# Prospective Design Document

## Team


Bill Dong: wzdong@princeton.edu

Ben Liang: bliang@princeton.edu

Michael Man: mman@princeton.edu (Project Manager)

June Ho Park: junep@princeton.edu

Yang Tu: yangt@princeton.edu


## Overview

Prospective is a website that provides a way for students to connect with eating club events on an accessible and streamlined platform. On any given night, students will be able to see which clubs are open, and what they need to get in. The system will allow eating club officers to update event info and translate it into a user-friendly display for other users to plan a prospective night out. Prospective will focus on making the data input as painless as possible to put the officers at the forefront of our concerns.

The data we need to keep the site running will come from the officers of each individual club. Once the website gets a critical mass of users, or if the data from the officers starts to trickle out, we can add a feature to crowdsource information.

## Requirements and Target Audiences

### Problem

Currently, students have no good way of knowing which eating clubs are open on a given night and eating clubs have no good way of advertising these events. There is no single place that students can go to to get information on these events; Charter Club used to have a website with information about their Friday nights, but it is no longer updated.

### Existing Solutions

The current solution is for students to pass around information by word of mouth or occasionally, flyers around campus or Facebook posts. This is slow, inefficient, and susceptible

to misinformation.  Furthermore, it alienates students that are out of the loop.  Prospective solves this by providing a centralized platform for students to view which clubs are open.

### Intended Users

Prospective is targeted towards Princeton students interested in going to the Street. Rather than having to message friends asking which clubs are open, students can pull up Prospective and instantly have access to information on all the clubs.  The other primary target of Prospective is eating club officers, and eating clubs in general (the sign-in clubs in particular). Currently, the only ways clubs have to advertise events are flyers or Facebook posts.  Instead of putting up hundreds of flyers to attract students or creating events on Facebook and inviting hundreds of students to them, officers will be able to reach a wide audience with just a few clicks of a button. Other possible users may include the Princeton Student Events Committee, which often hosts non-alcoholic alternatives at Campus Club on popular nights out. Including Campus Club event advertisements would potentially expand the user base even further.

Most of the traffic to Prospective will occur on nights on which many students go to the Street--Thursday and Saturday in particular, with very little traffic on nights like Sunday or Monday. Whether to get more information about events occuring at each club, or simply to plan out their nights, students will have the biggest need for information on the Street at these times.

# Functionality

### Core features absolutely vital to Prospective's success include:

- Attractive display to see which clubs are open and what is needed to get in.
- Quick and easy input for club officers to update club's status.
    - Verification of officers through Princeton's Central Authentication Service.
- Links and contact information for eating clubs and eating club officers.

### Auxiliary features which may be added as possible or as necessary include:

- Student reporting on eating club statuses via crowdsourcing.
    - Verification of students through Princeton's Central Authentication Service.
- Present special info for theme nights or other special events.
- Display calendar of events for next week/month to plan future nights out.
- Display information for Campus Club events.
    - Collect information through Princeton Student Events Committee.
- Display how many other students are interested in going to a particular club that night and make this information public to students, so they can see which events will be most popular on a given night

## Sample Use Cases

## Scenario 1:

A student wishes to go out tonight. The student messages their friends and none of them know which eating clubs are open. The students opens up Prospective and notices that eating club X is open tonight and students can gain entry with just their puid. The student goes and has a fun night out.

## Scenario 2:

An eating club is planning to host a big event that will require a pass to enter. With just a couple clicks, an officer publicizes the event on Prospective for every student to see. Students interested in the event notice that it is pass only and have ample time to acquire one. Many students come and the event is a huge success.

# Design

## Server

For our back-end, we will host a Django server on a private Linux server. This server will communicate with the database and handle the AJAX requests we will send from the client-side. The Django server will access the MySQL database and return information for the following requests:

*/club_info/*

This part of the Django will handle a GET request for the state of the eating clubs. It will access the *statuses* table and return an AJAX object, an associative array of club_id (standardizes between client and server), and the status: pass/list/PUID/closed/no_data.

*/events/*

This part of the server code, handling a request type GET, will return an AJAX object detailing the events going on. For each, there will be the following fields: image_url, location, description, entry_status, date. This will be used to create the events page.

*/club_post/*

This will handle a POST request connected to the administrative interface for club officers. This will hold the header fields: club, date, type (subject to change, based on what information the club officers have). This will also be connected to the CAS authentication interface, and will verify the identity of the eating club officer. Open receiving this information, the Django server will update the appropriate entries in the table *club_status*.

*/event_post/*

This will handle a POST request connected to the events posting interface. It will hold take in an uploaded image, a title, description, club location, etc. (also subject to change), and add this information to the database table *club_events*

## Database

We will use a MySQL database hosted on our Linux server. The database will have two tables, *club_status* and *club_events*. *Club_status* will have rows with keys “club_id, date, status, poster.” Meanwhile, club_events will have rows with keys “event_title, location, date, description, image and entry_status.”

## Client

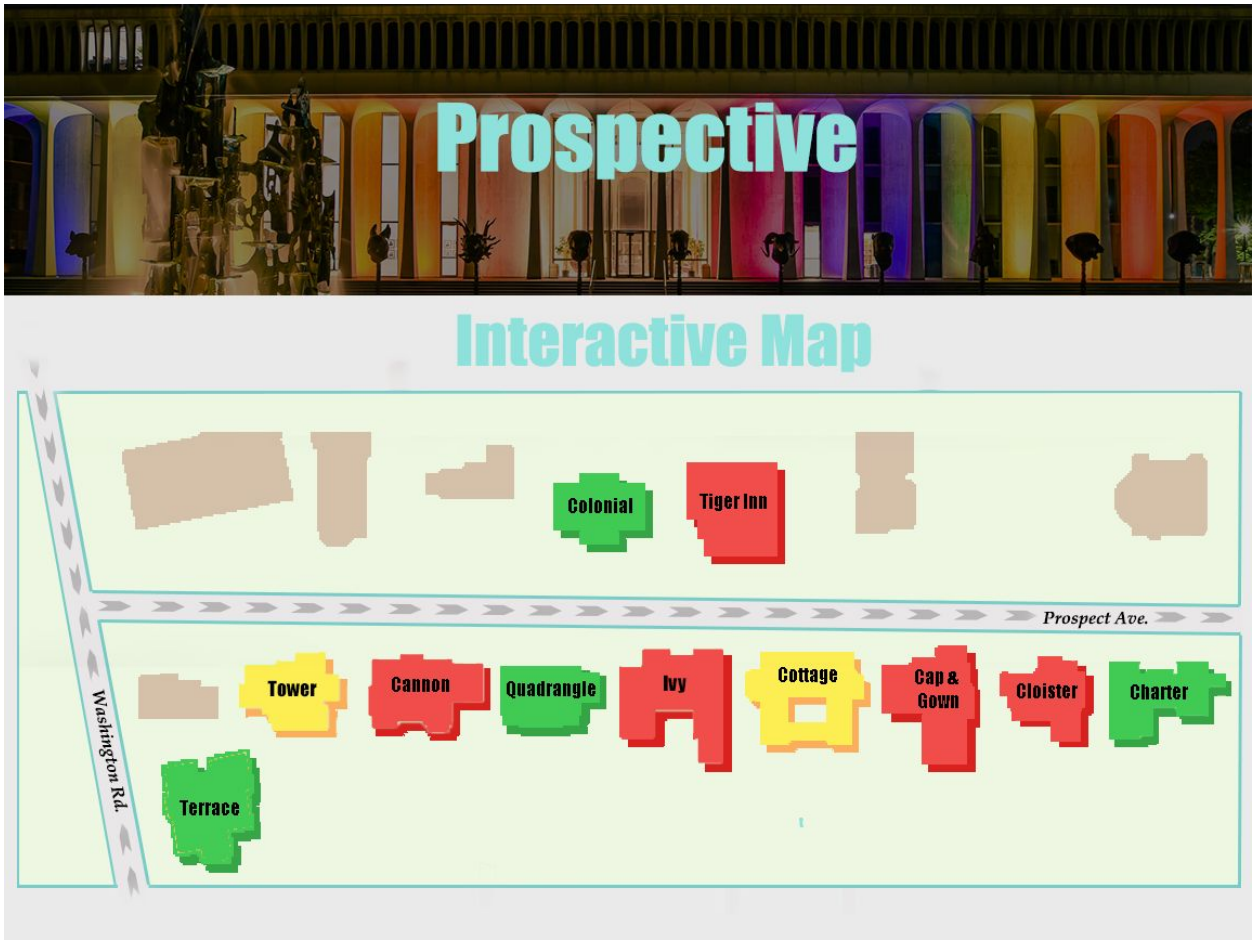
Our client will be built with HTML, CSS, and JS. The client will have a top-down view of Prospect Street, with each eating club highlighted. Depending on the status of the club, they will have the following colors:

- Green: Open, PUID
- Yellow: Open, List/Pass
- Red: Closed
- Gray: No data

Then, if a user were to hover over a club, an animated overlay will display information about the club, specifically the club name, status, and theme (if there is one defined in the events page), as well as a logo of the club that links to the eating club’s website. Furthermore, we plan on designing the website as a single-page application, and a menu bar will assists users in navigating through the following sections:

- Landing (picture of street, open/closed information)
- Events Page (Calendar?)
- About Us, Team Page

To achieve this, we plan on using Bootstrap and jQuery to achieve our goal, as well as using an HTML/JS template to provide a good starting point for the application.




# Timeline

## Week of March 18:

- Finished and submitted design document
- Decide on project name and domain
- Email eating club officers to pitch idea and ask for future collaboration
- Design front-end displays
- Launch server, domain
- Learn technologies

## Week of March 25:

- Landing page for web done
- Test with data inputted ourselves
- Begin working on officer input page

## Week of April 1:

- Officer input page done
- Demo for eating club officers
- Extra allotted time for debugging

## Week of April 8:

- All core features done for web
- Roll out prototype with working core features to friends (Due April 13th)
- Begin working on mobile (Panels)

## Week of April 15:

- Finish mobile page
- Fix problems from prototype
- Begin work on special events and calendar auxiliary features

## Week of April 22:

- Promote and advertise Prospective
- Roll out web and mobile to student population (Alpha test due 27)
- Debug mobile
- Finish special events and calendar
- Begin work on crowdsourcing if deemed necessary

## Week of April 29:

- Test, debug and fix user suggestions from alpha test
- Finish crowdsourcing if deemed necessary
- Roll out with special event and calendar if no dire bugs (Beta test due May 4)

## Week of May 6:

- Test, debug, and fix user suggestions from beta test
- Demo Days May 9th and May 10th

## May 13th:

- Final project due

## Future:

- Implement other auxiliary features

# Risks and Outcomes

By focusing first on core features and then adding auxiliary features as we progress, risk of not having a working product is reduced. We want to first get a basic version of our website running before adding new features such as descriptions for special theme nights and support for Campus Club, in order to ensure we have a working product.

A major risk is if eating club officers don't cooperate, or interest from clubs dies out. In order to account for this, data can be obtained by crowdsourcing Princeton students. Through CAS authentication, students could input information about club statuses. However, this would require a critical mass of users to be successful. Until crowdsourcing code is finished, we can manually input data ourselves to being developing a user base as quickly as possible.

A huge challenge will be designing an attractive and easy-to-use UI for both students and officers. Users will not want to waste lots of time looking for information. Front-end development could make or break the project, so lots of attention will be spent on it.

The majority of the group does not have any development experience, so there will be a significant learning curve in terms of learning the technologies that will be used for the project. Spring break was allocated as a week just for learning.

Significant traffic will most likely only be on Thursdays, Fridays, and Saturdays, leaving a limited number of days to observe real usage.