

Getting Started at Hackathons

Track 2: Building a MVP with Firebase and ReactJS



Hi, I'm Eric Jiang 🖐️

- Currently, the Project Lead for monPlan
- Co-founded GeckoDM and MARIE.js
- Co-founded and Pitched FutureYou to SMC, now spun that off as a seperate project
- 🐦 @lorderikir
- 🔗 <https://lorderikir.me>
- 📧 eric.jiang@monash.edu
- github.com/lorderikir

Prerequisites

- NodeJS (preferably with YarnPKG)
- An IDE 
- An Internet  Connection

What is Firebase?

Firebase is a mobile and web application development platform developed by Firebase, Inc. in 2011, then acquired by Google in 2014.

— Wikipedia

We are going to use Firebase Hosting in this demo

What is ReactJS

ReactJS is a **component-based** that is used to build user-interfaces whether its for websites (Frontend) or hybrid (ElectronJS) applications

LIVE JSX EDITOR

☒ JSX?

```
function HelloMessage({name}){  
  return <div>Hello {name}</div>  
}  
  
ReactDOM.render(  
  <HelloMessage name="Taylor" />,  
  mountNode  
);
```



RESULT

Hello Taylor

Installing Create-React-App

CRA is a zero-configuration generator tool that can get us started up quickly

```
npm install -g create-react-app
```

```
# or if you are using yarn
```

```
yarn add -g create-react-app
```

We will now create our new app.

So after we clone our git repository, we look at using create-react-app into our new directory

```
create-react-app myapp
```


We then go into our new directory and install all the dependencies we need

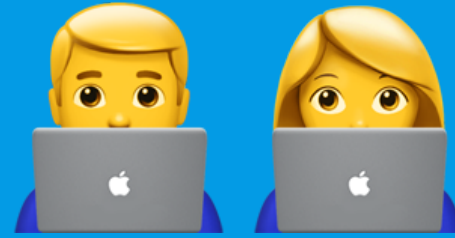
```
npm install material-ui@next material-ui-icons whatwg-fetch --save
```

```
# or
```

```
yarn add material-ui@next material-ui-icons whatwg-fetch
```

- We're using fetch polyfill here as `fetch` is built in natively into the browser, and it is not available for IE11 or prior
- We're also using the beta version of material-ui v1 (it will go GA Soon)

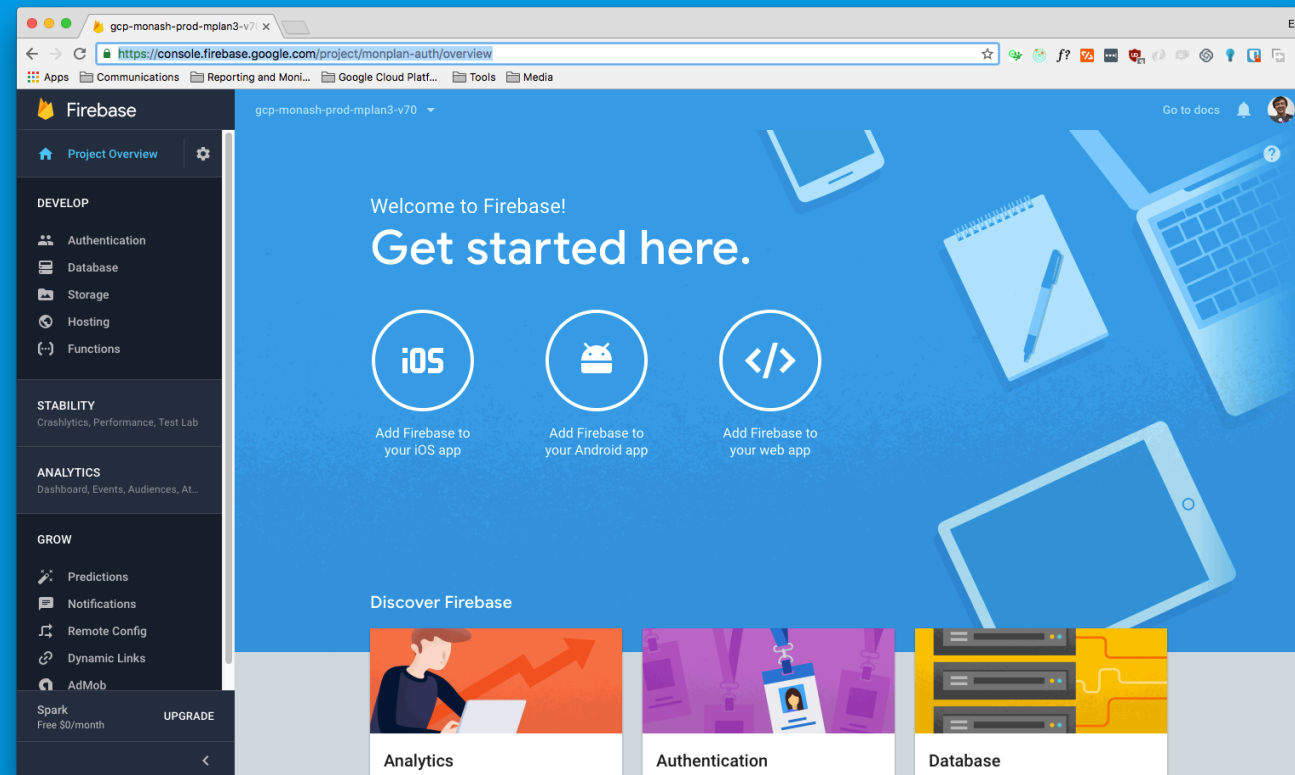
Time for coding!



Let's build an app which users can see and search all the rooms within Monash



Initialise Firebase Project



1. Go to console.firebase.google.com/
2. Create a new project
3. And we're good to go!

Let's build a Room Card First

```
// src/components/RoomCard.js

import React from "react";
import { Card, CardMedia } from "material-ui";
import { CardContent } from "material-ui/Card";
import Typography from "material-ui/Typography";

export default function RoomCard({ roomCode, roomLocation, roomPicture }) {
  return (
    <Card>
      <CardMedia
        image={roomPicture}
        style={{
          height: 200
        }}
      />
      <CardContent>
        <Typography variant="subheading">{roomCode}</Typography>
        <Typography variant="title">{roomLocation}</Typography>
      </CardContent>
    </Card>
  );
}
```

We can easily reference this directly in our main component

```
// src/app.js
import React, { Component } from "react";
import logo from "../logo.svg";
import "../App.css";
import RoomCard from "../components/RoomCard";

class App extends Component {
  render() {
    return (
      <div className="App">
        <header className="App-header">
          <img src={logo} className="App-logo" alt="logo" />
          <h1 className="App-title">Welcome to React</h1>
        </header>
        <RoomCard
          roomCode="S11_LECTURE_HALL"
          roomLocation="17 Rainforest Walk"
          roomPicture="https://www.monash.edu/__data/assets/image/0009/292365/science-lecture-theatre1.jpg"
        />
      </div>
    );
  }
}

export default App;
```

Now connect it to Firebase

Firestore Configuration... 🛠️

We will need to setup the configuration for Firestore

```
// src/config/firestore.js

// Import the Firestore modules that you need in your app.
import firestore from "firebase/app";
import "firebase/auth";
import "firebase/database";
import "firebase/datastore";

// Initialize and export Firestore.
const config = {
  apiKey: "MY AWESOME API KEY",
  authDomain: "MY DOMAIN",
  databaseURL: "https://defs-a-secret-project-ddatabase.firebaseio.com",
  projectId: "so-safe",
  storageBucket: "wow-firebase.appspot.com",
  messagingSenderId: "firebase-sender-id-goes-here-i-guess"
};
export default firestore.initializeApp(config);
```

We can also use some API calls!

I recommend you using `fetch` which is a polyfill built into web-browsers, but `axios` is also great for NodeJS Development.

Fetch was installed during the early development as it is not part of IE11 (Legacy Browser support!)

```
const MONPLAN_API_URI = "monplan-api-au-prod.appspot.com";
fetch(MONPLAN_API_URI + "/api/units")
  .then(resp => resp.json())
  .catch(err => console.error(error))
  .then(data => console.log(data));
```


Any questions? 🤔