# BOILER LABS
# Design Document

**Team 5**
Team Members:
Rishabh Ahluwalia
Chen Gong
Akshit Gupta
Abhinav Kumar
Shalin Shah
Ayan Singh

## Purpose

The purpose of this design document is to provide key details of Boiler Labs. The document will include: a summary of non-functional requirements, a design outline, design issues and design details.

The goal of Boiler Labs is to help users locate an ITaP Computer Lab (with available computers) that is nearest to the user and provide directions to that lab. It is also to provide the user with a list of all the ITaP Computer Labs on campus and display the details of a particular lab.

## Non-Functional requirements

1) **Performance**: The system as a whole should be able to perform well and attempt to mask slowdowns. For instance, if the internet speeds between the server and the client become slow, the client should not become unresponsive. It should still be interactive while data is synced asynchronously. As much of the processing load will be given to the web server as possible, since we do not know the exact hardware of the client. This will enable us to help keep performance as consistent as possible across clients.

2) **Reliability:** Reliability will be a main concern, mainly because if any portion of the system fails, the system as a whole becomes next to useless. While performance is a main concern as well, if the system is glitchy and unreliable, performance does not matter. We will attempt to balance these two requirements in a way that will maintain a reliable system while achieving adequate performance.

3) **Usability:**

The Android app will conform to the modern Google app development guidelines. We will conduct hallway testing and ask people without any programming experience to use our application, and evaluate their feedback in order to improve usability.

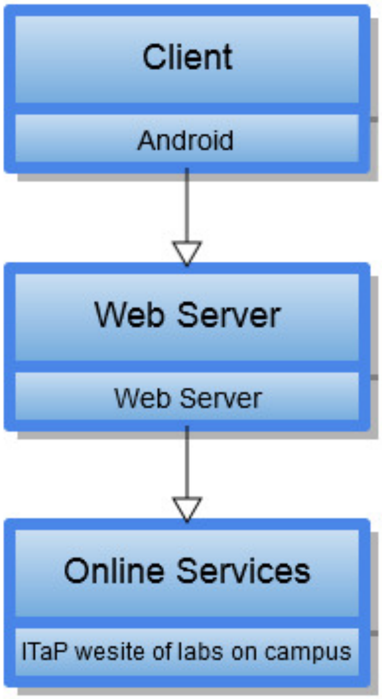
4) **Security:**

Securing user data will be a primary concern. No personal data (such as login information) will be sent as plaintext over the internet. Also, we will ask the user to give permission (which will be persistent to enhance usability) before retrieving data.

## Design Outline

The full system will be composed of 3 layers, 2 of which will be our responsibility to develop.

The following diagram will help to illustrate these layers**.**



The first layer, which is the layer that we will not be developing, is the third party layer, which consists of ITaP website of labs on campus. This will be used to provide lab data (name, type, location, status, and computer).

The second layer is the web server. We will be using javascript on a Tomcat server to continuously monitor activity of each lab on campus and calculate peak hours of usage. The server will also connect to the ITaP server and get all the data from there.

The third layer is the client which will be the application running on Android. This will connect to both, the Web Server and the ITaP server and grab the relevant data from each.

**Description of the interactions**.
i)**Interaction between Client and Online Services**: The Client will receive the required data from the online services whenever necessary.
ii)**Interaction between Web Server and Online Services**: The web server receives data from the online services ( primarily to determine which lab is the busiest lab).
iii)**Interaction between Web Server and Client**: The client can both send a query and receive and display the relevant data.

## Design Issues

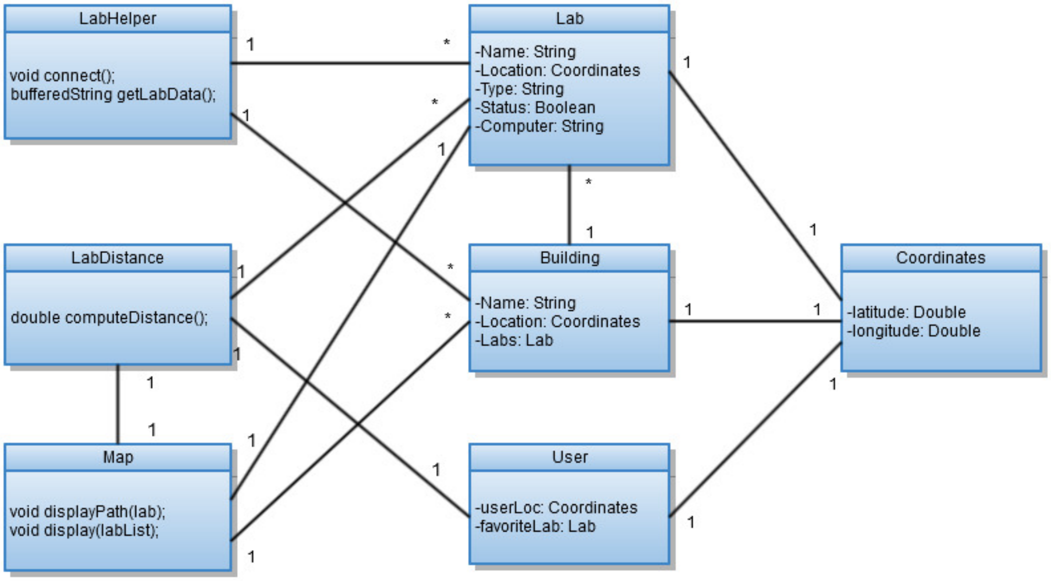
**Programming Language**
**Option 1:** Java
**Option 2:** Objective-C
**Decision: Java**
**Reason**: First of all, all of us are quite familiar with Java. Android Software Developer Toolkit provided by Google in Eclipse is widely considered as the best way to implement an Android application. Also, there is an official iOS application titled “Purdue” that performs a similar task.

**Version Control**
**Option 1**: GitHub
**Option 2**: Bitbucket
**Option 3**: SVN
**Decision: Github**
**Reason**: We made this decision based on the fact that all teammates are all familiar with GitHub and it’s a distributed version control system. Though Bitbucket is similar to Github, we choose GitHub because we all have GitHub accounts. Centralized version control systems such as TortoiseSVN are easy to use but hard to manage a server and backups.GitHub can help us merge codes, synchronize our codes and trace the changes in an efficient manner, therefore our decision is GitHub.

**Users Location**
**Option 1:** Google Maps
**Option 2:** Bing Maps
**Decision: Google Maps**
**Reason**: On Android phone, our first choice can only be Google Maps as we are using Google Maps on the phone. Also, Google Map is the most frequently used online map all over the world.

# Design Details

## Class Interactions



### LabHelper

This class is responsible for connecting the client to the web server and the ITaP server. This class will use JSON to query the data from the servers and format it. It will then initialize and populate the relevant Lab objects and Building objects. This class will be connected to the Lab class and the Building class as it is responsible for creating and populating them. There will be a 1-To Many connection with the Lab class as there will be only one instance of LabHelper and many instances of the Lab class. Similarly, it will have a 1-To Many connection with the Building class.

### Lab

Lab represents a single ITaP computer lab. The entity will store name, location, type, status, and computer of a particular ITaP computer lab. Also, it will use the Coordinates class to store the location.

### LabDistance

LabDistance calculates the distance between a particular ITaP computer lab and the User.

### User

User represents the person using the application on Android. The entity will store the location and preferences (favorite lab) of a particular user.

### Map

This class displays the map of Purdue campus and will provide the user with a path to an ITaP computer lab that is nearest to their current position. It will also display all the ITaP labs on campus

### Building

This class represents each of the buildings on campus with ITaP labs. It will also consist the location and number of labs in the building.

### Coordinates

This class is used to store the longitude and latitude for each building and lab.