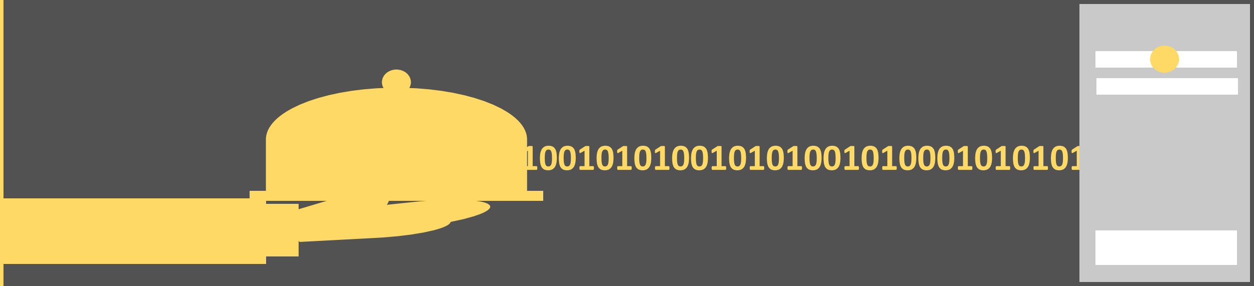
RESTFull API

Product Vision Document



A project vision by:
Alex Billson
Kevin Christopher
Martin Lim

Purpose of this document

This Product Vision document serves as a more detailed overview of both the purpose and requirements specification of the system to be developed. This will document will also help to shine a clear understanding to the value of the project and the scope at which it can operate.

OUR VISION

To develop a lightweight yet powerful REST API service for providing restaurant data for integrations

TARGET GROUP

Restaurant owners & Managers.

Developers

PRODUCT

A restfull API written in ASP.NET Core MVC using the WebAPI system to provide a logical, powerful and lightweight solution.

NEEDS

Integrate restaurants into mobile apps, web apps and the internet of things.

Provides a simple, flexible API system that can be easily deployed cross-platform and modified to suit the need of the client

VALUE

Through the usage of a RESTFull, dining venues will be able to provide an easy, universal system for developers, both internally and externally to utilise.

By providing an access point for a restaurants data, venues are embracing a modern age of apps and websites which will improve the livelihood of their business.

The problem it solves and the value it gives

The world is evolving and so does the way people run their businesses. Restaurant businesses had been an integral part of human society but in this new modern age where everything seems to move towards digitalization, so does restaurants.

This means that there will be a great need for restaurants to provide details in relation to what they offer as a business in a mean that is digitally consumable.

It would bring tremendous benefit for our organization to implement a system such as RESTFull which allows the ability to easily work with data relating to the restaurant to be utilized by developers and the owners.

The initial architecture of RESTFull

The API itself will be written using ASP.NET Core. This version of ASP.NET unlike previous versions runs cross platform and is based on the up and coming .NET Core framework form Microsoft. This choice works very well with what is required from RESTFull, its ability to be modularly deployed onto any platform with all dependencies and frameworks kept together allowing for any restaurant (not just the client's) to host their own version of RESTFull if they wish without being constrained to a single Operating System.

What are the cost estimation?

Luckily both ASP.NET Core and SQLite are open sourced technologies with .NET Core released under the MIT License and SQLite being public domain.

There are some minor investment such as:

-utilizing a lowest tier Linode Linux environment | $5

-hosting of the server

The scope of this project

The original build of RESTFull will consist of two iterations. In each iteration the development team will do parallel work, a front-end and a back-end. Note that developers or restaurant owners can modify the system to a certain degree to suit their needs.

Back-end details:

The back end is where the API is truly made. The back-end team will implement the technical database in the list above and the core functionality of the API.

Front-end details:

The front-end will link to the back-end and develop a user interface in which data are linked to the database. This allows for manipulating the data through a Graphical user interface. It also shows the potential and capability of the API sort of like a tech demo.

The back-end and front-end will work together especially in later iterations as further enhancement such as authentication requires a cohesion between the back and front end side of the team.

ITERATION 1

Purpose of this iteration:

Provide the basic core functionality of the system to allow restaurants to save general restaurant data to a portable SQLite database. This includes:

- Menu Items
    - Name and Description
    - Nutritional Value
    - Price
    - Item Type
- Restaurant Details
    - Address
    - Proprietor
    - Franchise
    - Official Business Name
- Special Offers and variances in price
    - Single item percentage discounts
    - “Multi buy” set price discounts in which multiple items result in a set price

ITERATION 2

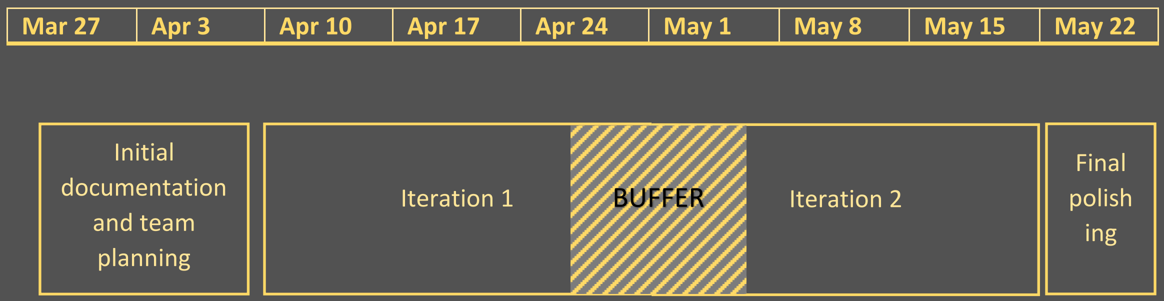
Purpose of this iteration:

To enhance and refine existing features previously done in iteration1 through the implementation of additional features which increases the overall quality and usability of the system. These enhancement includes:

- Authentication
    - All API calls will have some form of basic token authentication (optional)
    - Authentication will especially be stressed when dealing with POST, UPDATE and DELETE requests
    - Restaurants will be able to be provided a developer/user login system to generate a token for usage
- Better Menu item customization
    - Menu items will have better customization options such as deeper item descriptions, analytics for popular items
    - Items will be tag-able for easy categorization
- Table details (not a core function)
    - Tables for a restaurant will be accessible through the API
    - Include details such as seat count and possible bookings

Release Plan

Note that this plan is an estimation and actual release may occur either sooner or slightly later.



Mar 27	Apr 3	Apr 10	Apr 17	Apr 24	May 1	May 8	May 15	May 22

Initial documentation and team planning

Iteration 1

BUFFER

Iteration 2

Final polishing

Software development process

Our development team adopts the agile SDLC process. We will conduct periodical meeting as a status report. A form of backlog is implemented in Trello which team members can pull on our current specific iteration. We will repeat this iterative process until the project is finished.

We also understand the importance of unit testing and will perform them throughout the development process to make sure the code components performs as expected.

Our development team

BACK-END DEVELOPER

ALEX BILLSON

Head Programmer
Database Structure & Migrations

KEVIN CHRISTOPHER

Backend Developer
Data Modelling

- .NET Core
- ASP.NET MVC 5's Web API setup
- Utilizing Entity Framework for Database Migration and Connection
- SQLite

FRONT-END DEVELOPER

MARTIN LIM

Head of Front-End Design
UI Designer

- HTML, CSS & JavaScript
- AngularJS framework
- Bootstrap framework (UI)
- Adobe Illustrator