

## Scrum 1: Planning Meeting – 10<sup>th</sup> March 2017

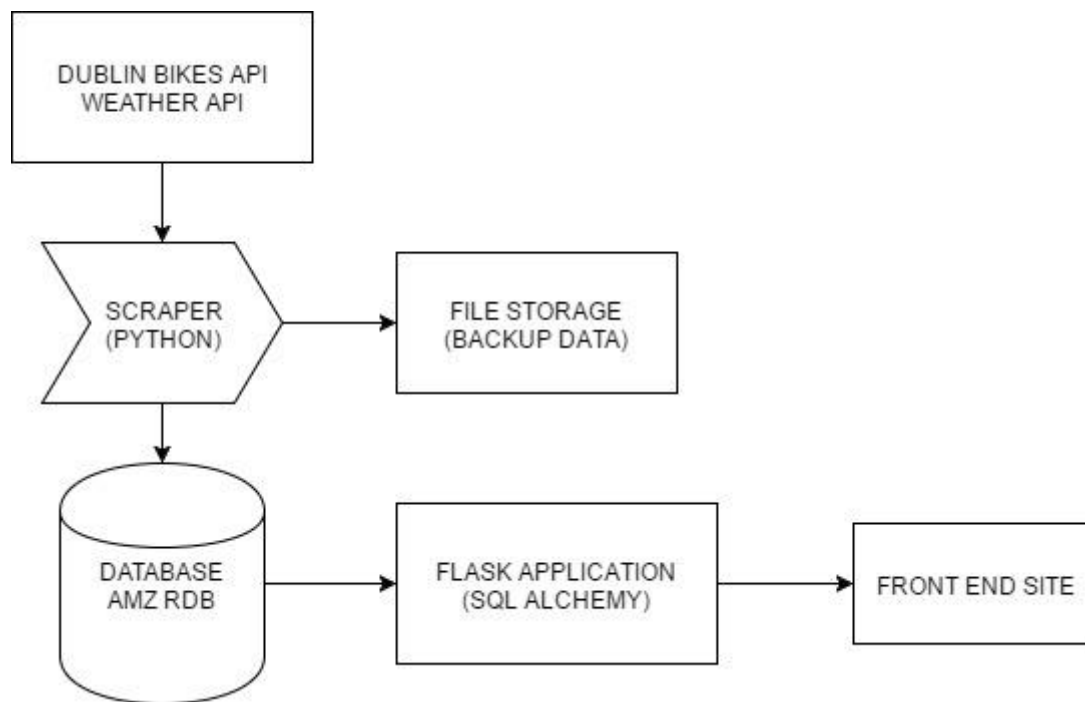
Attended by:

Customer: Devin Stacey

Team: Emma Byrne, Katherine Campbell, Pamela Kelly

### Plan for Scrum 1:

Visual Overview:



### Tasks for Sprint 1:

- Familiarise ourselves with the Dublin Bikes API, and as a secondary priority the Weather API.
- Formalise a structure for how the data scraped from the API will be stored in the database.
- Familiarise ourselves with the other technologies pertinent to the project: Amazon RDB, Amazon Hosting, Flask, SQLAlchemy.
- Sign up for a Trello account and begin organising the project tasks through that. Look at how to convert to burndown charts.
- Plan standups (may vary over the break in terms of what days suit – the team will decide on regular days prior to when classes recommence).
- Set up scraper to scrape information from the APIs. (Priority: High 1)
- Set up database so that information can be stored immediately once scraper is ready. (Priority High 1)
- Create basic html/css front end site. (Priority: Medium 1)
- Set up Flask App to process queries to the database from the front end site. (Priority: Medium 2)

## Standup Meeting: 14th March 2017:

After our initial research, we confirmed that the highest priority tasks are the scraper, the database and the front end site. We divided these tasks up as reflected below. We decided that for the database we are going to use Amazon RDS and MySQL, because we are all already familiar with MySQL. We have moved the Flask application task to the backlog for now. Choosing to focus on the three above listed tasks as high priority for this Sprint.

### Katherine:

- Looked at project organisation tools, i.e. Trello.
- Started researching Flask, looked at Flask tutorials.
- Looking at ways to develop the database from a user point of view.
- Next step: Work on database setup with Amazon RDS and MySQL.
- Not Blocked.

### Pamela:

- Looked at Dublin bikes API.
- Defined a possible structure for the database based on data types given by API.
- Started a team Trello board and organised structure.
- Next step: Work on setting up the scraper in python.
- Not Blocked.

### Emma:

- Looked at Dublin bikes API.
- Looked at scraping the API and using javascript to access information.
- Researched Amazon S3 and RDS.
- Next step: Work on the front end set up in HTML and CSS.
- Not Blocked.

## Standup Meeting: 17th March 2017:

### Pamela:

- Started work on the scraper. Signed up for a Dublin Bikes API key.
- Read through documentation and lecture notes on API requests.
- Wrote a basic script for sending requests.
- Next step: continue work on the scraper. Store information pulled from the API in files. For now store them in files until the database is up and running.
- Not Blocked.

### Emma:

- Started working on the front end HTML and CSS. Signed up for a Google API key for use of maps and markers.
- Wrote a basic HTML webpage which includes the Google map and 1 marker set at a specific location.
- Next Step: Set up markers for all the locations of the bike stations from some of the data that we have scraped already. Redesign colour scheme of webpage.
- Note Blocked.

Katherine:

- Familiarized myself with and have thoroughly read through the documentation for creating a MySQL instance and connecting to a database on a MySQL DB instance via Amazon Web Services.
- Read through Amazon VPC documentation.
- Next Step: Set up the database.
- Not Blocked.

Standup Meeting: 21st March 2017:

Pamela:

- The scraper script is set up to send queries every 5 minutes and store the results in files.
- Got sidetracked by some git and eclipse errors which needed to be resolved before moving forward.
- Next Step: Get the scraper running on EC2 continuously.
- Blocked: Have to wait for the database to be set up so that when the scraper is running on the EC2 it has somewhere to send the data to, other than storing it in the backup files.

Emma:

- Continued working on the basic HTML and CSS.
- The map and first marker marker are set up.
- Next Step: Set up markers for the rest of the station locations.
- Blocked: Waiting for data from the API in order to set up the remaining markers.

Katherine:

- Read over the Amazon RDS documentation. Decided the best route will be to go with a MYSQL database instance.
- Next Step: Start construction of the database.
- Blocked: Need to decide on an agreed password and username for the database so that we can all access it. Having some issues understanding Amazon VPC which is needed to move forward.

Issues Resolved:

- Used the data gathered from the scraper to send the rest of the station information on to Emma. Block removed.
- Decided on username and password for Amazon RDS.