Andrew Smith <andrewrs@cs.washington.edu, andrewrs@uw.edu >
Donovan Hunt <dhunt925@cs.washington.edu, dhunt925@uw.edu>
March 30, 2011
CSE 403 Written Proposal

# Activity Finder

## Vision

Imagine you wake up on a Saturday morning and have no plans. You know there are a lot of activities out there, but you don't want to wade through hundreds of search results to find something. Enter Activity Finder. This application gives you suggestions based on your budget, travel, and time constraints. It takes into account where you are, what you enjoy, what the current weather conditions are, and more to make it the best experience possible. This mobile app fights boredom and gives users a planning tool to make the most out of their time.

Our goals in making this application are creating something quick, easy-to-use, and insightful. If Activity Finder generates useful, fun activities, users will be hooked and will share the app with friends too. The application's target audience consists of smartphone users, 16-35, probably with a higher male distribution. We see this as a tool for finding individual activities, planning a date for two, or even discovering an adventure for a group of seven friends. By making this app mobile, a user can get this information on the go. We have considered current alternatives to be Yelp and Google Maps. However, both are capable of giving location-specific, rated information, and distilling one choice from either can be daunting. Activity Finder is meant to give suggestions, much like Netflix suggests movies or StumbleUpon directs one to interesting websites.

## Software Architecture

For this project, we will be using the Software Development Kits (SDKs) for both the iPhone and the Android platforms. We will make use of the Global Positioning System (GPS) on the phones to locate the user and find activities closest to him or her. We will create a front end on the phones that communicates with a server through the Hypertext Transfer Protocol. This server will use Web Services to process the requests from the phone and return the data requested by accessing the database. Requests are processed on the server-side before being sent back to the client application in a simple format that can be rendered on the phone in a rich user experience. The specific framework for implementing the Web Service will be decided based on the project team's experience with different technologies.

We plan on having three groups that focus on particular programming elements of the project. The first will work on the iPhone SDK, the second on the Android SDK, and the last will focus on the server back end. During the planning stage, there will be a unified direction on how a user will interact with the client apps, through paper prototyping. Each phone group will then work from the design on a planned schedule, adding features together. This way, members in the group can focus on developing their skills in one platform and become experts, but the project as a whole can be developed together and features can be added gradually.

The interesting part of this architecture is there will be communication between three drastically

different platforms.  Designing software that works on both an Android phone and an iPhone with a similar experience offers an interesting challenge in the planning process. We also are developing a Web Service that has potential to be accessed by other clients such as a website, meaning this product the potential to be plugged into many technical areas.

## Challenges

We currently foresee the major challenge in this project to be a learning curve with the phone SDKs or server-side technologies.  In order to minimize the risk of any group falling behind the others, we will have groups split up from the beginning as we plan the project and the features it will have. These groups can try to implement very basic versions of each feature in the form of simple code sketches and mockups.  This way, when it comes time to actually write the code, the groups will have experience with each feature and won't spend a lot of time looking up how to, say, get the GPS information for the iPhone.  The core pieces that we need with this application are UI, GPS, web connectivity, database interaction, and potentially weather information. Each of these can be explored on their own before integrating them in the real project. This approach allows the team to plan how long it may take to implement each feature and then allocate more time for parts that take longer.

## Mockup of Application

This shows an example screens of searching for something to do (left), a resulting list of suggested activities (center), and a description of a single activity.