

(Practical Problem-Solving Project)  
**CyclistsHub Web App**  
<https://cc-cw2.appspot.com/>

Student ID	s3712755
Student Name	Moafaq Ashshareef
Course Code	COSC2626/2640
Course Title	Cloud Computing
Submission Date	20 Oct 2019

## Table of Contents

<b><i>Summary.....</i></b>	<b><i>3</i></b>
<b><i>Introduction .....</i></b>	<b><i>3</i></b>
<b><i>Related Work.....</i></b>	<b><i>4</i></b>
<b><i>Software Design / Architecture .....</i></b>	<b><i>5</i></b>
<b><i>Implementation .....</i></b>	<b><i>6</i></b>
<b><i>User Manual .....</i></b>	<b><i>7</i></b>
<b><i>Conclusion .....</i></b>	<b><i>7</i></b>
<b><i>Appendix – Screenshots .....</i></b>	<b><i>8</i></b>
<b><i>References.....</i></b>	<b><i>15</i></b>

## Summary

This is a web application for cyclists in Melbourne. As a cyclist myself, I'm aware of cyclists' needs for an enjoyable ride. The objective of this web app is to gather information available in different sources in one platform making it as the entry point website, cyclists would visit usually and before going for a ride. Such a platform requires some cloud services and techniques to be used.

## Introduction

### **What are the motivations behind your idea?**

I'm always keen to provide something to cyclists as I'm one of them. The idea based on my need to have everything that benefit local cyclists in one place. I personally struggle when I first came to Melbourne to look on Google for bunch rides and races near me. The main initial idea is focusing on local cyclists by providing them with an online platform. It's like the social media of cyclists but locally as it requires going out in reality. Cyclists can for example explore events nearby or meetings or get involved in a new bike launching or join in a bunch ride.

### **What it does?**

Melbourne weather is always changing. personally, whenever I go out to cycle, I check the weather for wind speed, chance of rain and temperature to be prepared accordingly. This web app takes those values and display in at the top of the page with a statement to say if it's too windy to cycle when the wind speed exceeds 30km/h. Also, the web app provides the following features:

- Map with markers of events or bunch rides. User can click to navigate to the relevant post (posts are added in the admin dashboard and are retrieved from the database)
- Map with markers of nearby bike shop according to the user current location
- Map with markers of Melbourne bike share network with status of available and occupied docks
- Map with drinking fountains in Melbourne, useful for cyclists to refuel

### **Why it is required?**

Cloud services from Firebase are required for authentication to grant access to admin dashboard and to modify the database which is also created in the cloud with Firebase Cloud Firestore. Google App engine was used for deploying and hosting of the web app which was built with HTML and JS. At this stage, there is no sign-up for public users.

### **How it can be used as real-life application?**

In a real-life scenario, this application can be used on the go as the cyclist head to their next ride around Melbourne. They can locate drinking fountains in case they run out of fuel “i.e water”. Find a bike shop with a press of a button and let the Google Maps Places API does the rest. Check for bikes available to rent via Bike Share network which is updated every 15 minutes.

### **The advantages/positive/new things of your application?**

In comparison to other applications and platform, this application uses the power of map to plot events locations. While most platform doesn't offer this feature. Usually information is listed as a blog post [7]. Also places of drinking fountains, bike shops, and bike share docks are never displayed in one place. API calls were used to achieve this. It's also easy to use and with a simple user interface, thanks to Materializecss. As it's a web app, it can be run on any platform, computer to mobile.

## **Related Work**

When it comes to related application for cyclists, there are many websites and mobile apps. Examples include:

- <http://www.melbournecyclist.com>
- <https://cycling.org.au/vic>
- <https://www.strava.com/local>
- <https://www.bicyclenetwork.com.au/rides-and-events/>
- Google Maps navigation for cyclists routes.

I haven't found a platform which is dedicated only to nearby events and rides. Mostly, this type of information is found as blog posts [7] which don't really cover everything from my experience. Some events are only known between local people. This web app would be the perfect platform to be used by all interested locals.

## Software Design / Architecture

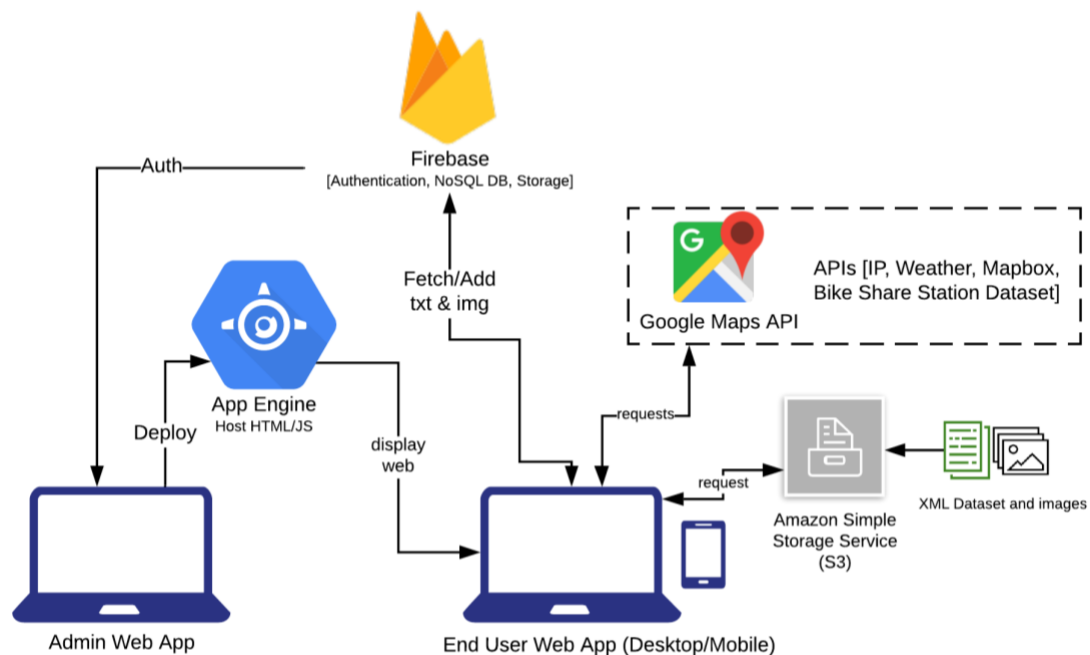


Figure 1: Architecture Diagram

### Cloud Services used:

- Google App Engine for deploying/hosting
- AWS S3 for storage of web app files (XML dataset and images)
- Firebase for Authentication, DB, and Storage of images uploaded when adding info to the DB.

### Cloud Tools used:

- REST APIs from Google Maps, Mapbox, OpenWeatherMap, IP location, Melbourne bike share stations. All return in JSON format which then handled via JavaScript
- AWS service with Amazon Simple Storage S3 and HTTP GET call to retrieve data from (DrinkingFountains.XML) dataset
- Real-time data creation and retrieval from Firebase Cloud Firestore

### API Keys:

**Melbourne Dataset:** UPwcXKkm4fIbcEmoRCjfxGAC0

**Weather:** eccf22da55066e5d6f4d1b4f4bcc8dde

**Google Maps:** AIzaSyAQqXzE9r5h9VYyoKuQ17pFQ5eIX1U4DxNM

**Mapbox:** pk.eyJ1IjoibW9hZmFxIiwiaYSI6ImNrMXUyZGcycjBIYWkzbW8wbmI0c3lrUGUifQ.5LaqXBC2IoMotWH1Kpttqg

### Login details with Firebase Authentication:

**Email:** admin@cyclistshub.com

**Password:** admin.2019

## Implementation

1. Signup in Google App Engine (for deploy and hosting), Firebase (for Authentication, Database and Storage), and AWS S3 (via AWS Educate Starter Account)
2. Signup to get keys in: City of Melbourne Open Data, OpenWeatherMap
3. Enable Google Maps JavaScript API and Places API (this latter used in bike shops search) to get keys
4. Firebase Cloud Firestore (stores data as collections of documents) is used instead of Realtime Database (stores data as one large JSON tree). Both have offline support so data can be created offline. Once online it updates.
5. Change rules in Database and Storage so that only logged in users can write (see figure 7&8)
6. In AWS S3, upload the images/dataset used in web application and then set CORS configuration to allow get information from dataset (see figure 11). "Make public" to all files
7. In Firebase, copy initialization code for the web and paste in your JavaScript file
8. In Firebase Authentication, enable Email/Password under sign-in method and then add user email and password
9. In Firebase Database, add collection with name "events" as this will store all posts added by the admin via the web app.
10. Now it's the coding part, in both HTML and JS
  1. For web visualisation (<https://materializecss.com>) is used
  2. Check comments in the source code
  3. initMap function is for setting the Google Maps API with markers
  4. XML HTTP request is used to GET from Drinking\_fountains dataset
  5. Bike shops are retrieved using Places API
  6. Bike share station docks are updated in the City of Melbourne Open Data every 15 minutes and therefore, is accessed via The Socrata Open Data API returning an API Endpoint of type JSON. One dataset holds the locations and the other holds the live status of docks. Both have station ID column. Combined the two datasets with 2 for loops
  7. displayEvents function display posts added by the admin and plot their locations on the map. Map is built with Leafletjs which is an open-source JS library for mobile-friendly interactive maps along with Mapbox to load the map layers.
  8. Database is retrieved in real-time without the need to refresh the page. This is done using onSnapshot() Firebase function and is ordered according to the time created
  9. When adding content, the image is uploaded to Firebase storage and their download link is added to Firebase database to be used when it's retrieved
  10. When deleting a post, its associated image is also deleted from Firebase storage (check btnPost2Delete listener function for more details)

## User Manual

Using this web application is fairly simple! Admin log-in in order to add/delete posts. Adding contents with fields [title, description, location as Latitude & Longitude, and an image] while deleting a post is by providing post ID which is auto generated. Public users check weather, posts along with their locations on map, and maps with different location markers related to cycling.

Please check the following video demo (on computer and mobile web browser):

<https://youtu.be/O4OWwpCoiQw>

## Conclusion

To conclude, implementing this project motivates me to learn more about web app development and the power of cloud services to achieve certain tasks in an efficient way. There are more to learn, to be ready in the industry, from framework and other cloud services to help developing web apps as well as mobile apps reducing the cost and delivering the product to the end user even faster.

## Appendix – Screenshots

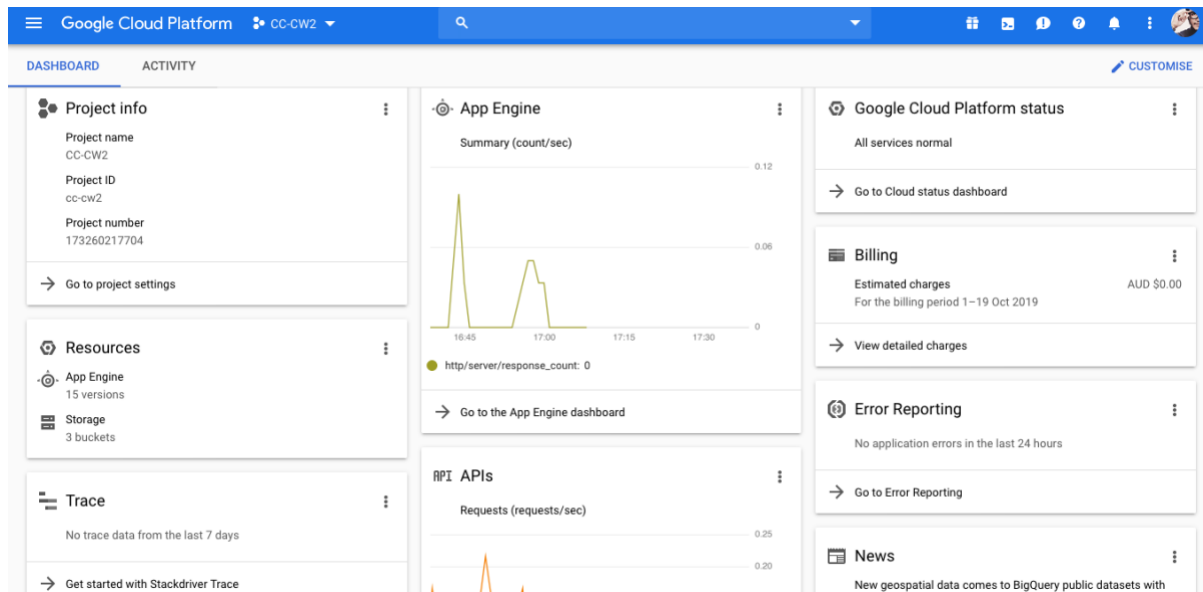


Figure 2: Google Cloud Dashboard [App Engine]

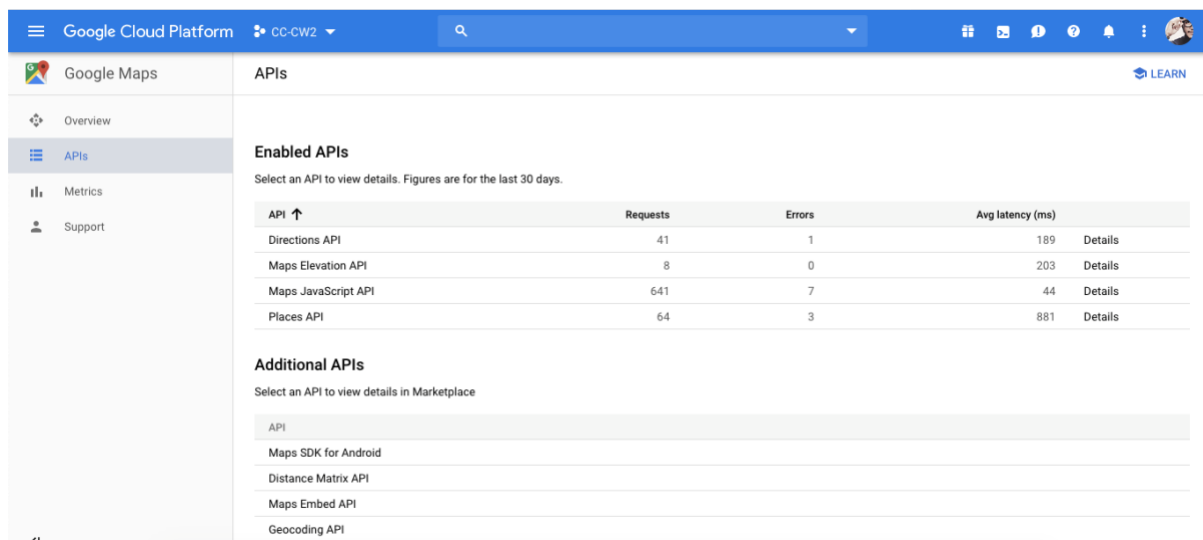


Figure 3: Google APIs

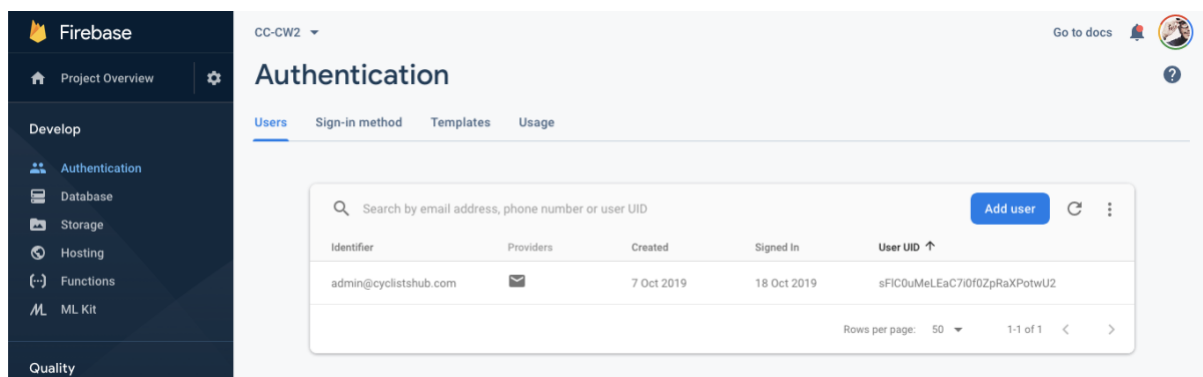


Figure 4: Firebase Authentication



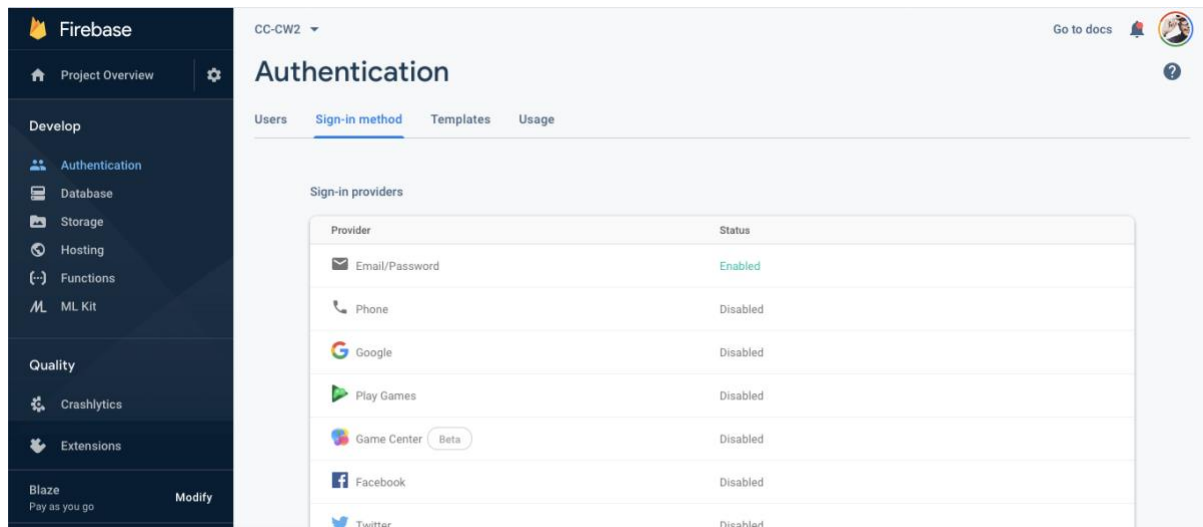


Figure 5: Firebase Authentication

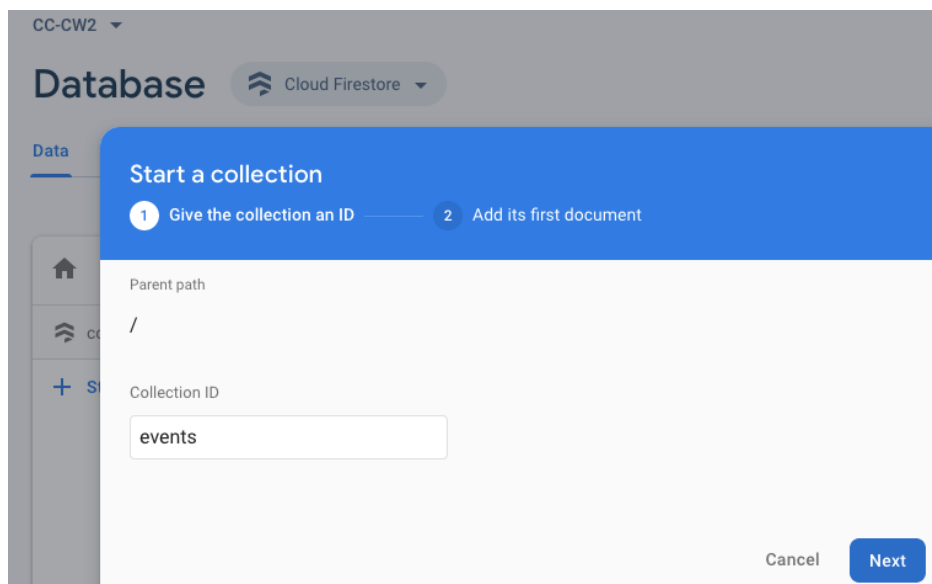


Figure 6: Firebase Cloud Firestore – Add Collection

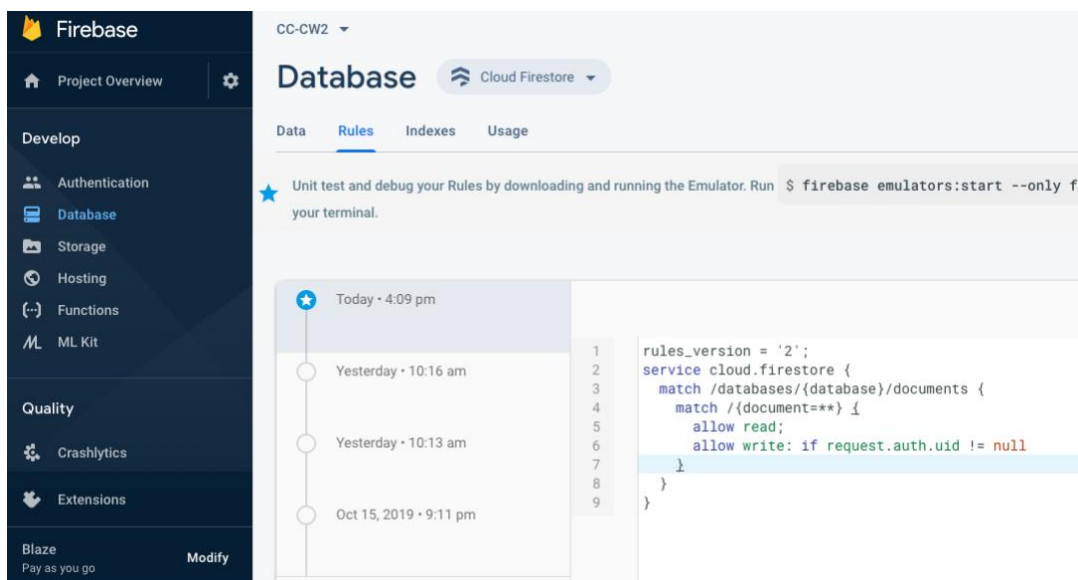


Figure 7: Database Rules to only allow authorized users to write

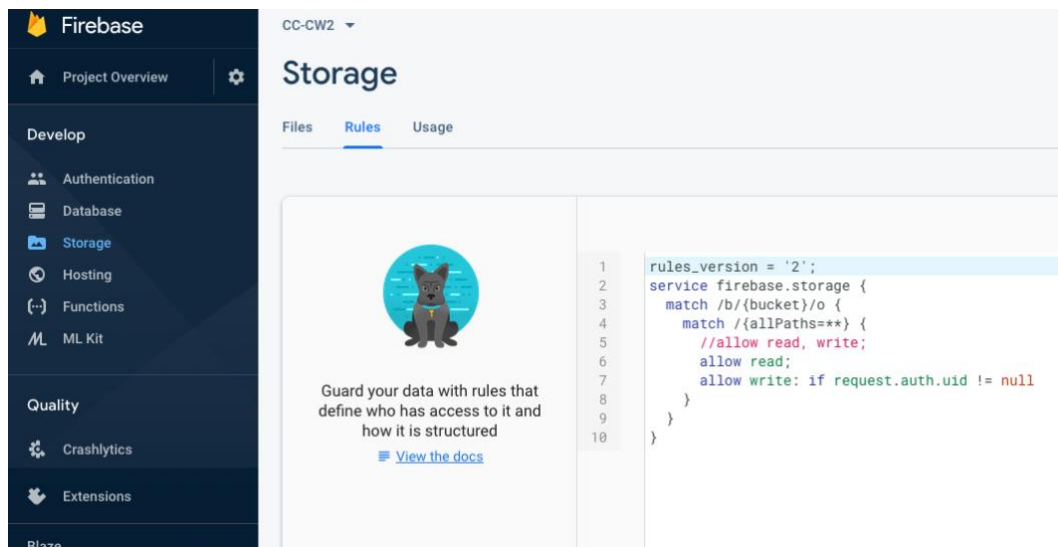


Figure 8: Storage Rules to only allow authorized users to write

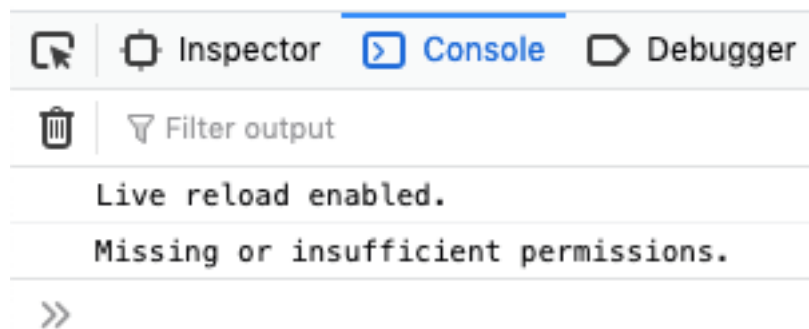


Figure 9: in case unauthorized user tried to add to DB

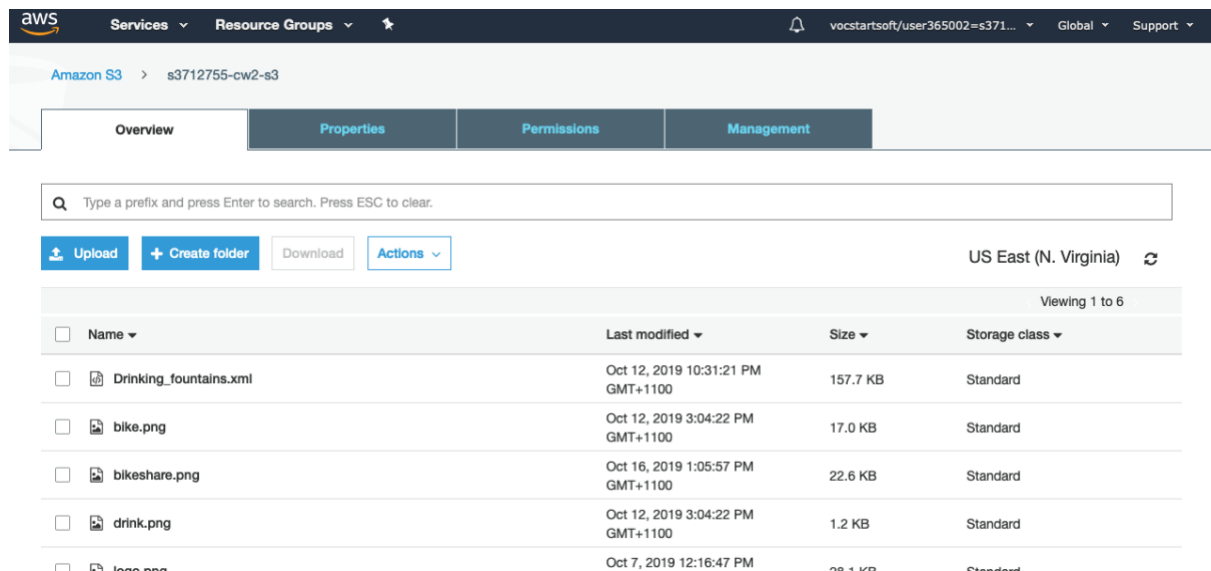


Figure 10: Amazon S3 bucket

Overview

Properties

Permissions

Management

Block public access

Access Control List

Bucket Policy

CORS configuration

CORS configuration editor

ARN: arn:aws:s3::s3712755-cw2-s3

Add a new cors configuration or edit an existing one in the text area below.

Delete

Cancel

Save

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <CORSConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
3   <CORSRule>
4     <AllowedOrigin>*</AllowedOrigin>
5     <AllowedMethod>GET</AllowedMethod>
6     <MaxAgeSeconds>3000</MaxAgeSeconds>
7     <AllowedHeader>*</AllowedHeader>
8   </CORSRule>
9 </CORSConfiguration>
10
11

```

Figure 11: Amazon S3 bucket CORS config

CITY OF MELBOURNE

OPEN DATA

Search

About

Apps & Analysis

Developers

More

Contact Us

Sign In

Bike Share Dock Locations

Transport & Movement

View Data

Visualize

Export

API

Melbourne Bike Share is a joint RACV/Victorian Government bicycle hire scheme. It allows commuters to hire a bike from a dock location and return it to another dock location in the city. This dataset contains the bike share dock locations and capacity across the city.

Updated

September 19, 2019

Data Provided by

City of Melbourne, Australia

More

About this Dataset

Updated

September 19, 2019

Data Last Updated

October 15, 2018

Metadata Last Updated

September 19, 2019

Quality

Update frequency

Daily

Reliability level

Reliable and timely

Figure 12: Dataset of Bike Share Docks location

station_id	Unique bike share station ID	Number	#	
name	Name of the bike station	Plain Text	T	
rental_method	Payment options	Plain Text	T	
capacity	Total number of bikes the station can accommodate	Number	#	
lat	The latitude of the bike share station	Number	#	
lon	The longitude of the bike share station	Number	#	
location	The latitude and longitude of the bike share station	Location		

Table Preview

View Data

Create Visualization

station_id	name	rental_method	capacity	lat	lon	location
2	Harbour Town - Doc...	CREDITCARD,KEY	23	-37.814022	144.939521	(-37.814022°, 144.93...
6	State Library - Swans...	CREDITCARD,KEY	11	-37.810702	144.964417	(-37.810702°, 144.96...
7	Bourke Street Mall - ...	CREDITCARD,KEY	11	-37.813088	144.967437	(-37.813088°, 144.96...
8	Melbourne Uni - Tin ...	CREDITCARD,KEY	19	-37.79625	144.960858	(-37.79625°, 144.960...
9	RMIT - Swanston St / ...	CREDITCARD,KEY	7	-37.807699	144.963095	(-37.807699°, 144.96...

Figure 13: Dataset of Bike Share Docks location [No availability status]

## Melbourne Bike Share stations, with current number of free and used docks (every 15 minutes)

Transport & Movement

Melbourne Bike Share is a joint RACV/Victorian Government bicycle hire scheme. This dataset contains the number of bikes located at each station and the number of empty slots at the station.

[More](#)

Updated  
October 19, 2019

Data Provided by  
City of Melbourne, Australia

### About this Dataset

Updated  
**October 19, 2019**

Data Last Updated  
October 19 2019

Metadata Last Updated  
October 19 2019

#### Quality

Update frequency  
15 Minutes

Reliability level  
Reliable and timely

Figure 14: Dataset of Bike Share Docks availability status

### Columns in this Dataset

Column Name	Description	Type
<a href="#">station_id</a>	Unique bike share station ID	Number #
<a href="#">available_bikes</a>	The number of available bikes at the station	Number #
<a href="#">empty_docks</a>	The number of empty docks at the station	Number #
<a href="#">capacity</a>	Total number of bikes the station can accommodate	Number #
<a href="#">last_updated</a>	When the data was last updated	Date & Time

### Table Preview

[View Data](#) [Create Visualization](#)

station_id	available_bikes	empty_docks	capacity	last_updated
10	1	10	11	2019 Oct 19 11:53:54 AM
11	19	8	27	2019 Oct 19 12:04:49 PM
12	4	6	11	2019 Oct 19 04:44:11 AM
14	12	7	19	2019 Oct 19 04:24:23 AM
15	6	5	11	2019 Oct 19 10:23:26 AM

Figure 15: Dataset of Bike Share Docks availability status [but no location]

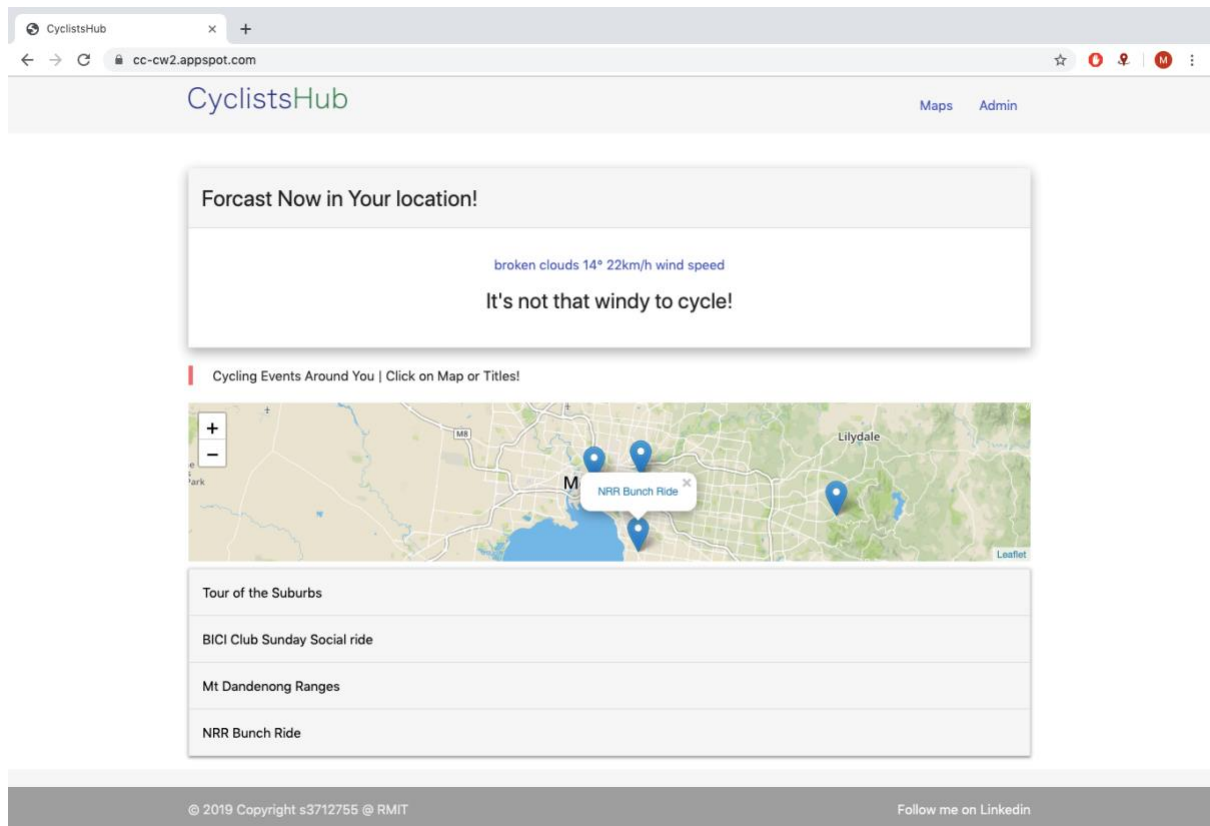


Figure 16: Main Page

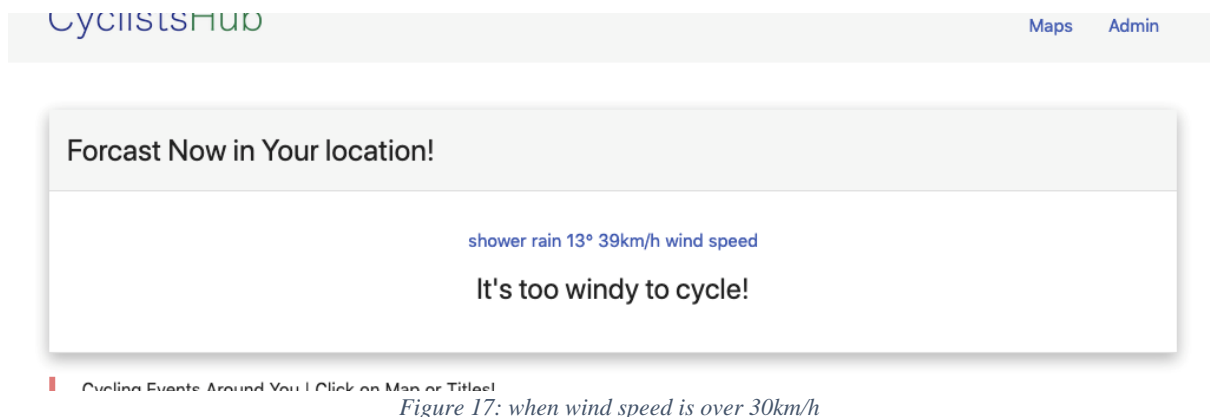


Figure 17: when wind speed is over 30km/h

CC-CW2 Go to docs

## Database

Cloud Firestore

Data Rules Indexes Usage

events > 3MgdGZeVvsk1...

cc-cw2	events	3MgdGZeVvsk1SASK8cio
+ Start collection	+ Add document	+ Start collection
events >	3MgdGZeVvsk1SASK8cio >	+ Add field
	E5uCeHSXz00n0BYzsSEf Jcx0mEBNk0zU8vk3YTx1 V0KqaJgISfML8B1iIjvj	description: "NRR stands for North Road Ride and it's available all weekdays. Monday, Wednesday and Friday is easy rolling. Start Time: 5:45am Distance: 40km"  image: "https://firebasestorage.googleapis.com/v0/b/cc-cw2.appspot.com/o/images%2FNRR.jpg?alt=media&token=6e434c7c-aba5-4442-86cb-1f48639519c4"  imageName: "NRR.jpg"  location: [37.900978° S, 145.006929° E]  timeCreated: 20 October 2019 at 18:45:26 UTC+11  title: "NRR Bunch Ride"

Figure 18: Database

CC-CW2 Go to docs

## Storage

Files Rules Usage

gs://cc-cw2.appspot.com > images

[Upload file](#)

Name	Size	Type	Last modified
NRR.jpg	196.97 KB	image/jpeg	20 Oct 2019
sundayRide.jpg	482.99 KB	image/jpeg	20 Oct 2019
tourofsuburbs1.jpg	286.85 KB	image/jpeg	20 Oct 2019

Location of default bucket: us

Figure 19: Storage

## References

- [1] "Adding a Google Map with a Marker to Your Website", Google Developers, 2019. [Online]. Available: <https://developers.google.com/maps/documentation/javascript/adding-a-google-map>. [Accessed: 14- Oct- 2019].
- [2] "Bike Share Dock Locations", City of Melbourne Open Data, 2018. [Online]. Available: <https://data.melbourne.vic.gov.au/Transport-Movement/Bike-Share-Dock-Locations/vrwc-rwgm>. [Accessed: 08- Oct- 2019].
- [3] "City of Melbourne, Melbourne Bike Share stations, with current number of free and used docks (every 15 minutes)", City of Melbourne Open Data, 2015. [Online]. Available: <https://data.melbourne.vic.gov.au/Transport-Movement/Melbourne-Bike-Share-stations-with-current-number-/tdvh-n9dv>. [Accessed: 08- Oct- 2019].
- [4] "Current weather data", Open Weather Map, 2019. [Online]. Available: <https://openweathermap.org/current>. [Accessed: 09- Oct- 2019].
- [5] "Drinking fountains Dataset", City of Melbourne Open Data, 2014. [Online]. Available: <https://data.melbourne.vic.gov.au/Assets-Infrastructure/Drinking-fountains/h4ih-tzqs>. [Accessed: 07- Oct- 2019].
- [7] "Melbourne Bunch Rides | CyclingTips", *CyclingTips*, 2019. [Online]. Available: <https://cyclingtips.com/bunch-rides/bunchridefinder>. [Accessed: 18- Oct- 2019].