WinWin

Milestone 4 **Design Phase** Presentation
By G14 Waterfall

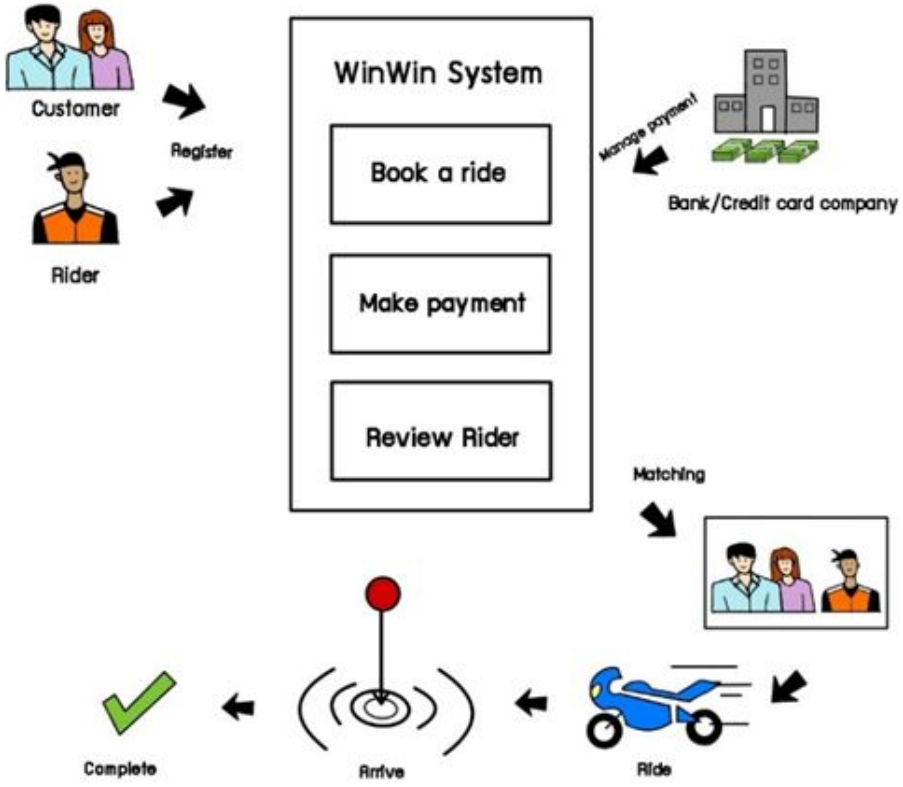
Contents

Introduction

Introduction

- Motorcycle taxi is the **second most used transportation in Bangkok**, besides MRT.
- Other Ride-hailing applications reduced **customer count of motorcycle taxis.**
- **Digitalization** of the motorcycle taxi system.

To-Be System


Customer
Register
Rider
WinWin System
Book a ride
Make payment
Review Rider
Manage payment
Bank/Credit card company
Matching
Complete
Arrive
Ride

System Design

- Overview
- Diagrams
- User Interface Design

System Design Overview
: Design Criteria

Coupling

Interaction Coupling
Dependencies between classes were considerably minimized with only data type interaction at most

Inheritance Coupling
Class diagram doesn't have generalization/specialization so there's no inheritance coupling required

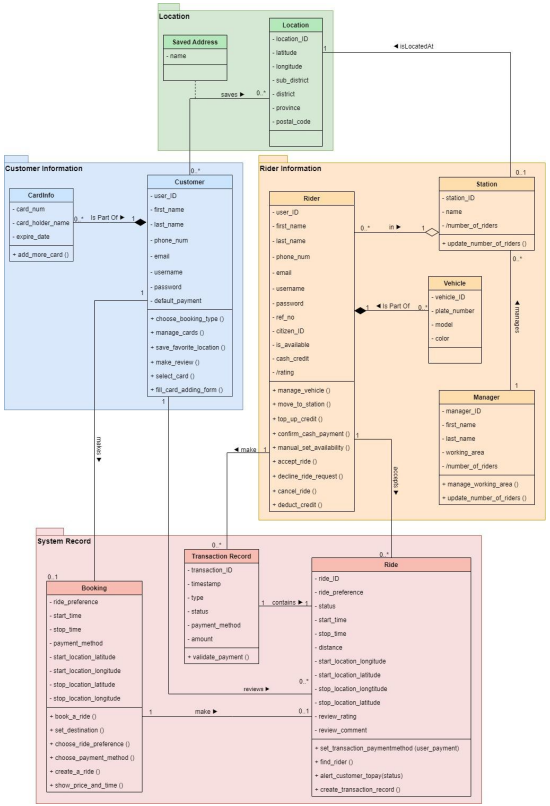
Cohesion

Method Cohesion
made to perform only one purpose

Class Cohesion
made to represent only one thing, with all the attributes and methods necessary to fully define the thing

Generalization Cohesion
As our design does not have classes in a hierarchy, this issue is redundant.

Class diagram with package



System Design Overview
: Design Activities

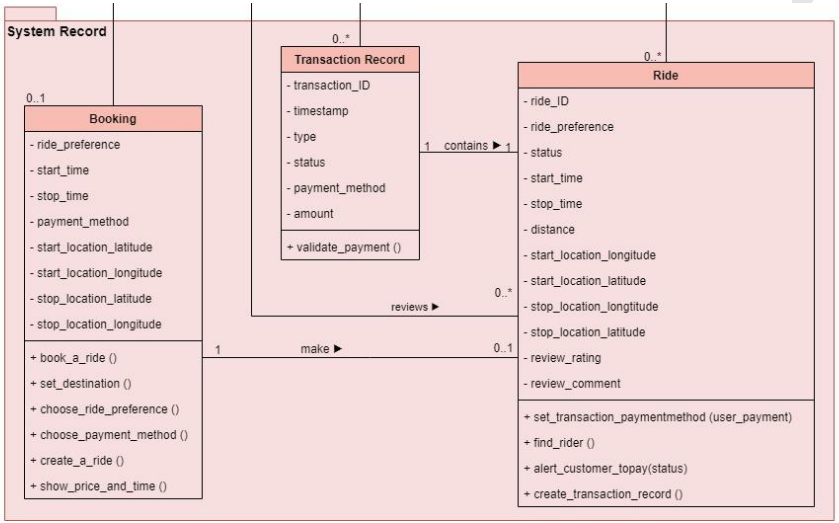
Adding specifications

Restructuring the Design

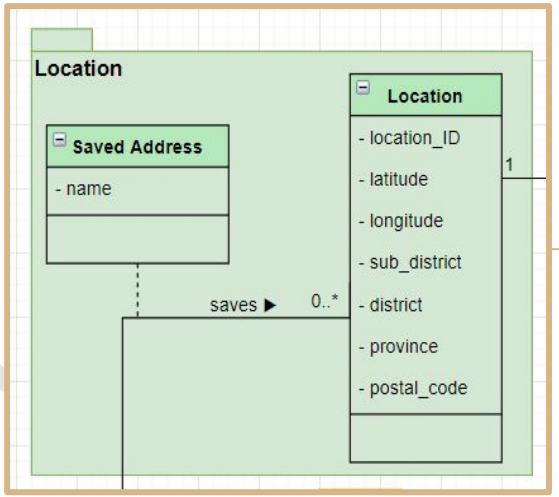
- **Factoring**: separating “Booking” class from “Ride” class
- **Normalization** : convert the association class (“Saved Address” class) to a normal class.

Optimizing the Design

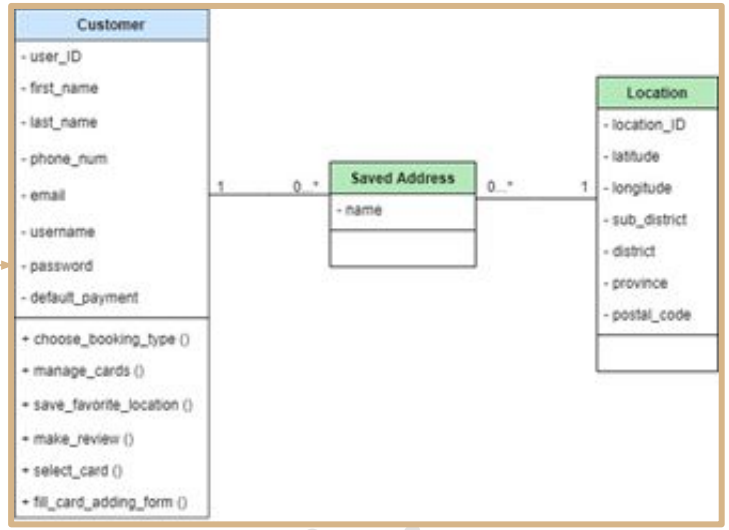
- Avoid re-computation by creating derived attributes and triggers:
  “Rider” class has a derived attribute ‘rating’



System Design Overview
: Design Activities



normalization



System Design Overview

: System Design Constraints

Operational

mobile application
automatically back up

Performance

respond in less than 5 seconds

Security

authenticate users
transactions confidential

Usability

easy to use

Cultural & political influence

strictly only in the Bangkok
displayed in Thai language

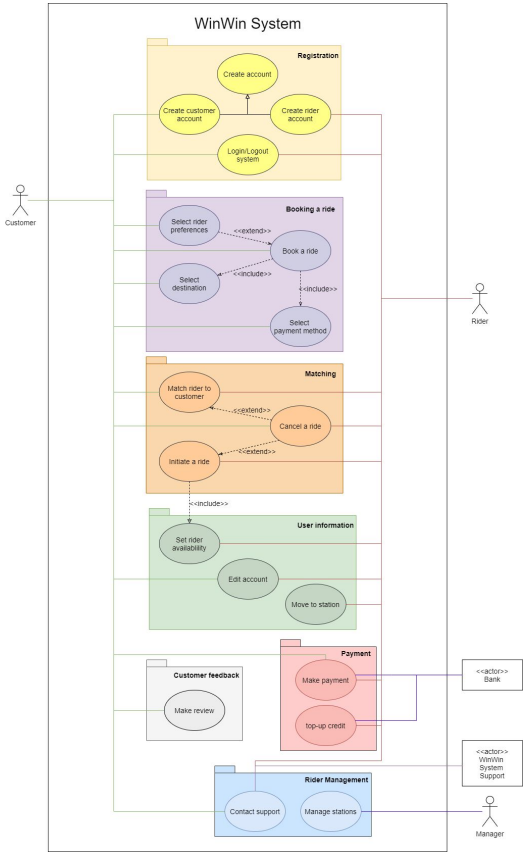
Legal implications

comply with the following acts

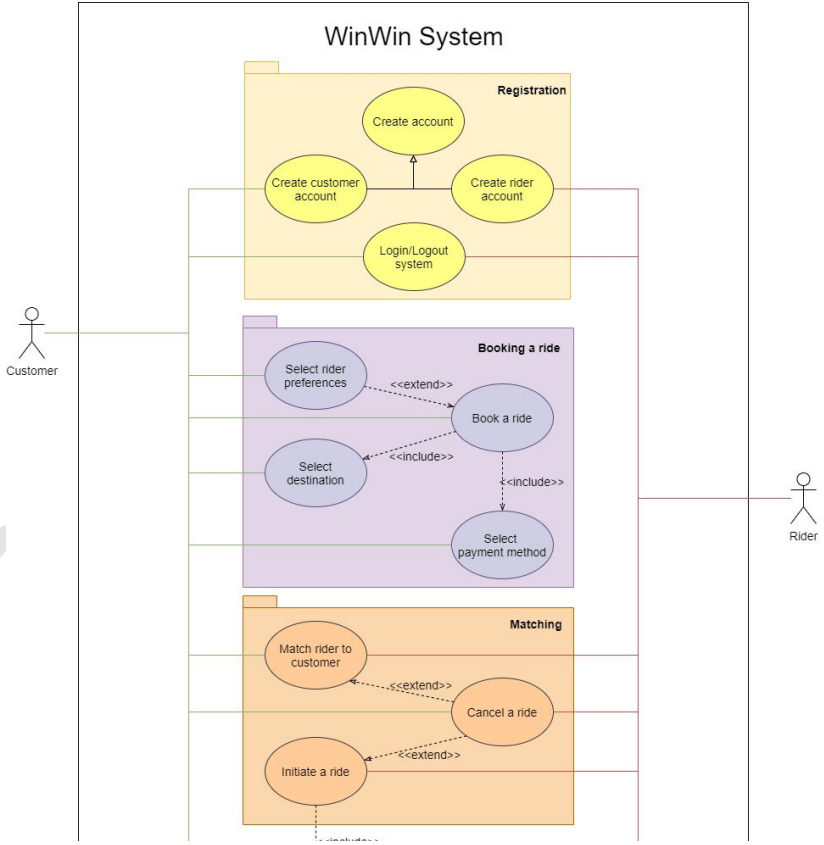
Diagrams

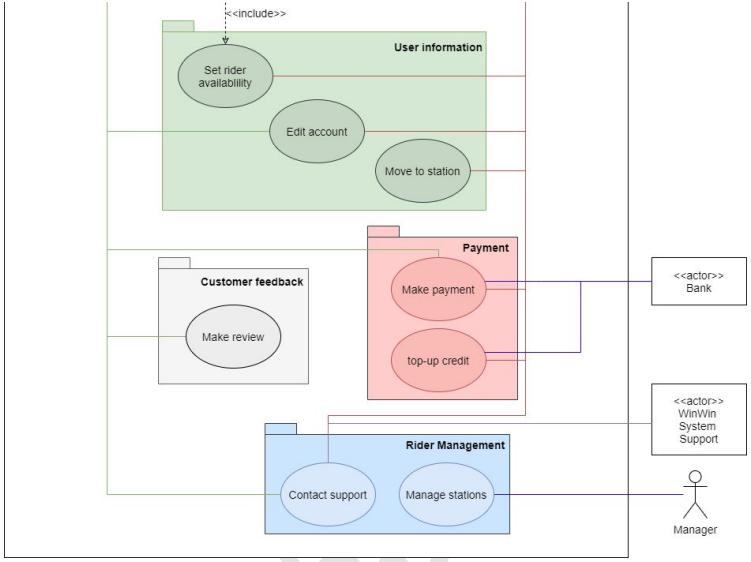
- Use case diagram
- Class diagram
- Component diagram
- CRUDE Matrix
- Method specification

Use case diagram with package

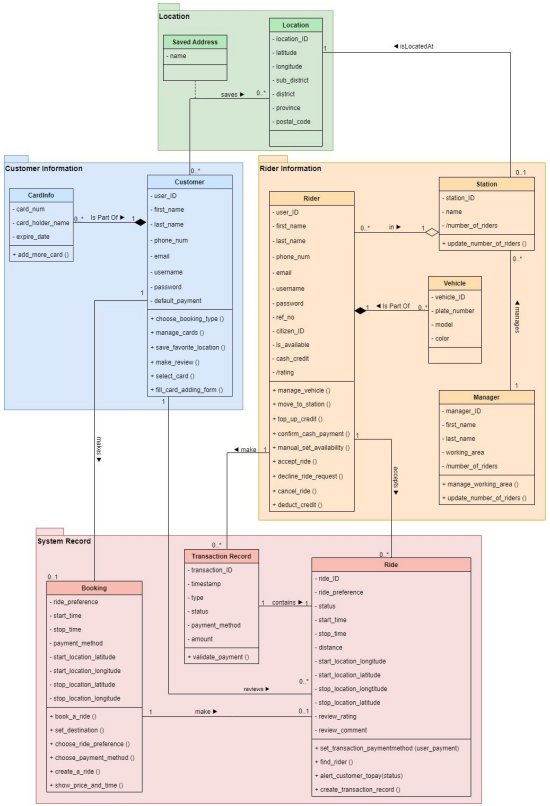


Use case diagram with package

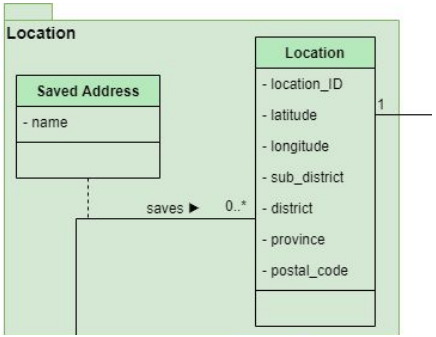


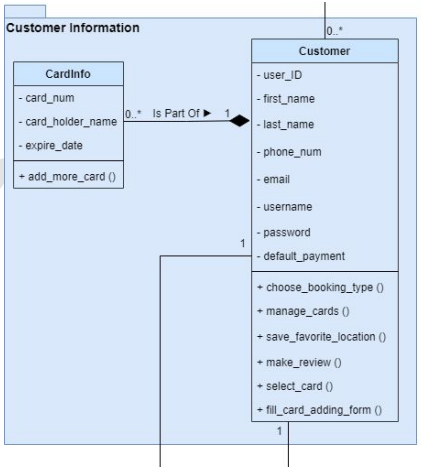


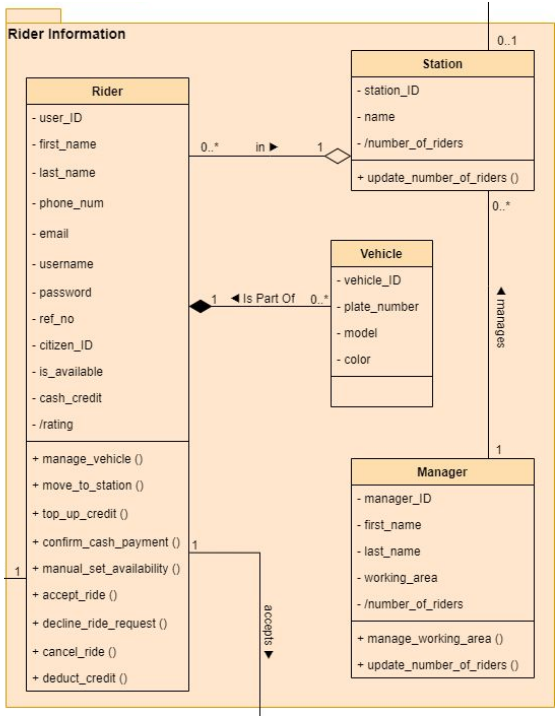
Class diagram with package

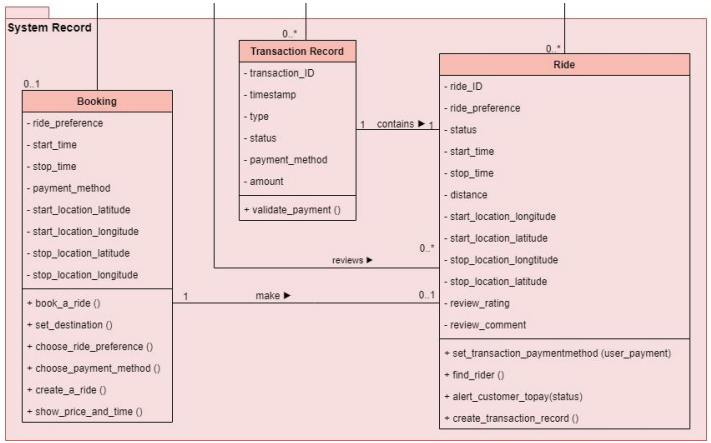


Class diagram with package

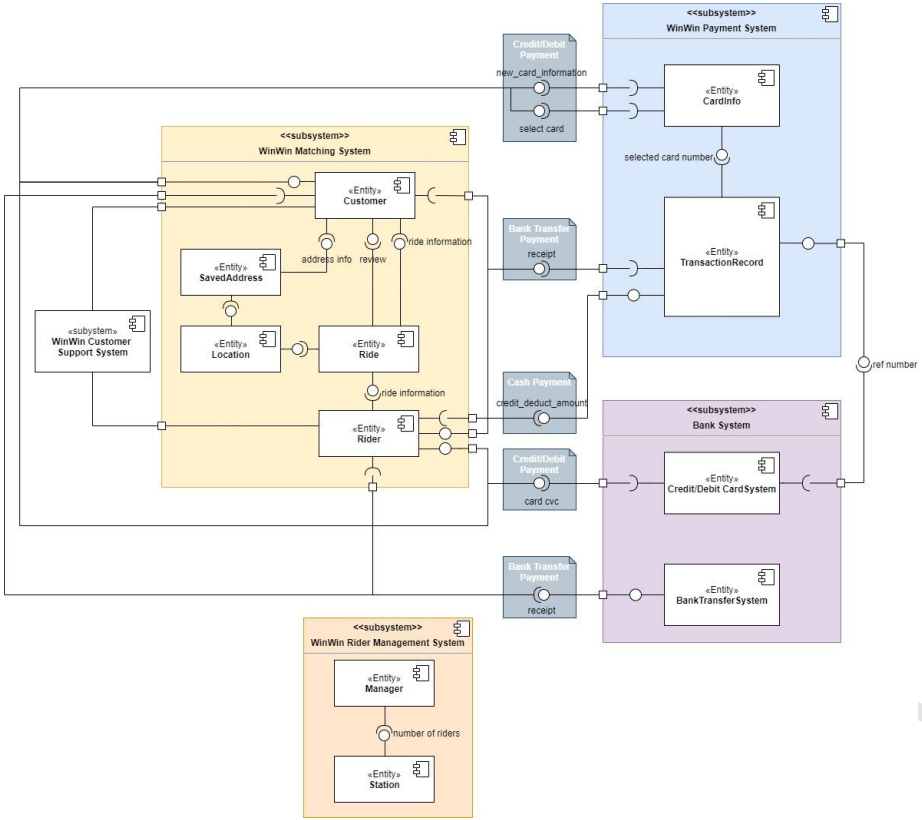








Component diagram



CRUDE Matrix : Book a ride

C: Create R: Read U: Update D: Delete E: Execute

Book a Ride	Customer Actor	Customer Class	SavedAddress Class	Location Class	Ride Class	Booking Class
Customer Actor		E	C R	C R		C R U D E
Customer Class			C R	C R		
SavedAddress Class				R		
Location Class						
Ride Class		R		R		R
Booking Class		R	R	R	C E	E

Method Specification

: Book a ride

Method Name: book_a_ride	**Class Name:** Booking	**ID:** BKG01
Contract ID: 001	**Programmer:** Alex	**Due Date:** 13 Nov 2021
Programming Language: Javascript		
Triggers/Events: Customer books a ride.		
Arguments Received: **Data Type:**	**Notes:**	
void		
Messages Sent & Arguments Passed: **ClassName.MethodName:**	**Argument:Data Type**	**Return Data Type**
booking.new ()	char(8) of customerID	Booking
self.choose_booking_type ()		
self.choose_ride_preference ()		
self.choose_payment_method ()		
self.set_destination ()	pair (double, double) of Latitude and Longitude	
booking.create_a_ride()		Ride
Arguments Returned: **Data Type:**	**Notes:**	
void		
Algorithm Specification: Create new Booking with customer_id choose_booking_type () choose_ride_preference () choose_payment_method () set_destination (latitude, longitude) newBooking.create_a_ride ()		
Misc. Notes:	None	

Method Specification
: Create Transaction Record

Method Name: create_transaction_record	**Class Name:** Ride	**ID:** RIDE01
Contract ID: 002	**Programmer:** Alex	**Due Date:** 13 Nov 2021
Programming Language: JavaScript		
Triggers/Events: Customer confirms booking.		
Arguments Received: **Data Type:**	**Notes:**	
void		
Messages Sent & Arguments Passed: **ClassName.MethodName:**	**Argument:Data Type**	**Return Data Type**
TransactionRecord.new ()	char(8) of ride_ID	TransactionRecord
self.set_transaction_paymentmethod ()	varchar(20) of user_payment	
Arguments Returned: **Data Type:**	**Notes:**	
void		
Algorithm Specification: create new TransactionRecord with ride_id set_transaction_paymentmethod (user_payment)		
Misc. Notes:	None	

User Interface Design

- design principles
- designs

Move to Figma →

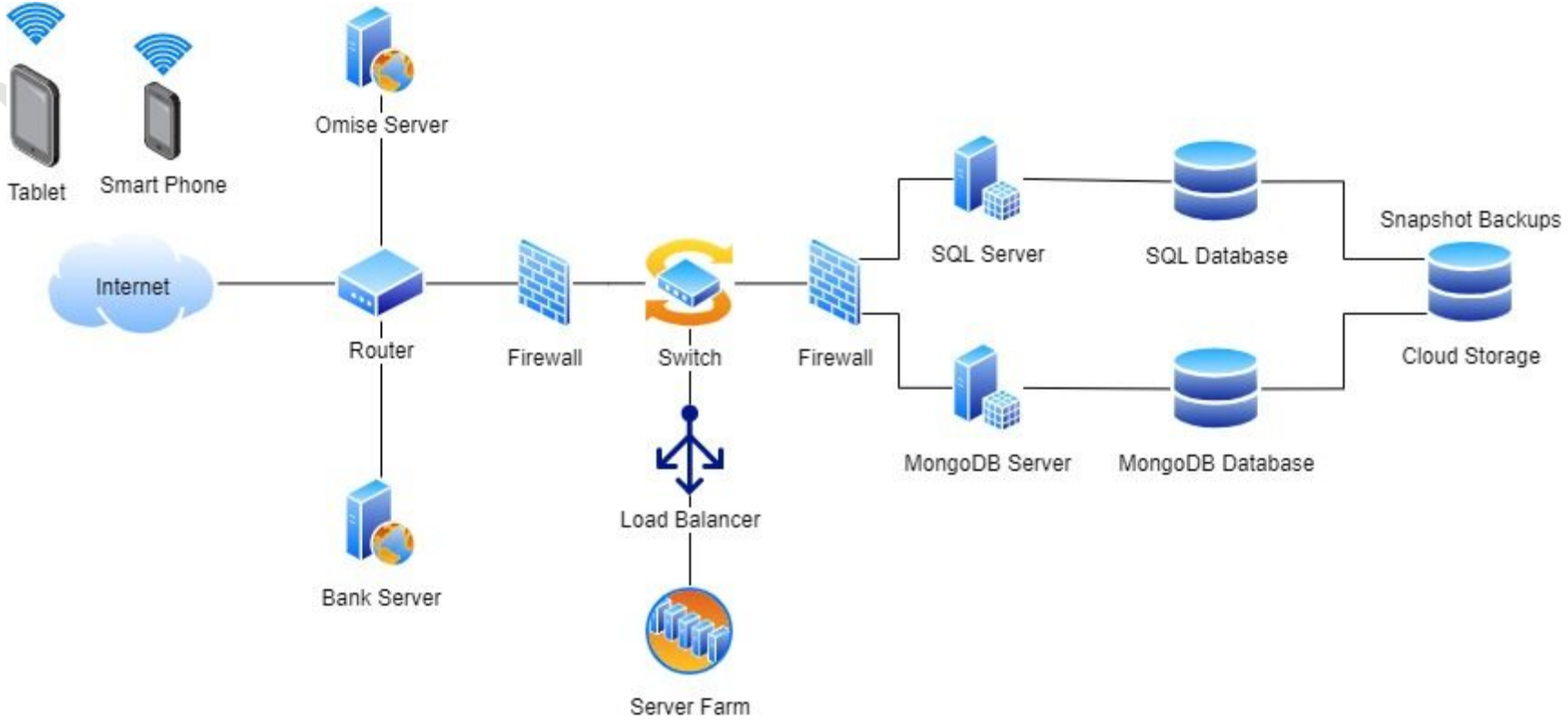
System Infrastructure

- Overview
- Hardware and software specification
- The impact

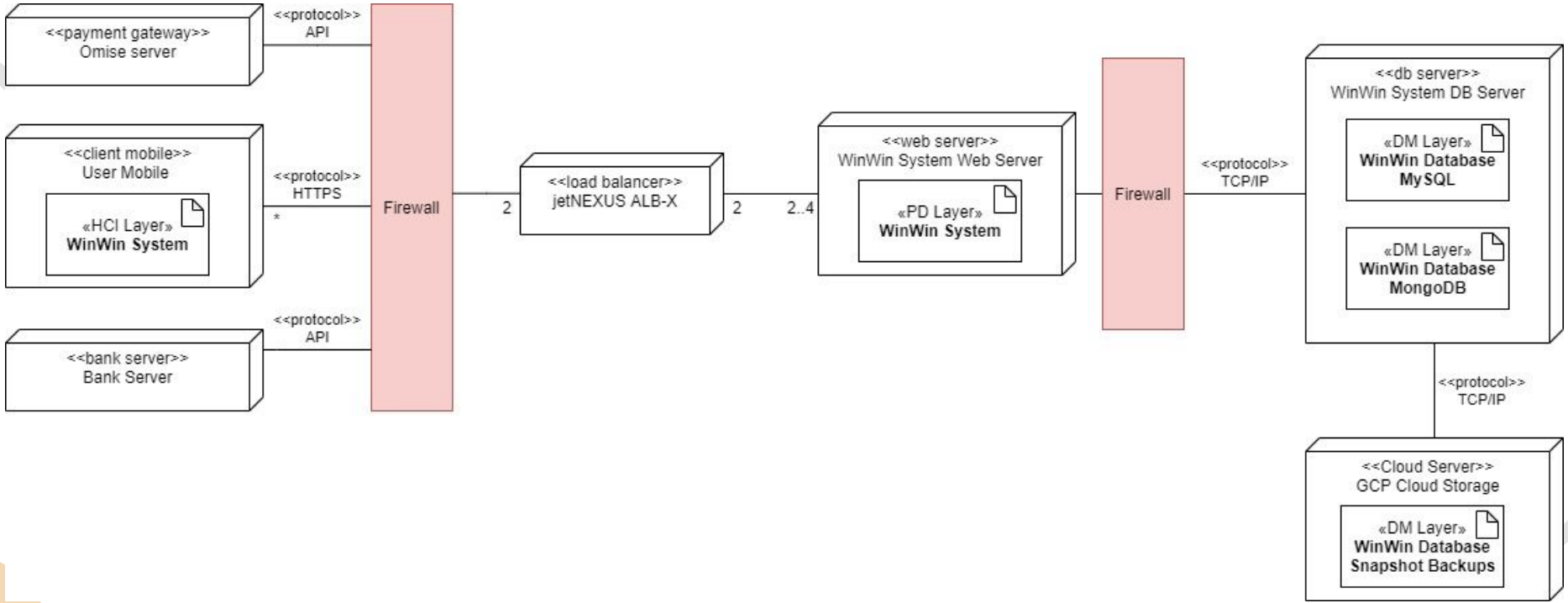
Non-functional requirements and their effect on physical architecture design

Type of Nonfunctional Requirement	Requirement	Effect for designing architectural model
Operational requirements	The system shall operate in a **mobile device** environment.	Client server has to **support mobile devices**.
	The system shall automatically **back up its database daily**.	Have at least **two databases and snapshot** to backup database daily
Performance requirements	The system shall respond in **less than 5 seconds** for every interaction between system and user.	Use thin client-server architecture with **multiple redundant servers** that can scalable easily
Security requirements	The system shall **authenticate** users and riders using userID-password.	Use thin client-server architecture that internet-based **authentication tool is more advanced**
	The system shall be able to keep users' **transactions confidential**.	**Use an external banking** system that is secure and reliable.
Usability requirements	The system should be easy to use for both new and experienced users.	-

Network Diagram



Deployment Diagram



Hardware and software specification

Specification	Standard Client	Standard Web Server	Standard Database Server
Operating System	- Android 5.0 - iOS 11.0	- Linux	- Linux
Special Software	- Winwin Application	- Apache	- MySQL - MongoDB
Hardware	- 4GB Memory - 128 GB disk drive	- 16 GB Memory - 1 TB disk drives - Dual-core 3.0 GHz processor	- 32 GB Memory - 1-4 TB disk drives - Dual-core 3.0 GHz processor
Network	- Ethernet/WiFi/ 3G	- Ethernet/WiFi	- Ethernet

The impact of your analysis and design solution

global context

- If customers use WinWin's application, the spread may be reduced because they no longer need to wait at the station.
- The system supports the use of smartphones and technology in order to drive future civilization.

societal context

- Using the service from local motorcycle taxis supports increasing work employment and cash flow in the community.
- Reduce problems about price rate that is uncertain.
- People can transport easily.

economic context

- Customers would have more options, and motorcycle taxis would have more jobs, resulting in a shift in the economy.

environmental context

- The system supports the cashless society, so it will reduce the usage of paper.

System Installation and Operation

- Migration plan
- Post-implementation activities

Migration plan

Conversion plan

Hardware installation

Servers and databases based on cloud service so no hardware is needed.

Software installation

It is necessary to install software such as the IDE as well as the environment and framework that are involved.

Data conversion

As our system is a whole new developed system to replace the old one. So we don't need to have the data conversion part.

Conversion dimension

Conversion style : Parallel

Conversion location : Phased

Conversion modules : Whole system

Migration plan

Change Management plan

Revising Management Policies

Our working culture should be flexible which could be done by using agile development.

Assessing Costs & Benefits

As our company is new to the market so our budget is low. We need to be careful about the costs and try to use the existing budget in accordance with the plan to achieve our system development goals.

Motivate adoption

We need to clarify in detail how our system can support local motorcycle taxis.

Conduct training

We will have the training session and for users we will creating an infographic on how the system works should be sufficient.

Post-implementation activities

System Support

- Online support

  user guides and FAQ

- Call center

  If the problem of the user can't be solved by reading FAQ. Users can **contact the call center** and staff will answer

System Maintenance

We have system maintenance when we get **change requests** such as problem reports. Support teams will create meetings to create plans and process to maintenance system.

Project Assessment

- Project team review

  After the system is installed, Members will **list what worked and mistakes** from this project. Project manager will summarize all lists to be **lessons learned documents**.

- System review

  System review will be **conducted every month** to compare estimated system and to-be system and analyze what we **should do in future**.

G14 Waterfall - Members

6230123921 Thitaree Setwipattanachai

6230252121 Tarm Kalavantavanich

6231301421 Kanokpich Chaiyawan

6231304321 Kittipong Deevee

6231307221 Jirawat Kusalangkurwat

6231333521 Nopdanai Sayamnet

6231353021 Raviporn Akekunanon

6231372021 Atiwat Deepo