

Project – Angular

[Start Assignment](#)

Due Dec 2 by 11:59pm **Points** 34 **Submitting** a file upload
Available after Nov 2 at 12am

Due: Dec 2 midnight (hard deadline)

Description:

Farmer Hoggett has lost all his pigs! 🤔

He forgot to lock his gate and all his pigs (that are stored in the assignment 3 application), have been lost; he needs your help again. Farmer Hoggett would like to create a web application that will allow the public to help catch and report his missing pigs. Fortunately, each of his pigs has a farm *logo* <H>, and a *Pid* (pig identification number) humanely branded on its left butt-cheek. When spotted, a helpful member of the public shall hog-tie the pig (see below), then create a report on the app with their name and information for a later reward (a year's supply of pork chops; synthetic, of course). Farmer Hoggett can then safely retrieve his pig and confirm the reward.



Each *Pig Report* will have **at least** the following attributes (you may add others as you see fit):

- **Reporting person's info:** a *name* + *phone number*
- **Pig info:** the *breed* (i.e., "Pot-bellied" or "unknown"), the *Pid* (i.e., 101). You may allow multiple reports with the same *Pid*; it will be the person who first reports it who will claim the prize.

- **Location:** contains a unique location *name*, *longitude*, and *latitude* of where the pig was caught. This is where Farmer Hoggett should go pick up the pig.
- **Extra notes:** contains some *extra information* to farmer Hoggett about where to find the pig (i.e., “under the drinking fountain outside B9200”)
- **Time/Date:** Contains the *date* and *time* of the created report.
- **Status:** when a report is first created, the *status* will be set to **READY FOR PICKUP**. Once farmer Hoggett retrieves the pig, he can set it to **RETRIEVED** (please see below).

User Requirements:

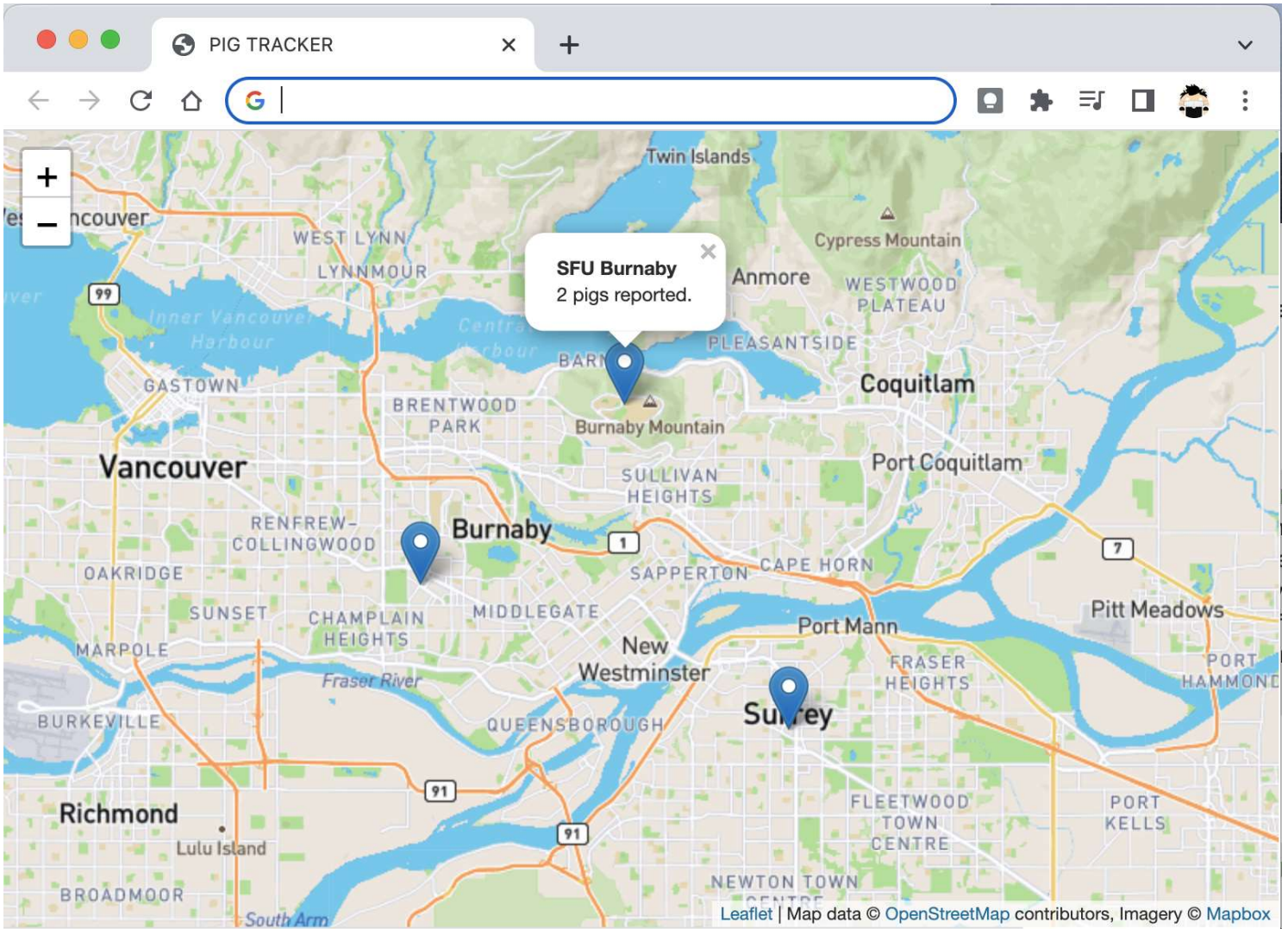
Data Table

- A data table of all pig reports should be displayed on the application's main page. The user shall be able to sort the data by each column. For example, in the figure below, the user shall be able to sort the reports by Location, Reported by, or Time Reported. **(3 marks)**
- Buttons or links shall be provided to change the status and delete individual reports. If the delete or the change status button is clicked, the app will ask for confirmation along with a prompt for a password; the valid password is “OINK!!” (all-caps). An MD5 hashed version of the password (“84892b91ef3bf9d216bbc6e88d74a77c”) can be hardcoded in your application (intuitively, not the best solution). If an invalid password is entered, the operation is disallowed. **(9 marks)**
- Links shall be provided for each pig report to reveal the full information about the report. This can be a pop-up modal or a new page. **(4 marks)**
- The user shall have the ability to create a new pig report by clicking on the CREATE PIG REPORT button. When (and only when) the button is clicked, a pig report creation form shall be shown to the user. **(2 marks)**
- To add a new pig report, the user shall select from a dropdown menu of existing places in the application or to add a new location. **(4 marks)**

Map

- A map indicating all pig reports shall be shown using markers. **(4 marks)**
- Clicking on an individual markers on the map shall show the number of reports at that location. **(2 marks)**

Here’s a possible design but you are free to be more creative.



Location	Reported By	Time Reported	Status		
Metrotown	Steve	2022-11-01 (5:30pm)	RETRIEVED	MORE INFO	✖
SFU Burnaby	Jane	2022-10-30 (1:34pm)	READY FOR PICKUP	MORE INFO	✖
SFU Burnaby	John	2022-10-22 (5:30am)	READY FOR PICKUP	MORE INFO	✖
SFU Surrey	Steve	2022-11-01 (6:30pm)	READY FOR PICKUP	MORE INFO	✖

CREATE PIG REPORT

Technical Requirements

- The exact designs and workflows are totally up to you (but please be mindful of usability and creativity). For this assignment, you are free to use libraries such as Bootstrap to help with the design process
- You must use Angular as the framework for your application.
- Your application should give meaningful feedback on incorrect inputs or error situations. **(3 marks)**
- Your application must have proper creativity and usability (i.e. it should be intuitive to use and learn) **(3 marks)**
- You must use the data manager API to save your data. The details will be sent to you via email in week 10. You may not use any other database.
- You must use the Leaflet maps API to display the pig reports (a demo will be provided in class). You may not use any other maps API.
- API references:
 - Hash algorithms API: <https://hashify.net/> ↗ (<https://hashify.net/>)
 - Leaflet: <https://leafletjs.com/reference.html> ↗ (<https://leafletjs.com/reference.html>)

Marking Scheme

- Proper implementation of each of the user requirements (28 marks)
- Creativity and Workflow in technical requirements (6 marks)

Submission

1. Delete the contents of your **node_modules** folder
2. Zip all files under the root folder of your application
3. Upload the resulting zip file

We will first unzip your file, then run

```
$ npm install
```

which brings back the **node_modules** folder (you should check that this works :), then run

```
$ ng serve -o
```

in addition to checking your code.