# 1. Introduction

People often find out there will be long wait when they got to the restaurant. Sometime you just don't care which restaurant but to get a quick bite. It will be better if you can see all the wait time in advance in your surrounding and just go to it. We will create a restaurant wait time application that will provide user real time information about surrounding restaurant wait time on a no reservation basis.

## 1.1 Summary

We shall call this project restaurant wait-time (RWW). This is intended for any user who are looking for place to eat to know in advance about what is the current wait time for a no reservation walk in. There will be 2 groups of user. First group consists of the restaurant owners/hosts who will update the wait time data via a REST API interface from a restaurant client application. Second group consists of restaurant goer who are using the user client application to search and retrieve the wait time data.

## 1.2 Requirement

### 1.2.a Azure REST API interface

- Using .net WEb API.Web API allow more control and debugging locally is easier as compare to Nodejs (in Mobile Service) plus it allow to use Web jobs which is one of the requirement.
- Use Azure SQL to store model data
- Support Twitter and Google authentication
- Web job to process any uploaded files (if needed)
- Use Blob or File Storage for any uploaded files
- Implement Redis Cache
- Implement Notification Hubs

### 1.2.b Restaurant Client Application

- All operation and communication should be using the REST API
- Allow user to update wait time
  - Wait time group by number of people in party (2,4,6,8+, etc)
  - Push out notification when wait time changed
- Allow user to upload needed files (menu, flyer, advertisement, etc)
- Allow Administrator to update restaurant information(type of restaurant, verify location, capacity, special event, etc)
- Call into the redis cache end point for performance

### 1.2.c User Client Application

- Allow any user to browser waiting on their surrounding area
- Allow login user to search resturant by location, type, wait time, etc
- Allow login user to retrieve wait time for individual restaurant or all surrounding restaurants.
- Allow login user to subscribe to individual restaurant for getting wait time update and receive notification when wait time changed.
- Allow login user to leave feedback and rating about the accuracy of the wait time for any individual restaurant

### 1.2.d Unit Test ALL Rest services

## 1.3 Numbers

This is a demo application so the expected number of user and restaurant are low. Expected peak usage time will be during Lunch and Dinner time. However we will design this with scalability in mind knowing in a real system expected user will be in millions where expected resturant user will be in thousands.

## 1.4 Terminology

- RWW - restaurant wait-time
- RESTAPI - Azure .net WEb API
- RCA - Restaurant Client Application
- UCA - User Client Application
- Cache - Azure Redis Cache

# 2. Functional Description

**Use Cases**

- See attached picture

**User Community**

- People looking for a quick bite and do not want to waste time by calling or go to the restaurant to find out.
- Restaurant want to let people know they have open space

**User Functions**

- Register as restaurant user use Twitter or Google
  - Initial restaurant user must be a administrator user who can register more restaurant administrator or non-administrator user.
  - Non-administrator user can update wait time by party number on time interval. Wait time can also be updated by hook into their existing reservation/queuing system (not implement here)
  - Non-administrator user can upload files
  - Administrator user can do everything non-administrator user did plus they can update restaurant detail information
  - Any wait time change will be pushed out as notification
  - Can delete account
- Register as normal user use Twitter or Google
  - Can change user information
  - Can search by criteria
  - Can get wait time for single or all resturant
  - Can leave feedback and rating about the wait time
  - Can delete account
- Anonymous user
  - No search only can see surrounding restaurant and can not receive notification or leave feedback

**Error Handling**

- Map Azure WEB API Service REST errors to standard HTTP Error codes
- All exception from the client application must be handled

**Security**

- Make sure all REST API interface that need authentication is secure
- Secure SQL injection attack
- Secure unauthenticated access to costly resources

**Boundary Conditions**

- Because this is a demo application we do not expect top performance and the lowest tie of Azure will be used.

**Constraints**

- Does not factor in user location, traffic situation when showing wait time and no recommendation is given. So place A maybe 10 minutes wait time but it is 20 minutes from you while place B is 15 minutes wait time but only 5 minutes from you so we should suggestion you go to place B.

**Platforms**

- Azure Web API
- RCA - Universal Windows App
- UCA - Universal Windows App

**Internationalization**

- Not in consideration under the current scope so English only

**Portability**

- REST API give us flexibility to deploy and use on any platform that talk HTTP
- Universal Windows App allow us to deploy the client application to any qualify Windows devices

**Expandability**

- Not in consideration under the current scope

**Customization**

REST API Client SDK is generated for any client customization application.

**Support & Maintenance**

Are any functions to be included to make maintenance and support easier, e.g. internal monitoring of traffic flows.

**Configuration Management**

- During development GIT will be used locally and sync back to a GIT server

**Documentation**

- Will be provided on a separate User Documentation