

Group 5

1. Introduction

Dublin City Management is web application providing information related to real-time events happening in Dublin city. This application is designed especially for city managers to provide an overview of the updates in the city at any given time. It provides information regarding weather, transport and traffic scenario and city events in Dublin. It also provides analytics and future predictions for various indicators. In this document, the functional architecture of this application is described in detailed.

2. Description

Below is the detailed functional diagram of the application

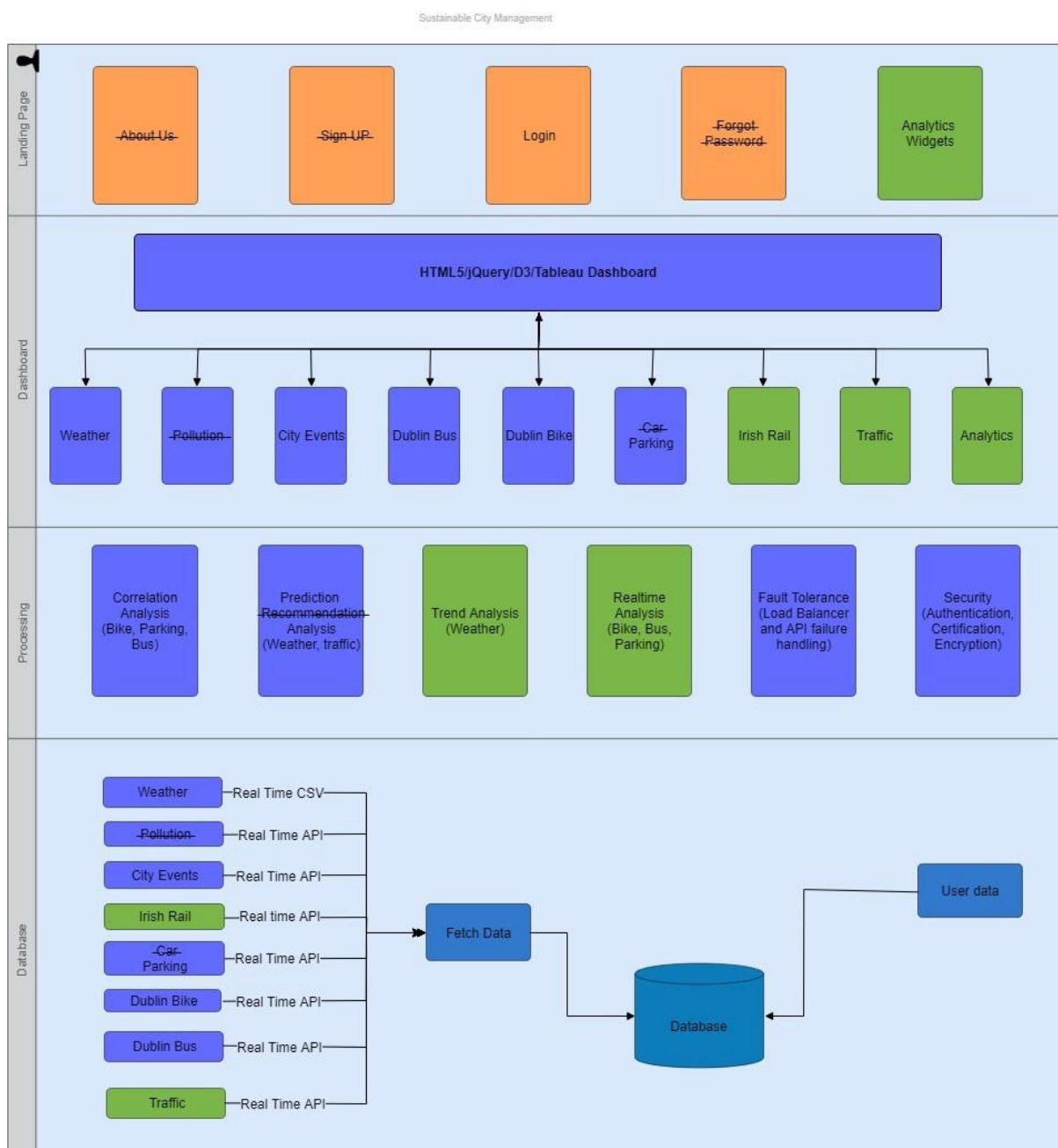


Fig 1: Functional Architecture

As seen from the diagram, we can see the architecture is divided into 4 layers which are as follows:

1. Database layer
2. Processing layer
3. Dashboard(views) layer
4. Landing page

In this diagram, the stroked section indicates the changes made in the final version of the architecture compared to the initial document submitted. Here, the sections in green colour indicate newly added features were not present in the previous version.

1. Database layer:

Database layer provides access to the stored data. In this layer data for all the indicators is gathered and stored in database. The data is collected for Weather, City events, Irish rail, Parking, Dublin Bike, Dublin Bus and Traffic. The data source is of two types real time API and real time CSV. Data collected have a heterogeneous format. City events and Dublin bikes were in the form of json file, whereas that of Irish Rail and Parking had an XML layout. Bus and Luas followed a SOAP/WSDL format, while Weather and Traffic data was obtained in CSV file. All this real time data is fetched at regular interval and stored in the database. Apart from this data about system users is also stored in the database, which was used for login authentication.

In order to have fault tolerance and to decentralize the architecture, two databases are used. One acts as a primary database while other is a secondary database. Backup of data is taken at regular intervals. Postgresql is used as database management system. It manages data and provides access to various data files.

2. Processing Layer:

This layer is the heart of the architecture. It does all processing and provides insights. The data is retrieved from the database layer. Tasks performed in this layer are as follows:

- i. Correlation Analysis: The data related to bike, parking and bus had information related to longitude and latitude. In order to correlate these indicators, the location parameter is used for correlation. By correlating the data is converted into 4 zones namely – centre city, east city, west city and north city.
- ii. Prediction Analysis: This block is responsible for making future predictions of weather and traffic. Machine learning algorithm like random forest and linear regression is used to make these predictions.
- iii. Trend Analysis: Operations on weather data is done in this section. Weather for all areas is analysed and trends are based on historic and realtime data. Trends are made for temperature, wind speed and humidity.
- iv. Realtime analysis: The Realtime data of bike, bus and parking is fetched from the database and statistics are generated. The correlated zones data is used, and various insights are generated.
- v. Fault tolerance and API failure handling: It is responsible for the smooth functioning of all components. Fault tolerance is provided by using a load balancer between two databases. If the link between the primary database is disconnected all the requests are diverted to the secondary database. Both the database is in sync and hence there is no impact on system database failure. API failure handling is done as the data is fetched directly from the database negating the dependency of API failure.

vi. Security: It includes safety features like user authentication, file encryption in the database, CA certification, maintaining session logs. Apart from this, AWS- DDOS and SQL injection protection are provided. All these measures ensure, the system is immune to any cyber-attacks.

3. Dashboard

An individual dashboard is created for all the indicators (Weather, City Events, Dublin Bus, Dublin Bike, Parking, Irish Rail and Traffic). These dashboards are not visible to general users. City admins have the authentication for the same. It provides the visualization of real-time data. These are created in order to provide city managers a clear understanding of various incidents in the city. The Irish rail and Dublin bus dashboard are interactive, which will enable city managers to customize as per the need. Weather dashboard provides information in the form of a trend.

Analytics dashboard will display all the correlated and predicted data. This dashboard will enable managers to view the traffic and other scenarios in the city in detail. Moreover, it will also provide prediction of traffic and weather thus giving an indication of future likely scenario. It also gives zonal level information about the number of parking, bike and bus in each zone. Below is the snippet of the analytics dashboard.

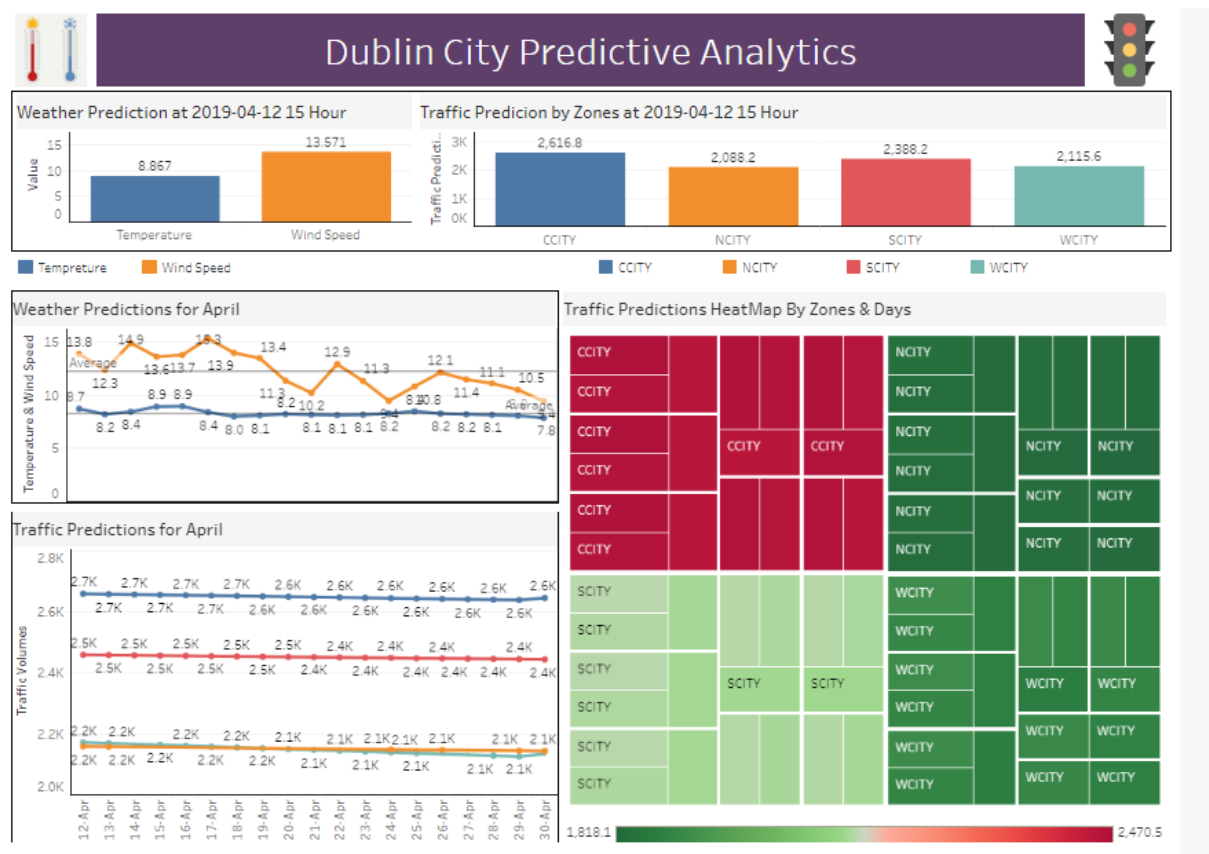


Fig 2: Traffic and weather prediction

Zone Updates Live updates for each zone in the city.	
Zone 1 : Centre + Car Parking + Dublin Bus + Dublin Bike	Zone 2 : South + Car Parking + Dublin Bus + Dublin Bike
Zone 3 : West + Car Parking + Dublin Bus + Dublin Bike	Zone 4 : North + Car Parking + Dublin Bus + Dublin Bike

Fig 3: Zonal information

2.1 Landing Page:

This is the view of the home page. This can be viewed by all users as it does not require any authentication. This page provides on the go updates regarding all indicators. Multiple analytical widgets are created to provide information about bike, parking, events in the city and Irish rail information. All these widgets are designed in a way to overall information of all indicators in one view. Along with this real time traffic map is also shown. This shows information regarding traffic condition and traffic events in the city. All the data which is processed in the processing unit is shown in this page as well on the dashboards.