LineUp

Team Aphrodite
Nora Allison (efa91)
Bryan Djunaedi (dju90)
Stephen LaPlante (laplansk)
Evan Leon (ejl6)
Thomas Rothschilds (tgr4)
Simone Schaffer (simone09)
Nicholas Shahan (nshahan)
Evan Whitfield (evanw2)

LineUp Developer Instructions

Joining the development team for LineUp is easy. These instructions tell you what you need to know to get started.

Software Install Instructions

To get the latest version to start developing, use:

`wget` https://raw.githubusercontent.com/Line-Up-Admin/CSE-403-Spring2014/master/prodSetup/install.sh

This will install the necessary software tools, and will clone the most recent revision into a 'lineup' directory in the current directory. For development, you should use git to manage the repository, as the install script is meant to run once.

Now that you have all the code and tools, you can develop using whichever editor you like and can run a Simple HTTP Server locally to see results within a browser by running the python script run.py and going to localhost:5000 in any internet browser:

        python run.py

Directory Structure Layout

The directory structure in the repository is set up to make development and deployment extremely easy; All the files/folders are positioned exactly as needed for running the system.

```
/                       #working copy root
/app                    #python code
/app/static             #static assets
/app/static/bootstrap   #standard bootstrap library
/app/static/css         #css for our site visuals and layout
/app/static/img         #logos, icons, etc.
/app/static/js          #angular application, controllers, directives, etc
/app/static/partials    #site pages
/prototype              #scratchspace
/tests                  #all tests
/tests/angular_tests    #tests for angular framework
/tests/flask_tests      #tests for flask framework
```

Build Instructions

Because Python is interpreted, there is no build step. You can use run.py in the 'lineup' directory to run the app locally, but you will need Python, Flask, and SQLite installed. To test code and run the server locally, developers will be running a python simple HTTP server.

It should be noted that the simple HTTP server will not reflect the actual production environment which will be hosted by Apache. We chose this environment because of the ease of use and setup (practically none) compared to setting up an apache server on every development machine. Since our

implementation currently does not include any advanced hosting features the difference between the web servers should be minimal. Still, development time should be allotted for deploying the product to make sure the web server hosting is correct.

Testing Instructions

All the tools necessary for testing on an individual basis are already included in the repository you have checked out. There are scripts written that run tests over the entire system, but likely, you will only want to run tests over the part of the system you are working on:

Client side Testing
Navigate to /tests/angular_tests and open the `SpecRunner.html` file in a web browser.

Server side Testing
Navigate to /tests/python_tests and run:
```
python *test*.py
```

Automated Build & Test

Our team uses Travis-ci (travis-ci.org)to run our test code and email the results to the entire team every time a commit is made to the repository. This should all already be set up with your repo checkout, but if you would like to receive these test result emails, you will need to make a change to the travis-ci configuration file. To do this, open the .travis.yml file in a text editor and add your email to the indicated region below:

notifications:
  email:
   recipients:
    - cse403group@uw.edu
    - [your email here]

New Releases

New releases will be represented in our github repository as a branch. To download a specific version you will need to clone that branch. The version number of the release is listed in the README.md file in the repositories root directory. If you are an admin of the repository and are ready to make a new release, do the following:

1. Check in all code to be included
2. Update the version number in README.md
3. Create a new branch with the version name

Issue List

       Should you find a bug in LineUp, you can submit a description of the issue in our Issue Tracker, [here](). We recommend that you follow the guidelines of a standard [bug report format](), and please review our [bug list]() to keep from submitting an already known bug. For instructions on how to do so, please refer to [this similar documentation]() for another GitHub project.