Cityzen's web pages) to show one poll at a time
- Location-based polls
    - How long should a user be within a geo-fence before receiving a poll?
- Limiting User Activity so that each user can only vote once
    - The current web application does not currently implement this feature.
- Integrating with the existing database.
    - The existing database is being queried by PHP scripts that we will have to learn.

Project Roles:

Connor Gordon - Project Manager
Francesco Agosti - Technical Lead
Hailey Johnson - Business Analyst
Everyone - Quality Assurance

Division of Labor:

Component	**Owner**
Push notifications and settings	Francesco

Login (via FB, Twitter, G+)	Connor
Social Media Integration	Connor
Web Application Integration	Hailey
Location-Based Tagging (and notification preferences)	Hailey
User Interface	Francesco

The following sprints will each occur in two week increments. The "pretotype" sprint was completed on September 17, 2014.

Sprint 1 - Pretotype - September 17
1. User interface. 2. Vote in-app capabilities 3.
Sprint 2 - Prototype Demo - September 29
1. Demo native push notifications - isolated from web server. 2. Update and maintain notification settings. Persist to storage. 2.1. First name, last name, and address. 2.2. Select default city. 2.3. Enable/disable GPS. 2.4. Enable/disable notifications. 3. View recent polls.
Sprint 3 - Baseline Prototype - October 20
1. Social media logins 2. Share user data between app and web 2.1. user/session cookies 3. Persist login info until user logs out 4. Settings UI: User can choose to which categories to subscribe.
Sprint 4 - Alpha - November 1
1. Receive notifications by: 1.1. Category. 1.2. Location.
Sprint 5 - Beta - November 19
1. Fix bugs/TBD

Sprint 6 - Production - December 11
1. Fix bugs/TBD