# Team work

**GitHub Link**: <https://github.com/lvh980106/CP3407_Assignment1>

Student 1: Lee Vern Hoe, Scrum Master

Student 2: Joel Lim, Member

# Project Description

**Problem statement:** Tourists are facing difficulties in buying ticket online due to the complex procedures and difficulties of going to the actual location to buy tickets.

**Solution:** Focusing on our customer’s convenience, we provide 2 easy methods for tourists hoping to efficiently travel around Singapore and take in the sights and sounds with ease. They can opt to use either the machines set up at every MRT station or download the online application onto their mobile phones to purchase tickets and get a brief synopsis about the places they would like to visit. These 2 platforms will provide similar functions, and can both be used to purchase tickets, but only the machine can print a physical ticket.

In the past there were long queues especially during peak hours or holidays to buy a ticket to enter. Now with the implementation of the online system, anyone can book a ticket and simply scan it to go in the moment they arrive, saving time and effort. It also allows the organizations to limit the number of people at one time by setting a cap on the amount of tickets to be sold, allowing patrons to know when the attraction is full and not waste their time making the trip down only to be turned away.

Losing physical tickets have also been a known issue in the past. Now, with the automated backups on the cloud servers, even if the digital ticket was misplaced they can simply check their email to get a backup and return to the machine and type in their email to receive a new physical copy.

The machine that we will implement will be placed at every MRT station across Singapore as most tourists will most likely be encouraged to use public transport due to the exorbitant prices of cabs and Grabs. As new tourists come into Singapore, they will want to experience the life of a Singaporean and take the train around. The kiosk will be big and noticeable and situated beside the train ticketing machines so that they will be aware of it and approach it for advice on how best to explore Singapore.

The mobile app can easily be downloaded from both the Android PlayStore and the IOS AppStore. Any potential tourists can search for it and download it from their own countries and read up on what Singapore is like and what we have to offer. This saves time as they do not need to do that while in Singapore and can plan ahead on what to do while here. They can also buy their tickets and if they want a physical copy, simply need to head to the nearest station, enter their email and a physical ticket will be printed for them.

Another reason why digital tickets are more useful in today’s world is to reduce ticket fraud. Ticket fraud has been commonplace since ticketing systems began and while not prevalent in Singapore still occurs occasionally, such as the recent Universal Studios scam early this year. By controlling ticketing and having specific ID’s with easy tracking with a computer, we can prevent ticket scams and fraud from occurring as much as possible.

While most tourist attractions are setting up web applications and online ticketing of their own, there is no centralized platform to do this and we aim to be the first. By being a consolidated hub of information regarding anything that tourists require, we provide convenience and a better experience for the tourism industry of Singapore. Most of these web applications also do not fully support mobile devices thus by having a mobile app we aim to overcome that disadvantage as in the modern world a mobile device is the go-to for everyone to do anything on, slowly making computers obsolete.

Stat Counter reported on 2016 October that more user around the world accessed the internet from mobile devices than from desktop computers. And in the future, mobile commerce will cover 45% of total e-commerce which means almost half of the population will start to make purchases online using their mobile device. Mobile device integration is necessary nowadays, the only possible blocker being the internet service in different countries. As mobile data service charges are still incredibly expensive in separate regions and not all tourists would like to purchase a data plan when travelling, Wi-Fi is needed and thus email is our chosen informing platform as emails will be checked occasionally by most working professionals who travel for relaxation.

**Technical aspects:**

The kiosk and mobile app will show roughly the same layout and hold the same content, with some specific content catered to each platform. They both offer the option to buy a ticket through a GUI that consists of pictures which leads to a brief description of the tourist attraction. This then allows the user to buy a ticket with an order form and payment will be done digitally through the app and through either card or cash for the kiosk. The kiosk then prints out the ticket and both will send an email to the user containing the digital copy, as the kiosk will ask for the email address of the user as well.

Asides from purchasing tickets, the kiosk offers the option to immediately print a ticket that was already bought through the mobile app by entering the corresponding email address. Once entered, the patron can choose to print it out in the event that a physical copy is preferred. The printing of tickets will be done through a device similar to an ATM receipt printer, with constant need to refresh the specialized paper and ink.

The online app offers the option to refund the ticket even if bought from the machine. By using the email address and logging in, the user can request a refund. This refund is only available for 24 hours from the purchasing moment and only refunds 80% of the original buying price. If paid through an online service such as PayPal, we will liaison with the company and get a refund from them as well.

Both the kiosk and the mobile app requires connection to the internet. While not shown on the front end, when performing any actions, the background server must retrieve information. This cloud server acts as a middleman between the database and the interface. The database server will be setup and contains the history logs, live transactions and keeps track of all events that happens.

# Project Planning and Scope

**Available days: 40 days**

**User Story 1**: Login page (Single sign-on) - 4 days

*It is so troublesome to purchase online, we need to register so many accounts for different websites.*

**User Story 2:** Registration – 3 days

*I need an easy way to register for this app.*

**User Story 3**: Description of attractions – 3 days

*It will be nice if we can have an overview of places we can go, so that we can just pick and see the details of it*

**User Story 4**: Ticketing platform – 5 days

*Where can I buy tickets to national zoo, museum, universal studio and other places? It is so far to travel, and the queue is so long.*

**User Story 5:** Guide – 3 days

*I can’t plan my travel plan properly, I need a guide to optimize my planning for travelling around Singapore.*

**User Story 6**: Refund, 1-day grace period to be refunded 80% of the cost – 5 days

*I got the wrong ticket, I want to refund my ticket…*

**User Story 7**: Advertising – 3 days

*We just arrive Singapore, we are backpackers, we want to know where the tourist spot of Singapore is.*

**User Story 8:** Ticket softcopy (email) – 3 days

*I lost my physical ticket, where can I get a backup?*

**User Story 9:** Ticket printing (kiosk) – 2 days

*I would like a physical copy of the ticket, but I ordered through the app.*

**User Story 10:** Multiple payment methods – 5 days

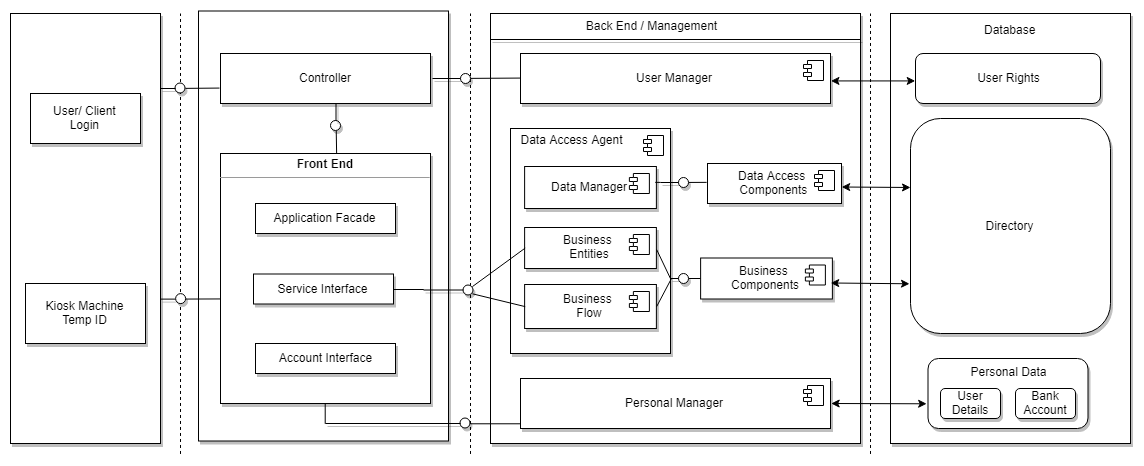
*Sometimes I want to use specific credit cards to pay for the tickets.*

**User Story 11:** Android and IOS integration – 4 days

*I use an Android, but my wife uses an iPhone, we both need the app.*

# Project Design

**Architecture Design:**

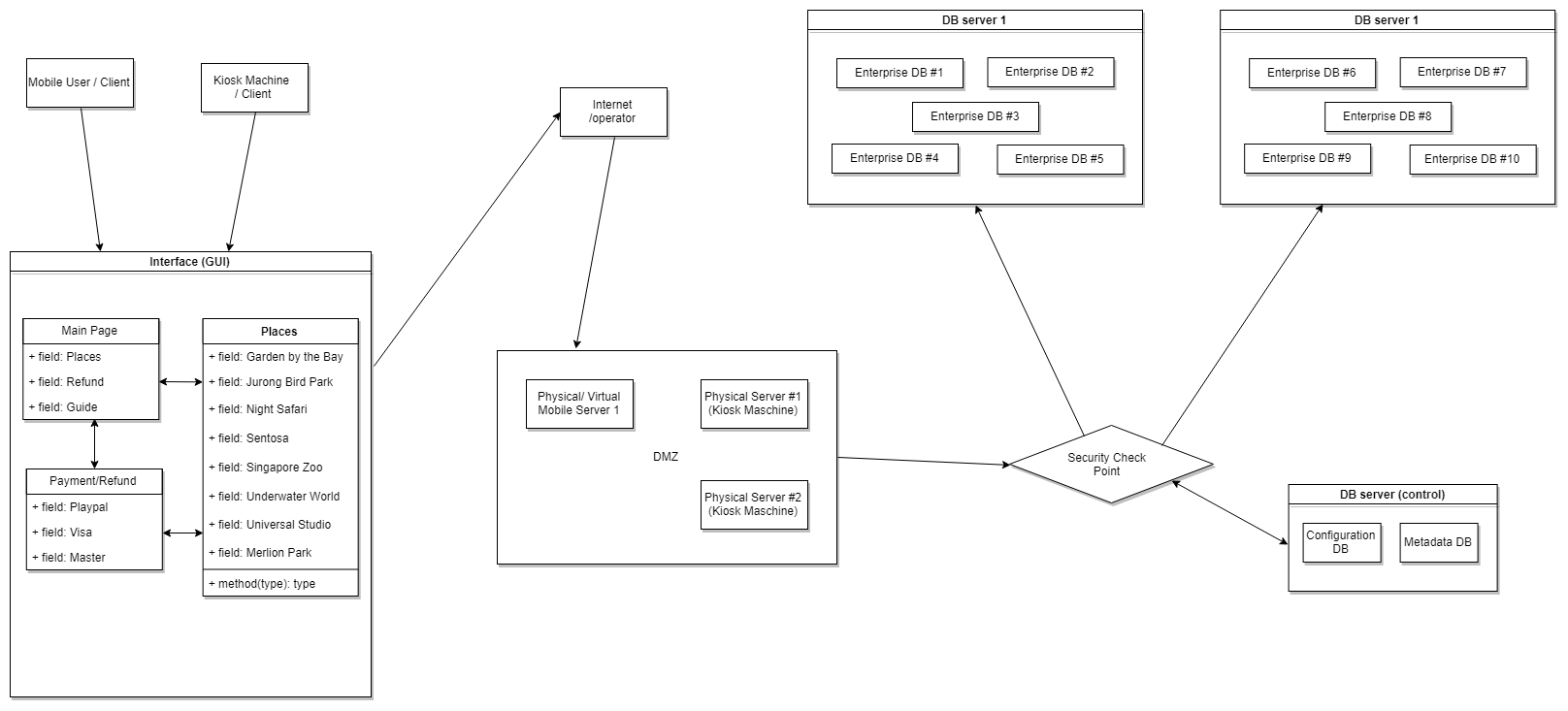


Architectural design is the process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system. The architecture design of our application is a client – server architecture, the functionality of the system to let the users to purchase tickets from a kiosk or our application. After the user login into the system, user is allowed to make changes in the front-end layer.

The system is designed to process user’s input to produce output, meaning users are requested to provide input at the interface to get the desired output. At the front-end layer, users are allowed to manipulate variables such as choosing the different services, selecting different ticket to purchase which directly interacts with the back end and databases. Clients are allowed to manipulate databases with rules and restriction, these rules will be managed in the management layer of the architecture before and changes are made in the database. Different user is given different rights such as admins are given access to change user booking details while normal users are only allowed to purchase and browse.

From the back-end layer, before any information is displayed to the user, data is retrieved from the database and processed in the management later before displaying to the user.

**Database Design:**

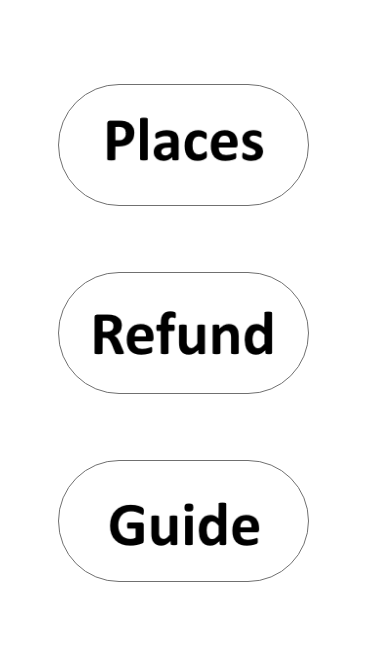
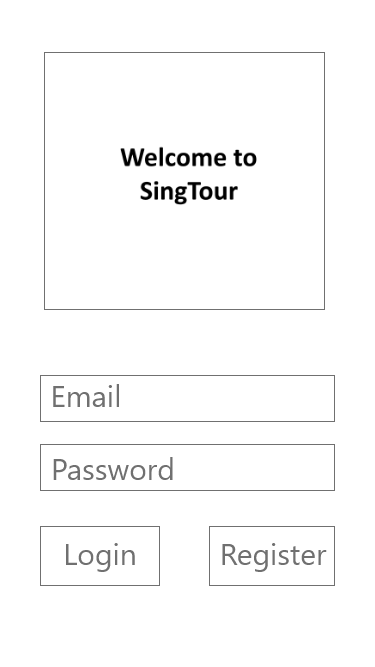


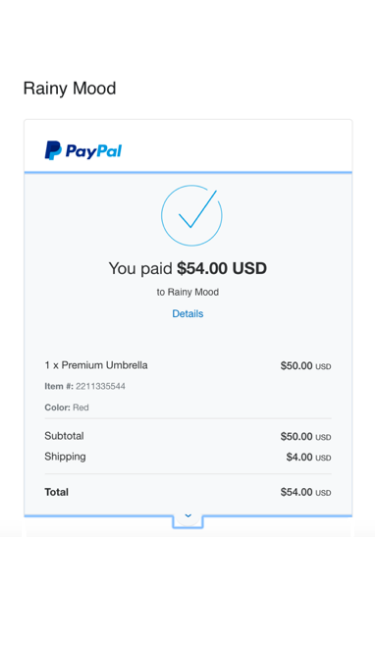
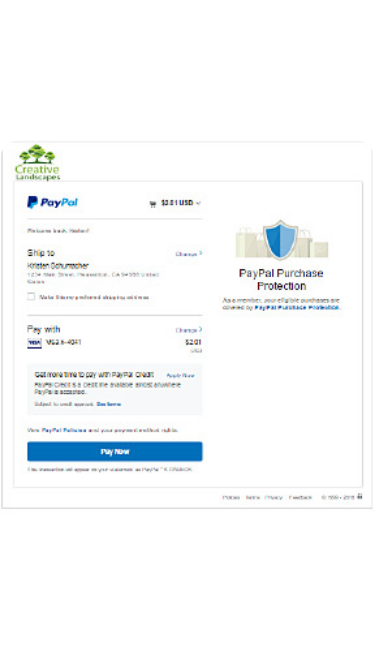
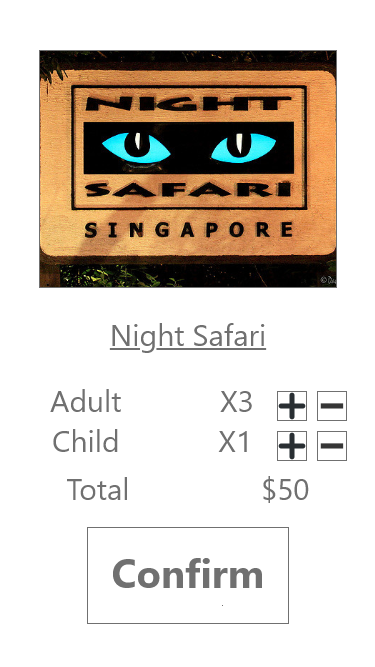
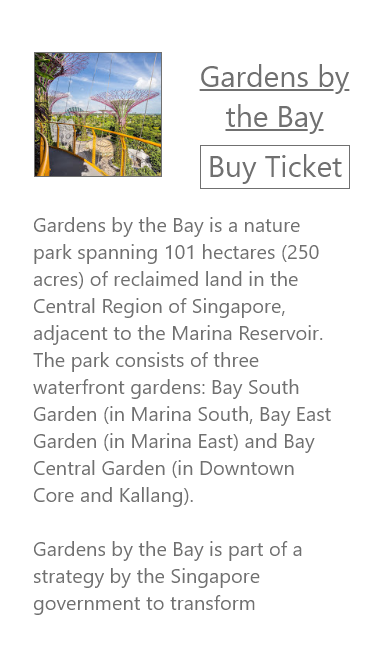
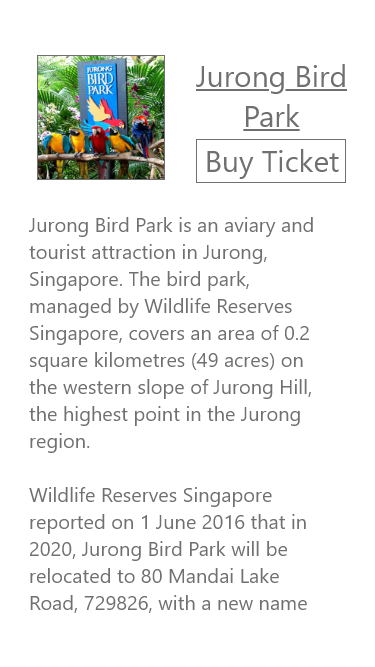
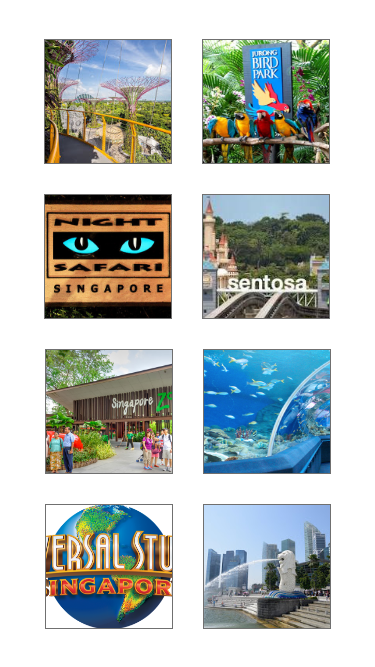
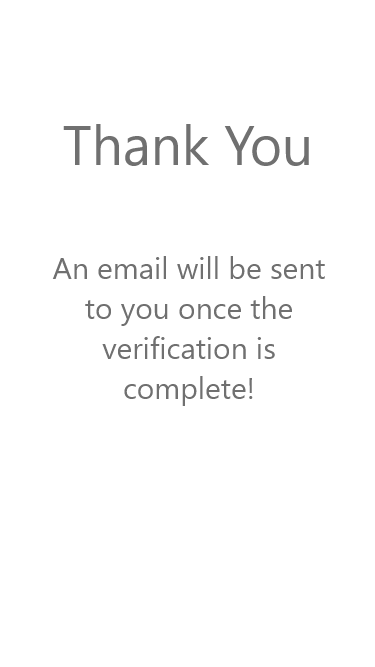
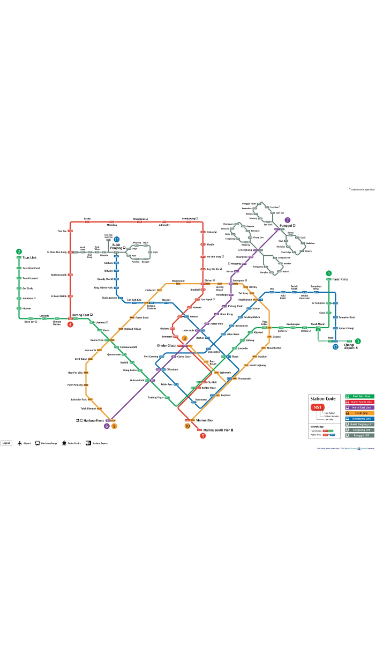
The database design of our system is divided into 2, due to the big number of databases from different enterprise, we have to divide our database into 2 servers and allocate databases equally in each server to allow further expansion.

When the client is accessing the interface, any command or changes to the database can only be made with internet connection, the client will access to different server based on the device they use in the DMZ, After the client login into the server, they will have to go through the security check point before accessing to the database. The security checkpoint and database control server will exchange information, configure data in the control server to the security checkpoint when the there is information requested from the security check point.

The enterprise database can be access after going through the security check point and changes can be made.

**Interface design:**





# Project development and release ICT Infrastructure.