OLMSTEAD: Prospect Park Navigator
Digital Prototype: http://www.nellshawcohen.com/dmdl/olmstead

Designed by Nell Cohen
Interaction Design, Fall 2014
Ralph Vacca, New York University

Problem: Navigating Prospect Park

Before Yelp and similar services made it easy to search and review nearby business, it was harder to quickly and reliably find promising new restaurants, shops, or services to try. Friends might have recommendations: but what if all of our friends live in a different neighborhood and don't know the places nearest us? What if they have different tastes than us? Or, of course, what if we're visiting a new area and don't know anyone or anything nearby?

For those who walk, jog, or run in Brooklyn's Prospect Park for exercise, relaxation, and aesthetic pleasure, similar problems exist with regard to park navigation. Prospect Park is large and contains numerous different crisscrossing paths in varied scenery. Aimless exploration often leads to circuitous routes, dead ends, less-than-interesting locations (a parking lot for maintenance vehicles; a high-traffic biking road; etc), or simply getting lost.

The conventional brochures and poster maps provided by the city and the Prospect Park Alliance, while helping with the "lost" part, offer little information about the on-the-ground experience. These sources indicate major landmarks, but not the kinds of scenic features or experiential characteristics that are of primary concern to walkers. Additionally, directions apps such as Google Maps only utilize the major paved paths and are similarly bereft of qualitative information, taking only efficiency into consideration when calculating routes.

As a result, people who walk in Prospect Park may be more likely to stick to the few pathways and areas that they've become familiar with. They miss out on many of the rewarding aesthetic, recreational experiences that they want to have in this special city park.

Solution: A Mobile App

I've designed OLMSTEAD (named after the park's 19th century design, Frederick Law Olmstead), a mobile phone app for people who frequently take walks in Prospect Park and want to engage in guided exploration. The app enables users to discover new routes in the park based on the aesthetic characteristics and atmosphere that they personally enjoy. It then guides them through navigation along the route. All of the data for routing and navigating in the app is based on a crowdsourcing model, because walkers know best what kind of information about the park will be most useful and meaningful for others walkers. During navigation, walkers will be shown tips and photos from other users about interesting nearby features in the park. They will also contribute observations and share their experiences in the park with other users and with their friends on social media.

Design Description

Functionality

The interactions within the app may be grouped into the following categories:

1. Finding, saving, and/or beginning a route based on start/end locations, time or distance, scenery, walking surfaces, etc. A search function enables the user to specify all of these and then select from a list of matches, ordered by similarity to the user's criteria. Once they have found a route(s) of interest, the user can save the route for later or begin navigating right away. They can also share interesting routes with a friend via Facebook, Twitter, or email (say, if you're planning a walk for the weekend with a friend).

2. Navigating along the route. Users check their phone for a real-time updates on relevant navigation directions (a combination of symbolic/verbal directions and map detail views) based on their GPS.

3. Discovering new features of the park, guided by GPS-tagged user tips (tags, photos, and comments) in close proximity to their current location.

4. Contributing to the crowd-sourced database of tags, photos, and comments.

5. Self-monitoring and tracking experiences in the park with "Walk History," a mapping of the user's walking history in the park while using OLMSTEAD (based on GPS tracking), as well as "Recent Activity," a timeline of the user's contributions (tags, photos, comments, and shares).

Walkthrough

On the home screen [**Fig. 1**], the user is invited to find a route, view saved routes, walk history, or recent activity. "Find Route" leads the user to a search screen, on which they select relevant criteria and then search **[Fig. 2].** "Search Results" are shown **[Fig. 3]** with all possible matching routes (those which are meaningfully distinct from each other) that match the user's criteria with at least 75% accuracy. A match with 90% could indicate that, for example, a walk is 3 miles instead of the user's requested 3.5 miles: however, the route still features prominently all of the scenic characteristics they expressed interest in. Alternatively, perhaps the distance of the walk perfectly meets the user's specifications, but wooded areas (which the user requested) don't feature prominently along that route.

If the user selects a route from this list, they will be provided with relevant details to give them a sense of what their walk will be like **[Fig. 4]**, including a list of notable landmarks and a visual representation of the route on a park map. Additionally, a tag cloud will show the most prevalent tags along the route—this will give the user an immediate sense of what kinds of scenery they might encounter. Finally, if photos have been uploaded along the route, a set of clickable photo thumbnails will also be displayed.

From the Route Details screen, a user may save and name the route (which places it into "Saved Routes"); share it through their linked social media accounts (Facebook, Twitter, email); or begin navigation directly.

During Navigation **[Fig. 5]**, users will be shown the upcoming relevant direction; a map detail view with current GPS position; and buttons for editing the route and adding tags, photographs, and comments. Below that, the user is shown a tag cloud; list of comments; and photos contributed by other users, which have been uploaded and tagged with GPS coordinates within 100 feet of the user's current location. User contributions may also be "upvoted," "downvoted," or "flagged" for moderation. Visual characteristics, such as font size and opacity, denote the relative proximity of comments and tags to the user's position.

Not all areas of functionality are implemented in my digital prototype, particularly those related to contributing during navigation. I have indicated functionality on the Navigation screen with representative icons. While I am not yet specifying the details of how these would work, it should be noted that comments are intended to be very brief and should have a character limit. Additionally, the tagging function may incorporate autocomplete suggestions of common tags (trees, waterfall, etc) in order to ward off duplicates due to misspellings and to save time. Finally, clicking "Edit Route" during navigation would lead the user to the same search screen they used to locate the route, with "Current Location" updated to reflect their current GPS position.

Techniques and Tools

A range of design techniques was used to develop this project. At points throughout the process, I drew concept maps and/or flow charts to determine which screens would be present in the app, which would be prototyped, and how the overall design flow would work. I also facilitated a group brainstorm with my colleagues to determine desirable features and functionalities. I also drew sketches to brainstorm possible interfaces, scenarios of use, and features. These two processes left me with a variety of divergent concepts, the majority of which I wasn't able to implement in the final design, but generated several interesting ideas that did end up in the app.

Storyboards (drawn in Adobe Illustrator), scenarios, and a persona further clarified the focus of intended app use for my final design. I then conducted a large-scale design synthesis by organizing my final list of interactions into functional categories using post-it notes. Out of this, I as able to identify the "Find Route" function as the crux of the design and the focus of my subsequent prototyping.

Next, I wrote up a formal QOC to clarify individual design features for my paper prototype of the "Find Route" function. I also did more informal QOCs with regards to the "Navigation" function. I created a paper prototype and conducted a "Wizard of Oz"-style user test of the "Find Route" functionality.

Each screen for the paper prototype was designed in Adobe Illustrator and then printed, cut, and taped. If I were doing this project over again, I would have chosen not to use Illustrator to design the screens for the paper prototype. I found that the fidelity of the paper prototype screens were perhaps greater than necessary to derive the relevant information from the user testing.

Based on the outcome of the paper prototype testing, I made rough sketches of a revised user interface. I found that conclusions from my QOC for the "Find Route" function had led me to produce a design in my paper prototype that proved somewhat cumbersome during testing—using a "Wizard" to guide the user through the route refinement process **[Fig. 6]**—and I ended up reverting to my pre-QOC concept of utilizing checkboxes, radio buttons, and toggles for route search **[Fig. 2]**.

Finally, I produced a digital prototype in HTML/CSS using the jQuery Mobile library. Additional icons were sourced from Font Awesome and The Noun Project. "Dummy" images came from the Prospect Park Alliance and Flickr.

I began building my digital prototype in FluidUI, but found that the free version of this software was fairly inflexible and lacked all of the ready-made interface elements (particularly icons and buttons) that I had hoped to utilize. Additionally, the fidelity of the GUI (i.e., accuracy to what a mobile app conventionally looks and feels like) left much to be desired on both iPhone and Android devices that I tested.

jQuery Mobile was relatively time consuming to use, since I had to hand-code each screen individually, as opposed to the "drag-and-drown" graphical interface of FluidUI. Learning jQuery Mobile also required frequent reference of the documentation (although the documentation was very well organized). However, this approach gave me a much more polished product resulting from overall less effort and requiring fewer adjustments: it looked *right*, right away. I believe it was ultimately more efficient than the alternative solutions for the needs of this project.

Design Evolution

Between the initial problem statement and the final digital prototype, several shifts in focus took place. For example, in my initial design approach write-up and sketches I had imagined that the app would facilitate the uploading of music, visual art, etc, and place these around the park in virtual space as "easter eggs" that would be "unlocked" based on the user's real-world GPS positioning. As I moved forward, I removed users' ability to share creative work and limited user contributions to tags, comments (brief "tips"), and photographs. While I still think the aforementioned creative sharing would be extremely interesting to incorporate into a different project, I realized that it was beyond the scope of functionality needed for OLMSTEAD. It would have added a degree of complexity that may have detracted from the core functions of the app: what Lidwell refers to in "Universal Principles of Design" (2003) as the *flexibility-usability tradeoff*.

Additionally, I had previously envisioned several different ideas for a graphical "environment" for navigation—e.g., an augmented reality view of the park environment (seen through phone camera) with user tags and comments superimposed over the camera view. However, for this utilitarian app that is intended to be used on-the-go, I realized that I would need to keep the navigation interface simple, optimizing for efficient delivery of only the most essential information: i.e., "Which way do I walk next?"

During the peer group brainstorm, we considered audio notifications, a la Google Maps navigation, to guide the user. Audio notifications would have the advantage of being "hands free," enabling the user to listen to music while they are walking or running and being notified aurally when they need to turn down a given path.

I considered pros and cons for the audio notifications approach and decided that it would be too problematic. This function would involve push notifications, which require constant GPS tracking and eat up phone battery. Additionally, it wouldn't be effective if the user weren't using headphones. Finally, audio notifications can become intrusive and irritating: e.g., the walker may want the option of wandering off the route without being constantly rerouted and notified. I decided that the user would need to actively check their phone when they wanted directions: while not absolutely ideal, it places control in the user's hands to decide when and if they want assistance.

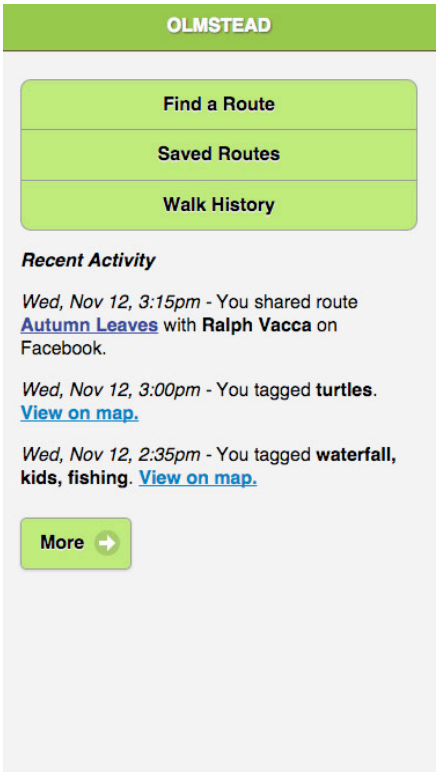

Fig. 1, Home

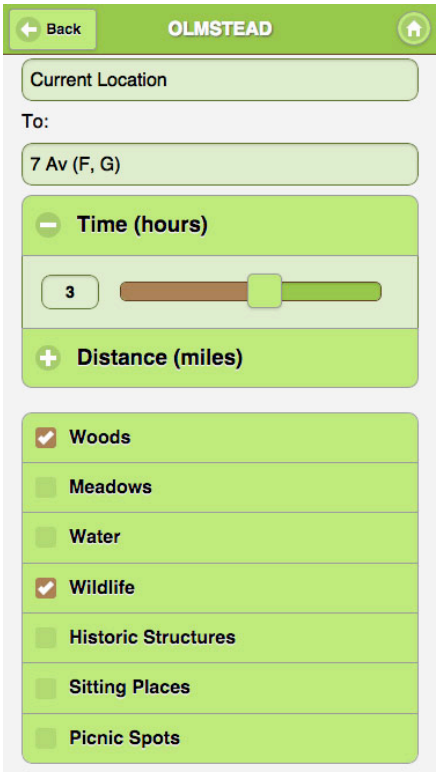

Fig. 2, Find Route

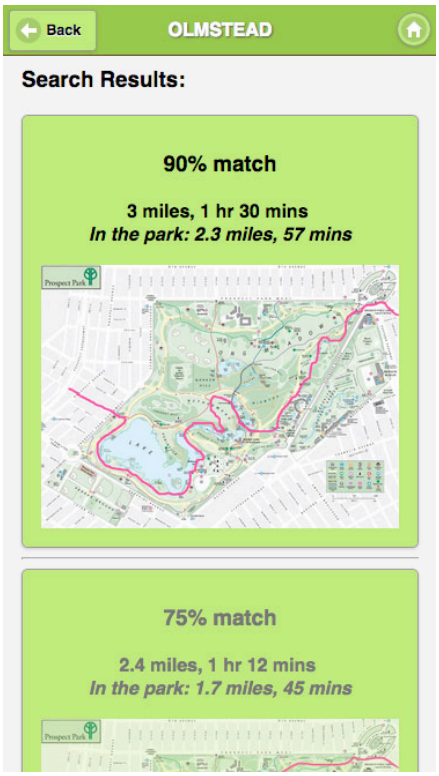

Fig. 3, Search Results

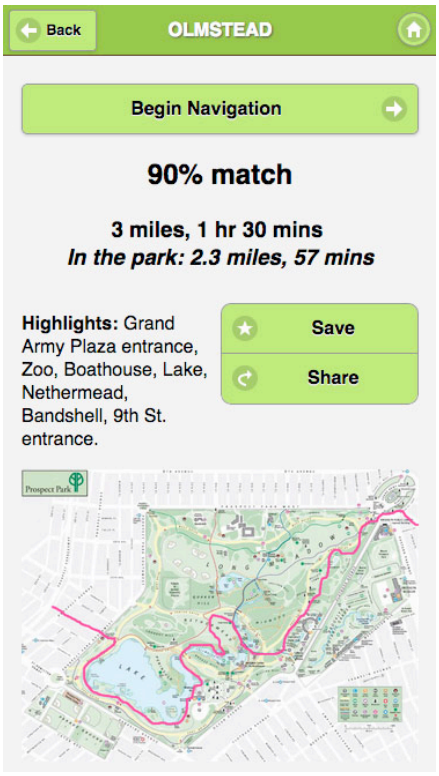

Fig. 4, Route Details

Fig. 5, Navigation

Fig. 6, Find Route "Wizard" in paper prototype