# Table of Contents

# 1. Introduction

## 1.1 Project Overview

Our project, CUc@nteen (meaning "see you at canteen") aims to provide a platform for CUHK canteen customers to gain canteen information. We hope to create an environment where interactions between customers themselves and also in between customers and canteens could occur. Hopefully this kind of communication could eventually create a win-win situation for customers and canteens, where enhancing the quality of canteen food would increase customers standard of living, and attracting more people to go to different canteens in our University could be beneficial to the canteen rendors.

## 1.2 Objective

Our system design comes from the famous application Openrice. It is a website that shows all the information, menu and comments of restaurants in Hong Kong. The system is divided into two parts, the frontend part which users could use and interact with the interface, and the backend part, where developers and servers deal with data transferal and management. Our goal is to provide a perfect Openrice system particularly for canteens in CUHK.

## 1.3 Expected Customers and Market

The targeted users of our system are customers (students, professors and staffs of the University) and canteens of CUHK. Every one who has an email account for registration would be welcomed to join the system as customers. Canteen accounts will be limited by giving several accounts and corresponding passwords when they sign a contract with the University.

## 1.4 System Features

Our system is mainly composed of 5 parts, including registration and login, latest news release, canteen ranking, canteen information and menu, commenting system. Canteens could update their menu information and release any news of their canteen on the system, whereas customers of our system could comment on the food provided and view others review on the menu. More details of these features will be included in later sections.

# 2. Background

Our group observe that many students from the Chinese University of Hong Kong are not familiar with the canteens we have in our University. Even in our group, it was the first time for most of us to find out that there are actually 35 restaurants in total in our school campus.

In the University, we could always find canteens near the main campus fully packed will long queues lining up outside the shop. It is not difficult to realize that the number of customers in particular canteens is significantly smaller. Some of the nice restaurants serving high quality dishes are not well known among students just because they are not located at where target customers usually pass by. In our opinion, this is somehow unfair to those restaurants, and missing out such precious places in CUHK is considered a loss for customers. Therefore, our team would like to take the first step to change the current situation.

Eating is one of the most important issues in our life. The problem of where to eat and what to eat becomes a concern of most of the people in our University every day. As we know, people decide where to eat base on multiple considerations such as location of restaurant, cost of dishes, taste of food provided and comments from other customers. In order to provide an all rounded service and information to customers, there must be a medium that includes all the required and present them to users.

In view of the above, our group borrows the idea from the famously used app called Openrice in Hong Kong, and we hope to create a CUHK version of it. The app can be used for canteens in CUHK for updating their menus and post announcements (e.g. discount periods, new product arrivals). While as for ordinary people, which are potential customers of the canteens, could view the information (e.g. location, menu, comments of dishes, cost of meals) of the restaurants through our system. By knowing more about the restaurants, hopefully our system can boost the motivation of customers and attract them to visit different canteens in CUHK. Thus, enhance the customer flow of these dining places.

## 3. Specification

We have divided our system into three main parts: User Login and Registration, Menu Display and News Updates, Commenting and Ranking System. This part of the report shows the data flow diagram and the description on the design specification of data flow and behavior of these three parts.

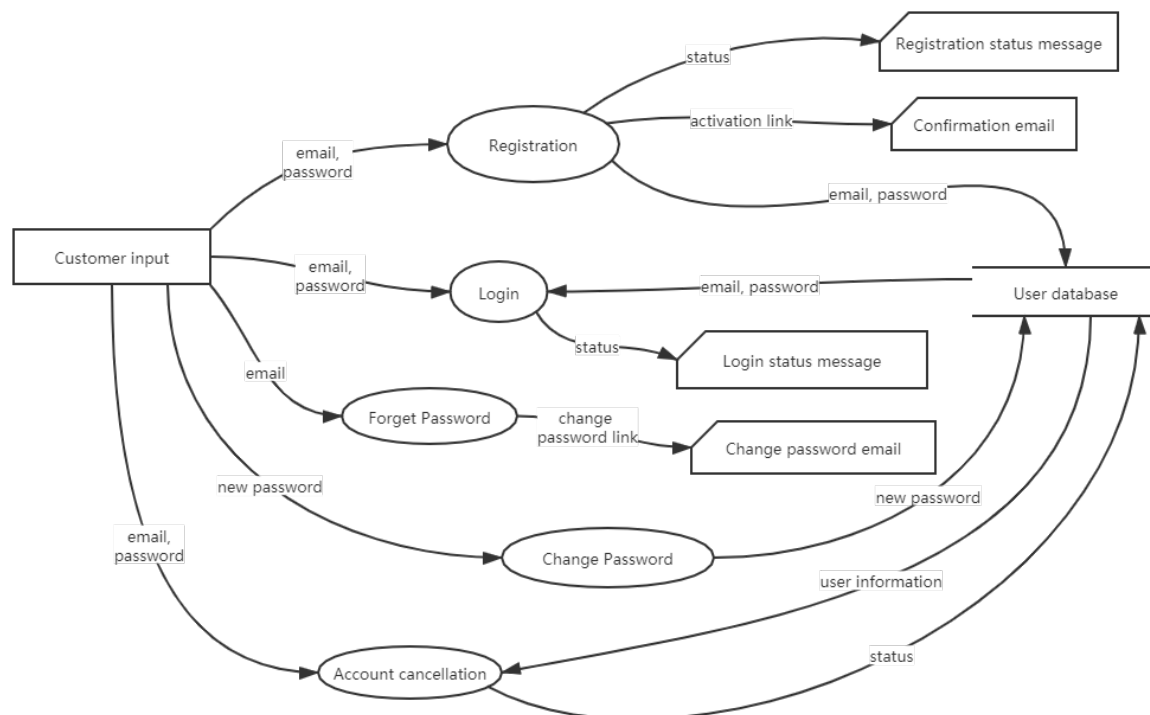## 3.1 User Login and Registration



Figure 1 Data Flow Diagram: User Login and Registration System

Users will have to go over a user registration process before they could use our system. The user registration function is mainly for the use of customers, as in our initial idea it is not necessary for every canteen staff to have an account. To create an account, we require users to enter their email and password. A format validation check will be applied to ensure that the email format is like"xxx@xxx.xxx". There would also be a data verification process that asks users to confirm their password by retyping in their desired password in another field. An email will be sent to the email account of the user afterwards and they are required to activate their account by clicking on the link provided in the confirmation email.

Users who did not login to the system could also view the contents of the pages. For those users who would like to post comments of menus, they will have to login by typing their email as username and their own password. If users have any login problems, our system provides a function for users to change their password. These users will be asked to type in their email for receiving an email, and in the email, they can click into a link to change password. By submitting the new password in the link, our database will update the user's data in the password field.

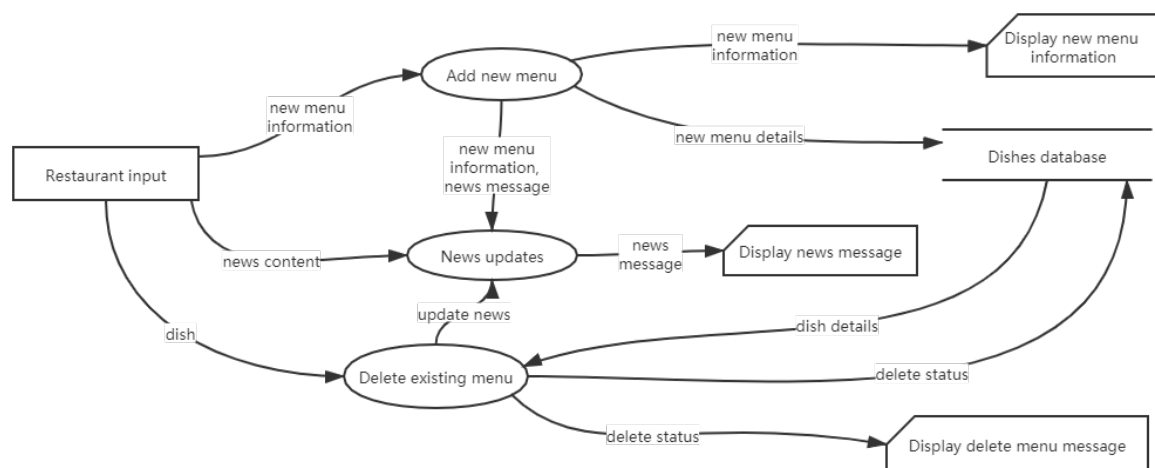## 3.2 Menu Display and News Updates



Figure 2 Data Flow Diagram: Menu Display and News Updates

Displaying the food served by the canteen will be a major concern of our system. Since what food a restaurant provides can affect the consumers' decision of visiting the place or not to a large extent. The menu of the canteens can only be updated by canteen accounts. After account login, they can add a new item to the existing food list by adding the title of the new product, cost and a photo of it. When adding a new item, our system would automatically generate a news item on the latest news page, stating that new items are already available for customers. We hope this process to be a real time process.

The latest news page provides users information about the activities and discount offered by CUHK canteens. There will be two parts related to news issues, one is the automatic scrolling advertisement consisting a banner or a photo, another one is a tag to the news page for viewing the text messages of the news. Pictures and messages for the news function will be stored in the database permanently. For the scrolling advertisement, the latest 8 news will

be shown. As for the text news page, the news will be sorted such that the latest news will be at a higher position on the page.

The addition of news into our system will mostly be done by the canteens. After canteens login to the system, they could add message that could be released to the public by providing an image and a text description. We advise restaurants to provide their an eyeball catching picture to attract their own customers, we will also offer default images for representing all kinds of news (e.g. discount, new product, product to be removed from menu) if the canteens do not have their own.

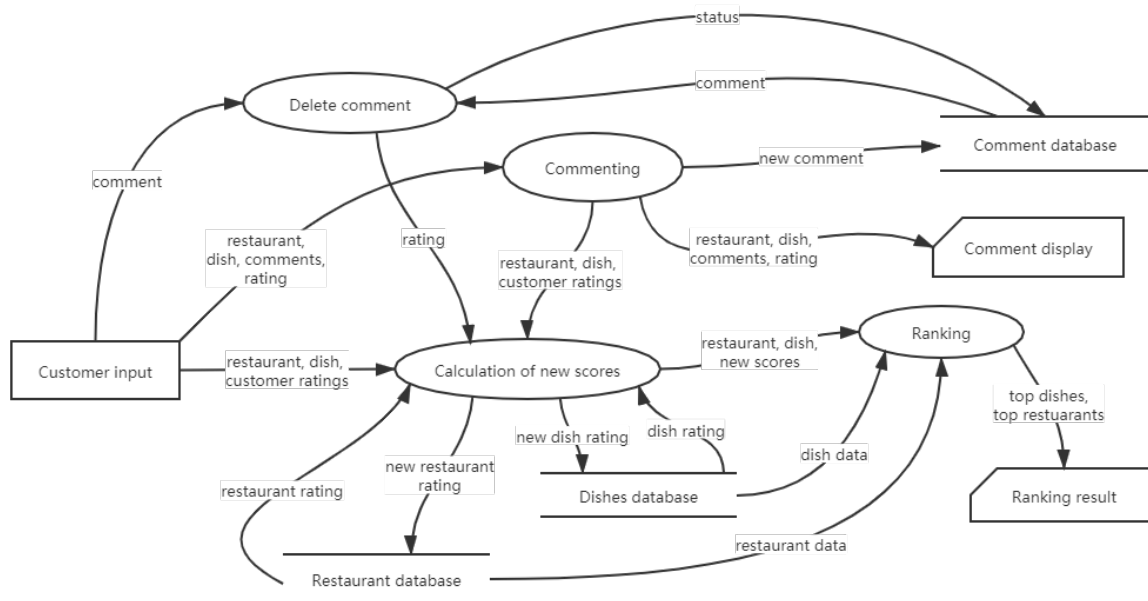## 3.3 Commenting and Ranking systems



Figure 3 Data Flow Diagram: Commenting and Ranking System

The commenting system will also be a major part of our system. This function is mainly used by customers of canteens. People can comment on food on the menu by clicking on the dish button and specify they have and experience of taking the food. The commenting system allows users to upload photos of their meals to our server, it can also store the discussion text statements written by the customers. This system can also help facilitating the ranking system, as every time when users are writing a food review, our system will require them to rate on the taste of the food, the environment and the cost of the food as mandatory fields. Users are also welcomed to discuss on other optional issues, such as attitude of canteen staffs, tidiness or cleanliness, willingness of revisiting the canteen and takeaway services etc.

For the ranking system, there will be two types of ranks: rank of restaurant and rank of dishes. In our system we allow customers to rate and comment on restaurants. We are planning to come up with a score calculating mechanism to compute the general rating of the canteen by taking the above criteria in account. The rank of restaurant shows the most favorable canteen with the highest rating points as commented from customers. The rank of dishes would be considering all menus from the 35 canteens in school, and the food of highest popularity would be listed.

The rankings would be updated by us at the backend constantly using batch processing. Every time when a customer comments and rates the canteen, our system will have to store the corresponding points and reviews, so as to carry out calculations and sorting afterwards.

## 4. System Architecture

This part describes the system architecture of our system. Architecture diagram of the whole system and UMLs of different components will be presented together with the text explanation of our design. We would also briefly discuss what platform and language we are planning to use during the development stage of our system.
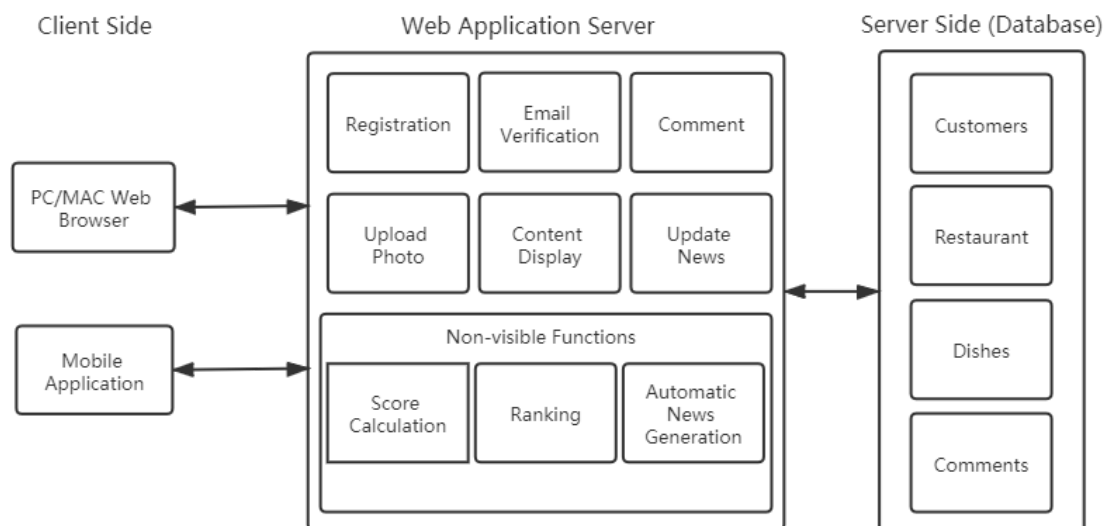
## 4.1 Architecture Diagram



Figure 4 Architecture Diagram

The above figure describes the overall architecture design of our system. In the frontend part, users can get access to our system using a web browser or a mobile. With our user interface the customer users will be able to use functions such as registration, email verification, comment, upload photos and also view contents, whereas restaurants will also be able to add new information to be displayed in our system. Some functions are for facilitating some features of our system will be hidden from users. These functions mainly require some calculations and also logic controls of some data. For the server side, there will be 4 tables in the database storing the information of customers, restaurants, dishes and also comments of the dishes respectively.

## 4.2 System Components

### 4.2.1 Frontend Development

Our expected product will be a website, composed of a html, CSS, Javascript files. We are planning to use the Bootstrap framework to assist us with the development of our website. We expect the user interface to be user friendly, where information are clear to users.

### 4.2.2 Functions

As stated in part 3 of this design document, there are three major functions:

- User Login and Registration

  This function requires a database to store the input information from the users. Customers account can be freely registered while each restaurant in the Chinese University of Hong Kong will be provide an official account. Email should be sent automatically to new customers in order to do confirmation of the registration process.

- Menu Display and News Updates

  This function is the main part of displaying information in our system. The customers can view dish, restaurant information, comments and news updates of the canteens through our website. The part requires the system to retrieve data from the database as well as update any changes to previous records.

- Commenting and Ranking

  This part does calculations on the ratings of dishes provided by the restaurants. It also controls the logic of ranking restaurants and dishes according to the feedback from customers of the canteens. A frequent communication is needed in between different tables in the database. A sequence flow of how the dish can be commented will be shown in figure 7 of section 4.3 of this design document.

The overview of functions will be shown in figure 6 of section 4.3 of this design document.

### 4.2.3 Middleware and Communications

As mentioned in 4.2.2, it is obvious that all the functions are highly related to the usage of database. In order to achieve our desired result, we plan to adapt a way for the frontend to communicate with the backend. We plan to use the GET and POST methods of Ajax to do the request and respond procedure, so that we can pass data from the user interface to the databases and vice versa.

### 4.2.3 Backend Development

The backend part mainly focuses on the management of the database. In our initial design, we are planning to have around four tables, namely customers, restaurants, dishes and comments, where customers and restaurants both belongs to the user class. The detailed fields and related methods will be shown in figure 5 of section 3.4 of this design document. As for the database we are now considering using one of the following:

- Redis

  It offers a framework for building in memory application that is at the same time versatile and scalable. By using RAM as the memory, Redis speeds up the query of data. Moreover, Redis is a multi-utility tool and can be used in use cases such as caching, messaging-queues, any short-lived data in your application (e.g. web application sessions, web page hit counts).

- MySQL

  It is a secure and reliable database management system. MySQL has high performance designed to meet the demanding applications while ensuring optimum speed, full-text indexes and unique memory caches.

- Microsoft Access

  It is extremely user friendly in terms of having various wizards to speed up the process of creating a database application. This greatly reduces the development time. Also, Microsoft Access is well-known, popular for software developers, meaning that there will be enough technical support and resources.

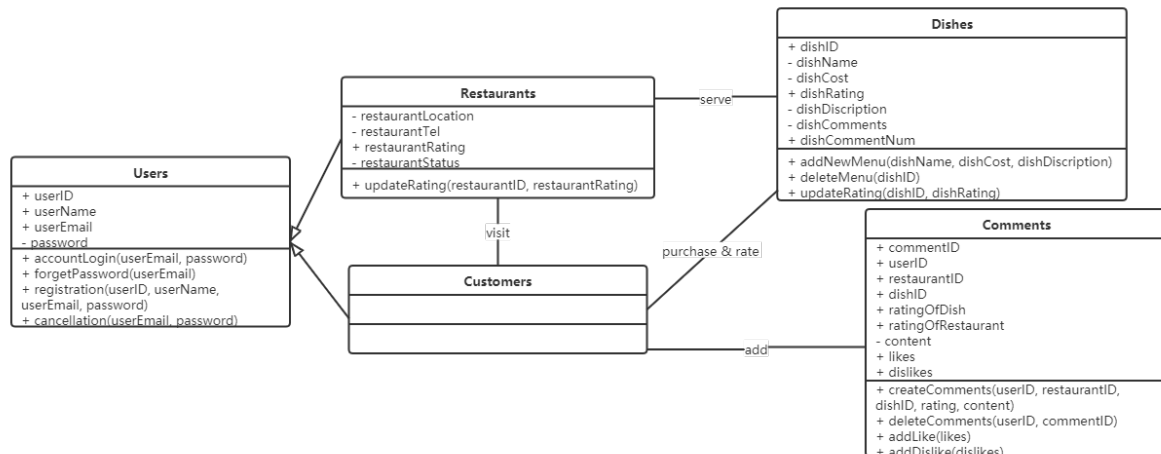## 4.3 Description of Major System Components by UML



Figure 5 Class Diagram

The above class diagram shows the relations between the classes in our software development design process. It is also basically all the classes we plan to have in the database. Customers and Restaurants are accounts that inherit the Users accounts. Customer is purely an ordinary users account whereas Restaurants have a few added fields specifying some details of the canteen. All dishes (no matter they are served by the same canteen or not) will be stored in the Dishes table with an unique dish ID as the primary key. Comments of all dishes will be stored as well in a separate table in the database.
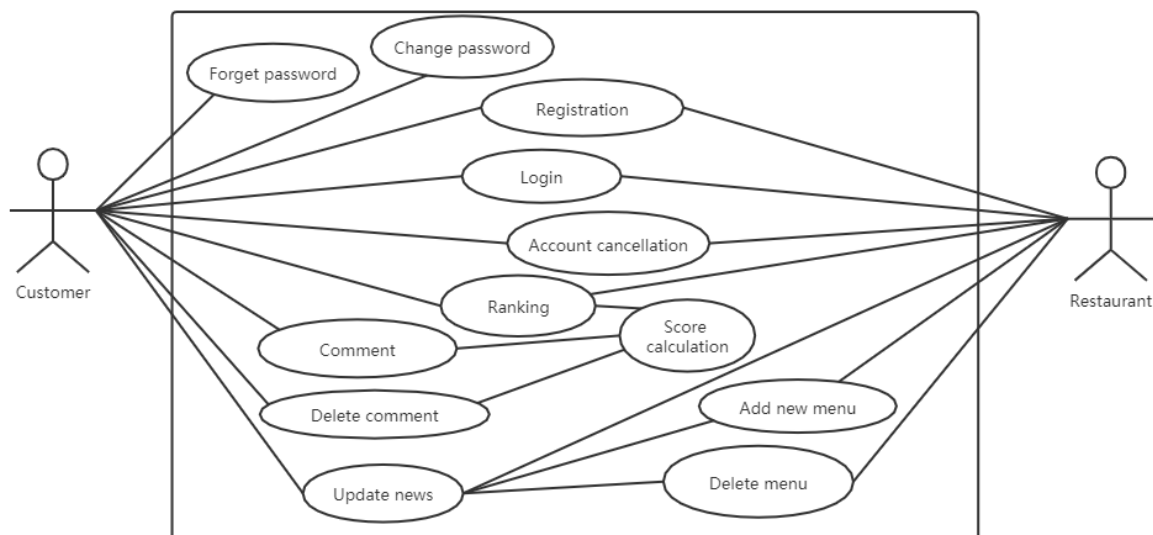


Figure 6 Use Case Diagram

Figure 6 is a use case diagram showing all the functions involved and their relations with end users. The functions that can be directly be used by an end user will be linked to the method. As seen in the diagram, both customers and restaurants can register, login and do account cancellation. Note that these parts of the restaurant account will be done by the management team of the system in order to ensure that each restaurant has an official account. One may notice that some functions (such as the score calculation) function is not directly in touch with the users, stating that the function will be carried by program instead of triggered by user input or interactions.
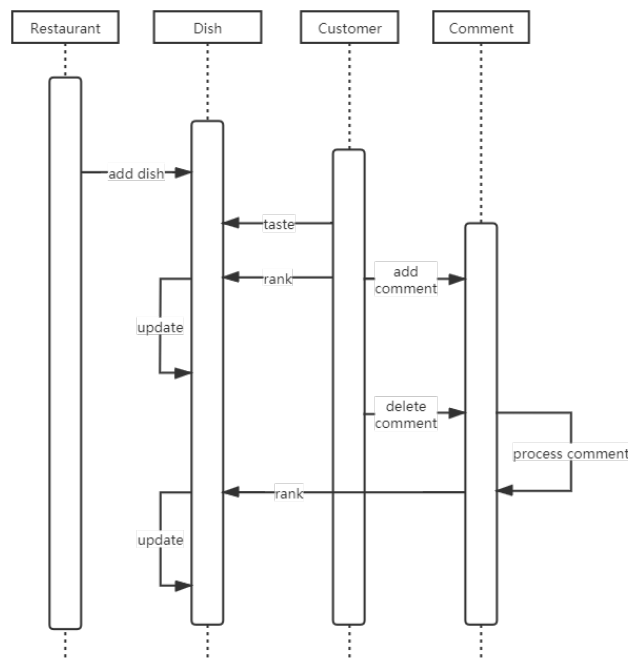


Figure 7 Sequence Diagram: Comment and Ranking

The above sequence diagram describes the full sequence of how a dish will be commented and updates its rating. Firstly, the restaurant added the dish to the Dish database. This step is also counted because a user can only choose to rate and comment on an existing dish, according to our design. After the Customers try the dish, the customer can choose to add comments and also rank the dish. When the comment is submitted, the original rating score will have to be taken out from the Dish database to update its value. We are now designing a formula to fairly adjust the ratio in between the new rating score and the existing data. The second part where we also have to update the dish rating is when the customer manually deletes the comment on the dish. When the customer deletes the comment, a query will be used in the Comment database to retrieve the corresponding record and getting the rating of the comment. With the rating, another formula will be used to update again the dish rating value and store it back into the Dish database.

## 5 References

https://www.quora.com/What-are-the-benefits-of-Redis
https://www.datamation.com/storage/8-major-advantages-of-using-mysql.html
https://www.software-matters.co.uk/microsoft-access-benefits.html