

# WEATHER DRESSER

Tells you what to wear everyday based on the real-time weather



## THE IDEA

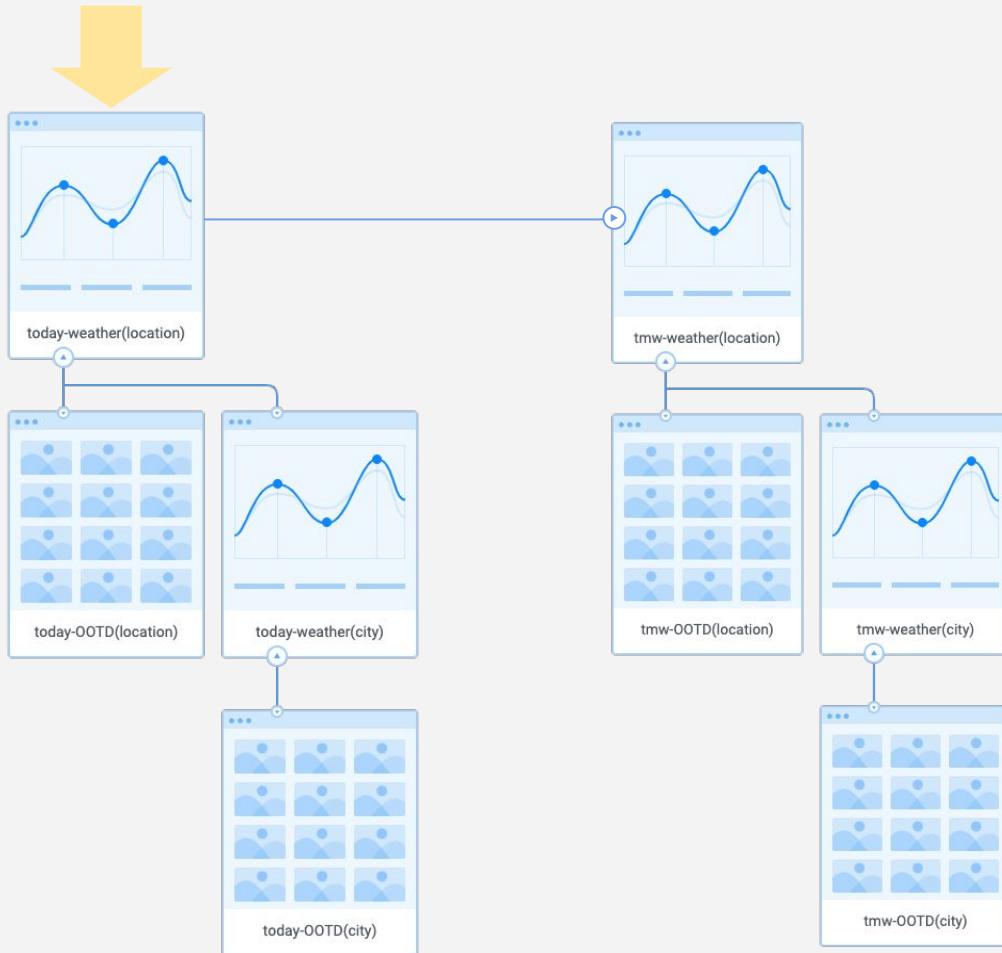


Melbourne is famous for its unpredictable '4-seasons-in-a-day' weather, every Melbournian can relate to how important it is to check real-time weather every morning before we go out so we can pick the right clothes. To save your time in every rush morning, you will be shown matched outfits images according to different weather conditions a real-time so to avoid wearing the wrong outfit for the day.

We all have that weather app that tells you the weather but we probably do not have one that tells you what to wear. So other than simply telling you the weather forecast, Weather Dresser also gives you inspiration about what you should wear for the day, aka "OOTD".



## USERFLOWS



## Icon list

Day icon	Night icon	Description
01d.png 	01n.png 	clear sky
02d.png 	02n.png 	few clouds
03d.png 	03n.png 	scattered clouds
04d.png 	04n.png 	broken clouds
09d.png 	09n.png 	shower rain
10d.png 	10n.png 	rain
11d.png 	11n.png 	thunderstorm
13d.png 	13n.png 	snow
50d.png 	50n.png 	mist

```
{
  "coord": {
    "lon": -0.13,
    "lat": 51.51
  },
  "weather": [
    {
      "id": 500,
      "main": "Rain",
      "description": "light rain",
      "icon": "10n"
    }
  ],
  "base": "cmc stations",
  "main": {
    "temp": 286.164,
    "pressure": 1017.58,
    "humidity": 96,
    "temp_min": 286.164,
    "temp_max": 286.164,
    "sea_level": 1027.69,
    "grnd_level": 1017.58
  },
  "wind": {
    "speed": 3.61,
    "deg": 165.001
  },
  "rain": {
    "3h": 0.185
  },
  "clouds": {
    "all": 80
  },
  "dt": 1446583128,
  "sys": {
    "message": 0.003,
    "country": "GB",
    "sunrise": 1446533902,
    "sunset": 1446568141
  },
  "id": 2643743,
}
```

## MOCKUP DESIGN

splash-src

clear

TODAY 天気ワードローブ

Clear

29°

MELBOURNE

▼min: 18° ▲max 32°

[SEE MY OOTD](#)

2019 All Rights Reserved

snow

TODAY 天気ワードローブ

Snow

3°

MELBOURNE

▼min: 18° ▲max 32°

[SEE MY OOTD](#)

rainy

TODAY 天気ワードローブ

Rainy

12°

MELBOURNE

min: 18° ▲max 32°

[SEE MY OOTD](#)

## MOCKUP DESIGN

cloud

TODAY 天気ワードローブ

Cloudy



21°

MELBOURNE  
min: 18° max 32°

[SEE MY OOTD](#)

thunderstorm

Enter city: Finland  x

Thunderstorm



16°

MELBOURNE  
min: 18° ▲max 32°

[SEE MY OOTD](#)

outfit1

DRESSER ワードローブ

G'Day!

This weather like this is perfect for:



< >

[SHOW ME ANOTHER](#)

outfit2

DRESSER ワードローブ

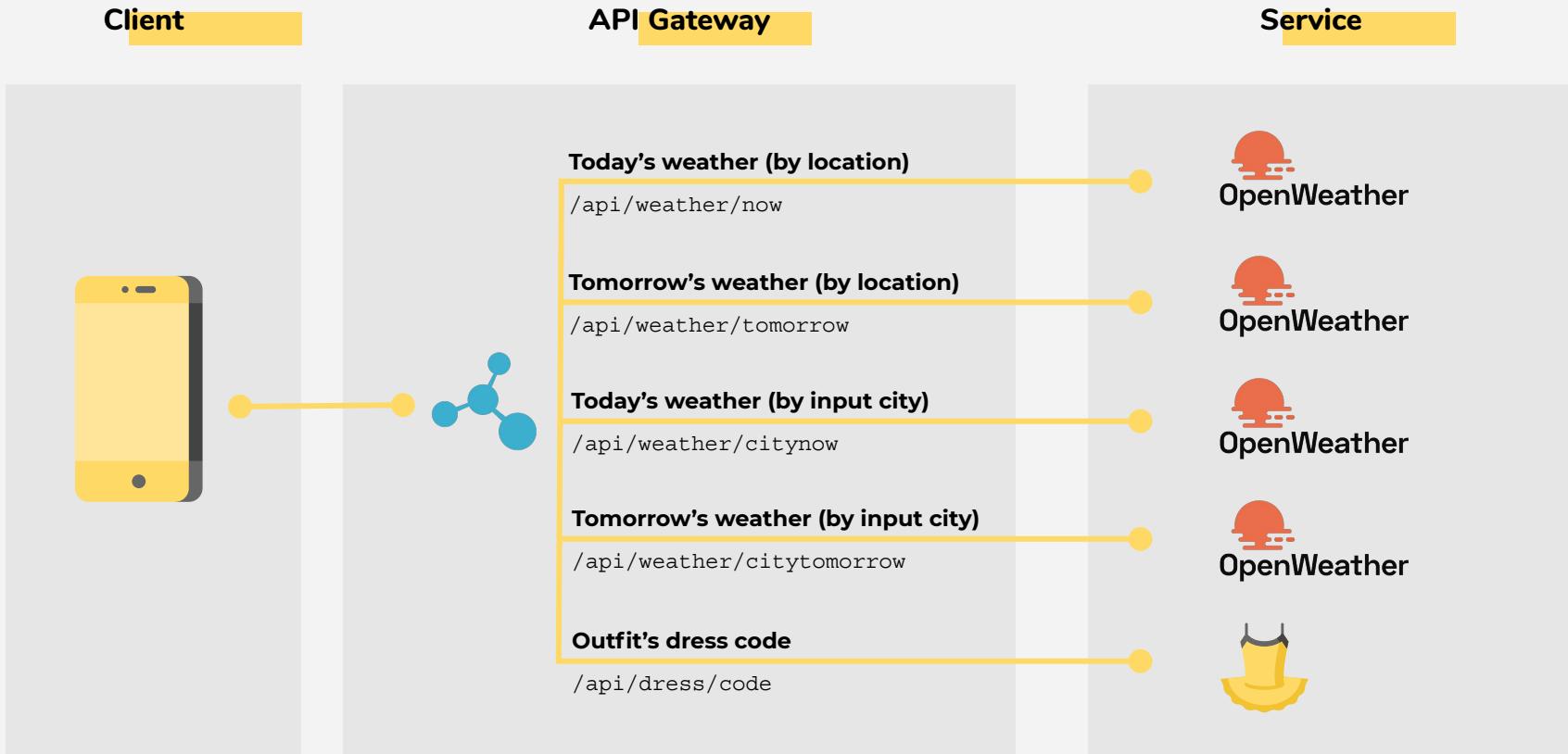
G'Day!

This weather like this is perfect for:

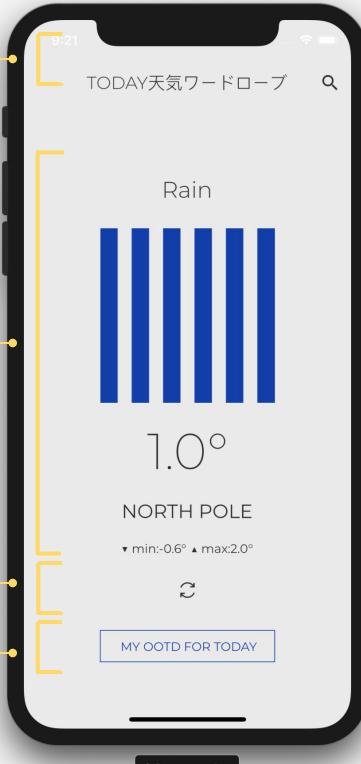


< >

[SHOW ME ANOTHER](#)



### TabBarView(Today)



AppBar

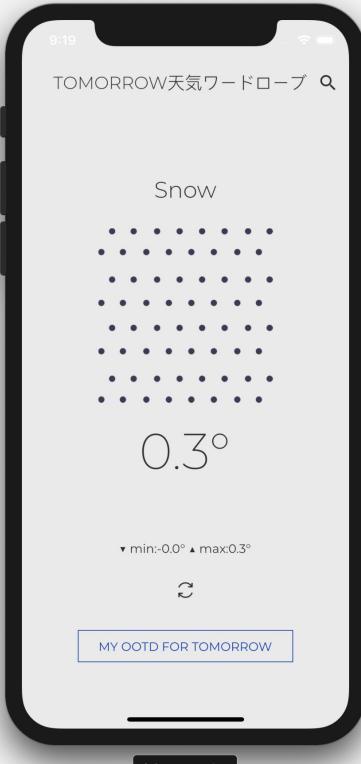
Weather Widget

loadWeather()  
IconButton

PageRoute()  
OutfitPage



### TabBarView(Tomorrow)



iPhone X - 12.1

iPhone X - 12.1

iPhone X - 12.1

iPhone X - 12.1

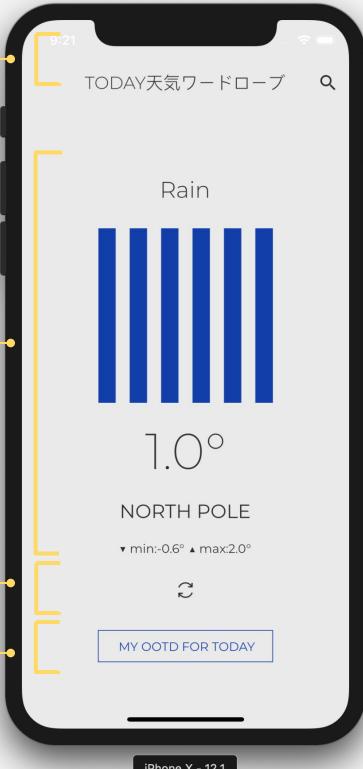
### TabBarView(Today)

AppBar

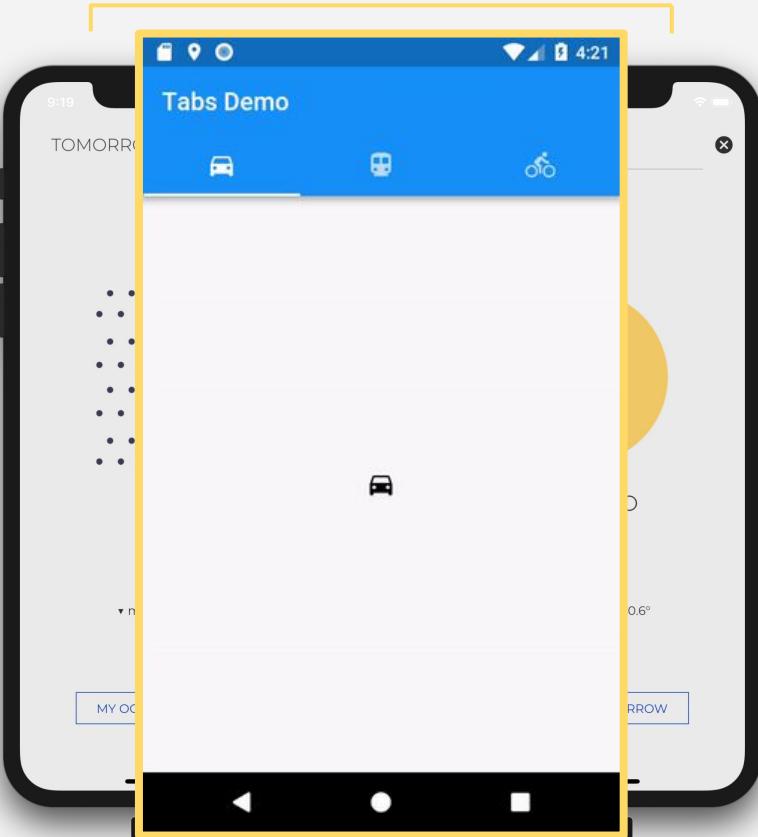
Weather Widget

loadWeather()  
IconButton

PageRoute()  
OutfitPage

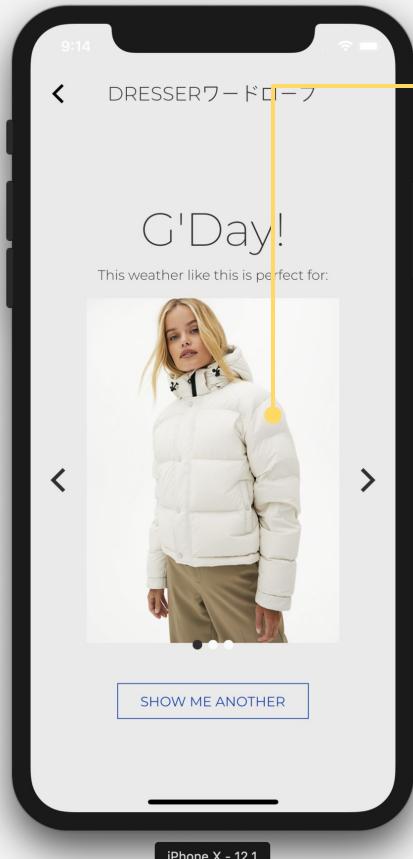


### TabBarView(Tomorrow)



## THE PROCESS

swiper  
Widget



DRESSERワードローブ

G'Day!

This weather like this is perfect for:



SHOW ME ANOTHER

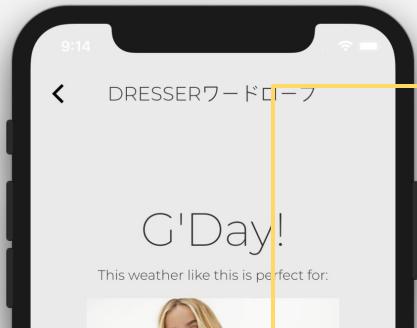
iPhone X - 12.1

dressCode =  
/cold-rain/2/top.jpeg  
temp condition random no. imageNames []

```
// set the second part of the answer based on description
if (
    // if description contains light rain or clouds, weather is variable
    req.params.description.indexOf('light rain') > -1 ||
    req.params.description.indexOf('clouds') > -1 ||
    req.params.description.indexOf('mist') > -1
) {
    answerString += 'variable';

} else if (
    // else if description contains only rain, the weather is rain
    req.params.description.indexOf('rain') > -1 ||
    req.params.description.indexOf('storm') > -1
) {
    answerString += 'rain';
} else {
    // all other cases, is clear
    answerString += 'clear';
}
```

## THE PROCESS



dressCode =

/cold-rain/2/top.jpeg

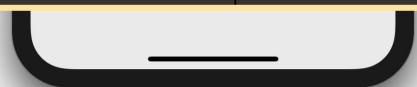
temp

condition

random  
no.

imageNames []

S v	<ul style="list-style-type: none"><li>banner.png</li><li>dress</li><li>favicon.ico</li><li>index.html</li></ul>	<ul style="list-style-type: none"><li>cold-clear</li><li>cold-rain</li><li>cold-variable</li><li>fresh-clear</li><li>fresh-rain</li><li>fresh-variable</li><li>hot-clear</li><li>hot-rain</li><li>hot-variable</li><li>warm-clear</li><li>warm-rain</li><li>warm-variable</li></ul>	<ul style="list-style-type: none"><li>1</li><li>2</li><li>3</li></ul>	<ul style="list-style-type: none"><li>accessory.jpeg</li><li>bottom.jpeg</li><li>top.jpeg</li></ul>
--------	---	---	---	---



iPhone X - 12.1

```
answerString += 'rain';
} else {
    // all other cases, is clear
    answerString += 'clear';
}
```

Postman

New Import Runner +

Filter

History Collections APIs BETA

+ New Collection Trash

now 1 request

GET https://api.openweathermap.org/d...

tmr 1 request

```

5 },
6   "weather": [
7     {
8       "id": 521,
9       "main": "Rain",
10      "description": "shower rain",
11      "icon": "09n"
12    }
13  ],
14  "base": "stations",
15  "main": {
16    "temp": 282.47,
17    "pressure": 1016,
18    "humidity": 81,
19    "temp_min": 280.37,
20    "temp_max": 284.26
21  },
22  "visibility": 10000,
23  "wind": {
24    "speed": 7.2,
25    "deg": 240
26  },
27  "clouds": {
28    "all": 75
29  },
30  "dt": 1570485566,
31  "sys": {
32    "type": 1,
33    "id": 9554,
34    "message": 0.0083,
35    "country": "AU",
36    "sunrise": 1570477560,
37    "sunset": 1570523370
38  },
39  "timezone": 39600,
40  "id": 2158177,
41  "name": "Melbourne",
42  "cod": 200
43 }

```

Postman

New Import Runner +

Filter

History Collections APIs BETA

+ New Collection Trash

now 1 request

GET https://api.openweathermap.org/d...

tmr 1 request

```

2 "cod": "200",
3   "message": 0.0093,
4   "cnt": 40,
5   "list": [
6     {
7       "dt": 1570492800,
8       "main": {
9         "temp": 286.1,
10        "temp_min": 284.618,
11        "temp_max": 286.1,
12        "pressure": 1018.1,
13        "sea_level": 1018.1,
14        "grnd_level": 1014.54,
15        "humidity": 80,
16        "temp_kf": 1.48
17     },
18     "weather": [
19       {
20         "id": 500,
21         "main": "Rain",
22         "description": "light rain",
23         "icon": "10d"
24       }
25     ],
26     "clouds": {
27       "all": 55
28     },
29     "wind": {
30       "speed": 8.97,
31       "deg": 204.341
32     },
33     "rain": {
34       "3h": 0.5
35     },
36     "sys": {
37       "pod": "d"
38     },
39     "dt_txt": "2019-10-08 00:00:00"
40   }

```

Postman

New Import Runner +

Filter

History Collections APIs BETA

+ New Collection Trash

No Env

GET https://api.openweathermap.org/d...

GET https://api.openweathermap.o...

GET https://api.openweathermap.o...

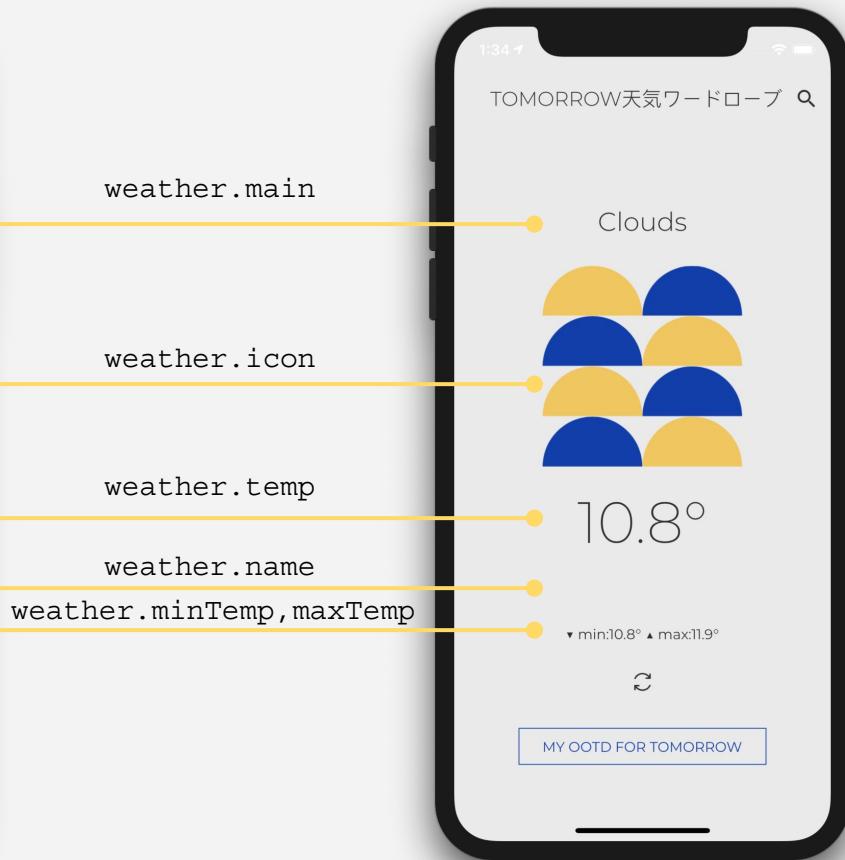
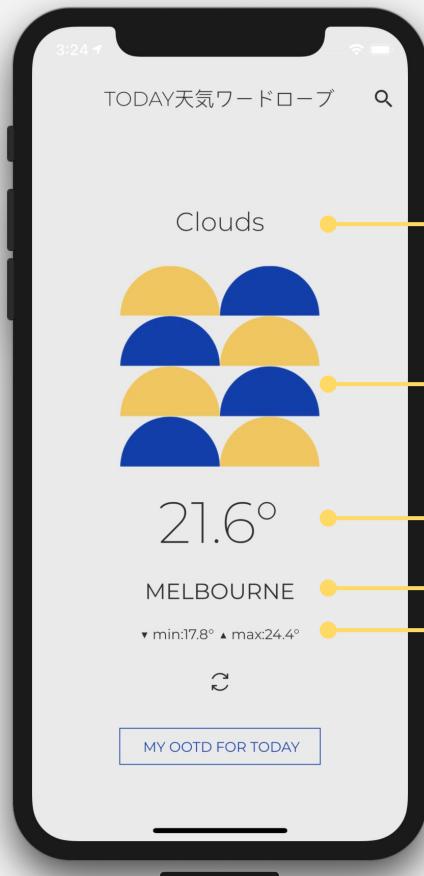
GET https://api.openweathermap.org/d...

```

GET https://api.openweathermap.org/data/2.5/forecast?APPID=931dd482db9863c889f68196f104be72&lat=-37.803423&lon=144.965135

```

## THE PROCESS



## THE PROCESS

```
// the api contains the next 5 days, with array of every three hours
// get useful items from list and store into custom 'tomorrow' which has exact same format as the 'now' one
var fiveDays = JSON.parse(body);
var cod = fiveDays.cod;
var t1 = fiveDays.list[7]["main"]["temp"];
var t2 = fiveDays.list[10]["main"]["temp"];
var temp = fiveDays.list[10]["main"]["temp"];
var main = fiveDays.list[10]["weather"][0]["main"];
var description = fiveDays.list[10]["weather"][0]["description"];
var icon = fiveDays.list[10]["weather"][0]["icon"];

// taken 6am[7] and 3pm[10] as min and max temp but to accurately check which is higher
var temp_min, temp_max;
if(t1 <= t2)
|   temp_min = t1;
else
|   temp_min = t2;

if(t1 > t2)
|   temp_max = t1;
else
|   temp_max = t2;
```

```
// pass all I need to 'tomorrow'
var tomorrow = {
  "cod": cod,
  "main": {
    "temp_min": temp_min,
    "temp_max": temp_max,
    "temp": temp,
  },
  "weather": [
    {
      "main": main,
      "description": description,
      "icon": icon
    }
  ]
};

resolve(tomorrow);
```

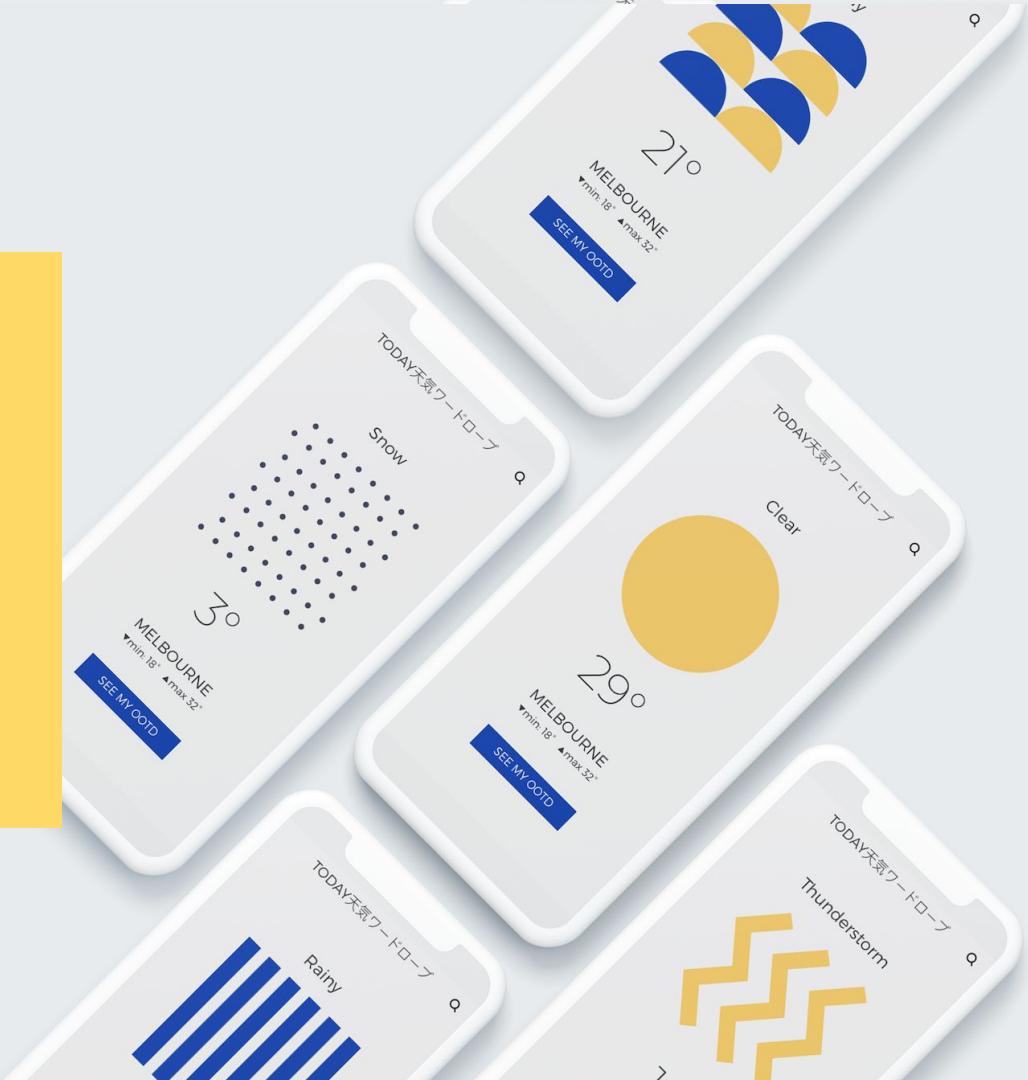
FOR TOMORROW

iPhone X - 12.1

iPhone X - 12.1

# PROJECT DEMONSTRATION

iOS / Android



## REFLECTION



### CHANGES

- User login
- Store users' own OOTD  
in different categories
- Short of time

### CHALLENGES

- Familiar with Flutter!
- Build UI with code
- Getting everything dynamic
- Require a lot of testings

### TAKEAWAYS

- Explore new framework
- Flutter can be effective  
(use Android Studio!)
- Using Microservices
- Version control with Git