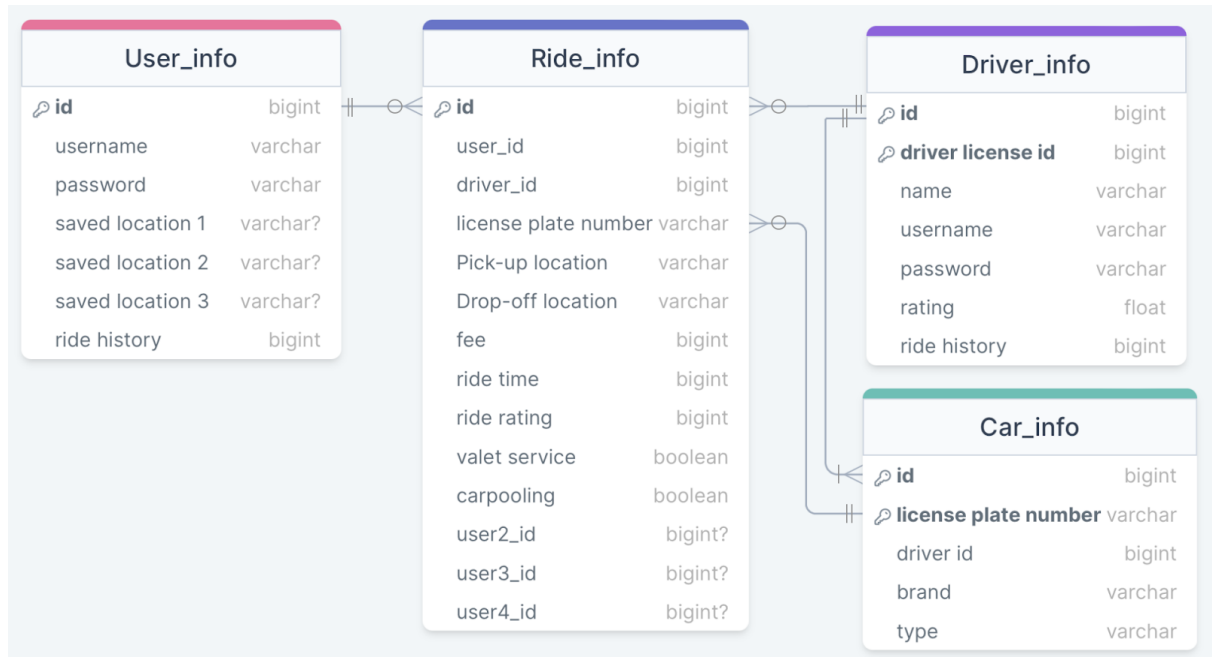


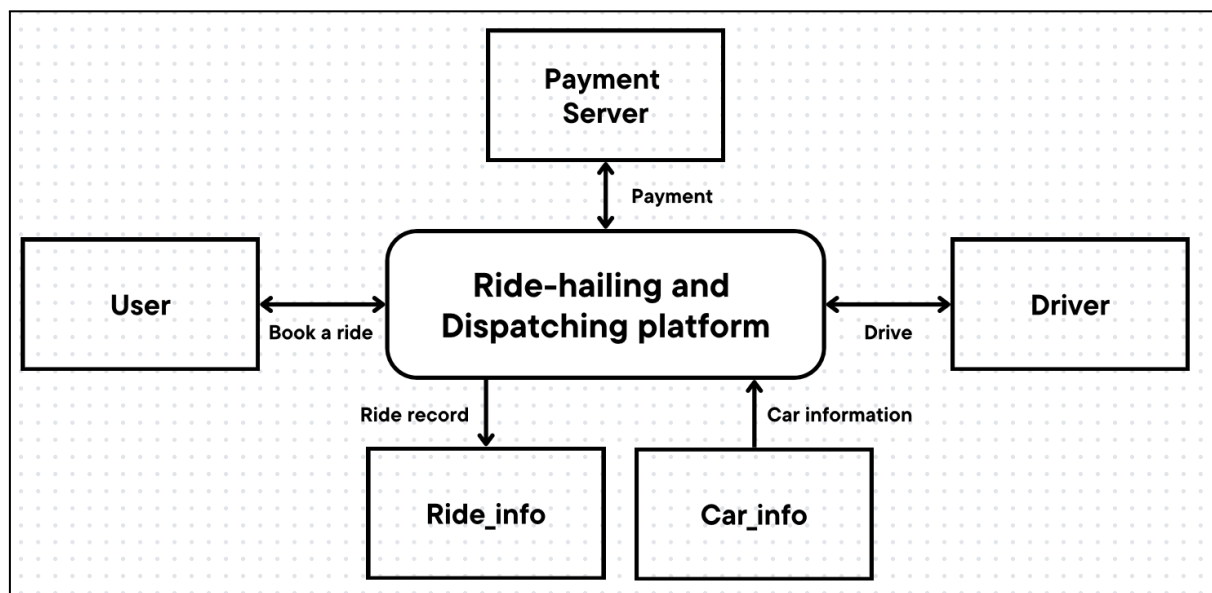
## Part II

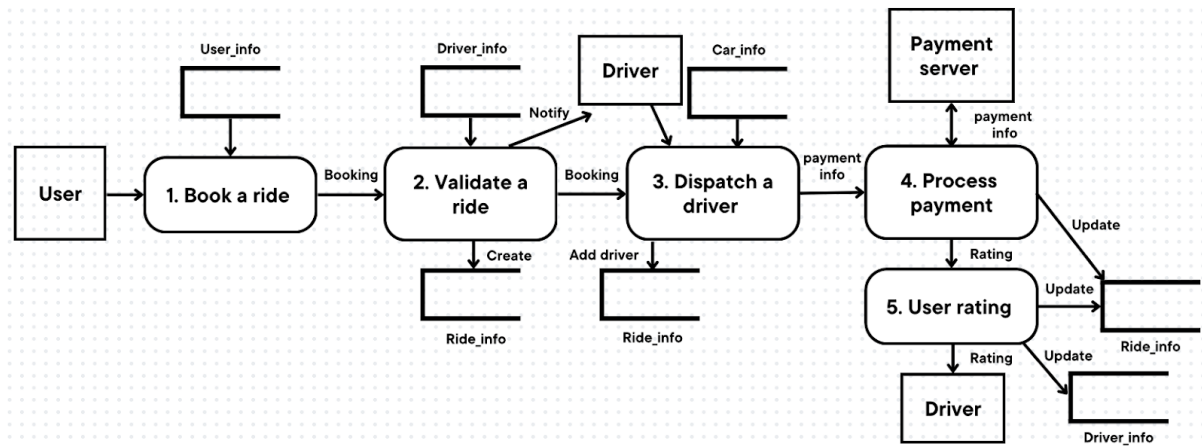
### A. ERD



### B. DFD

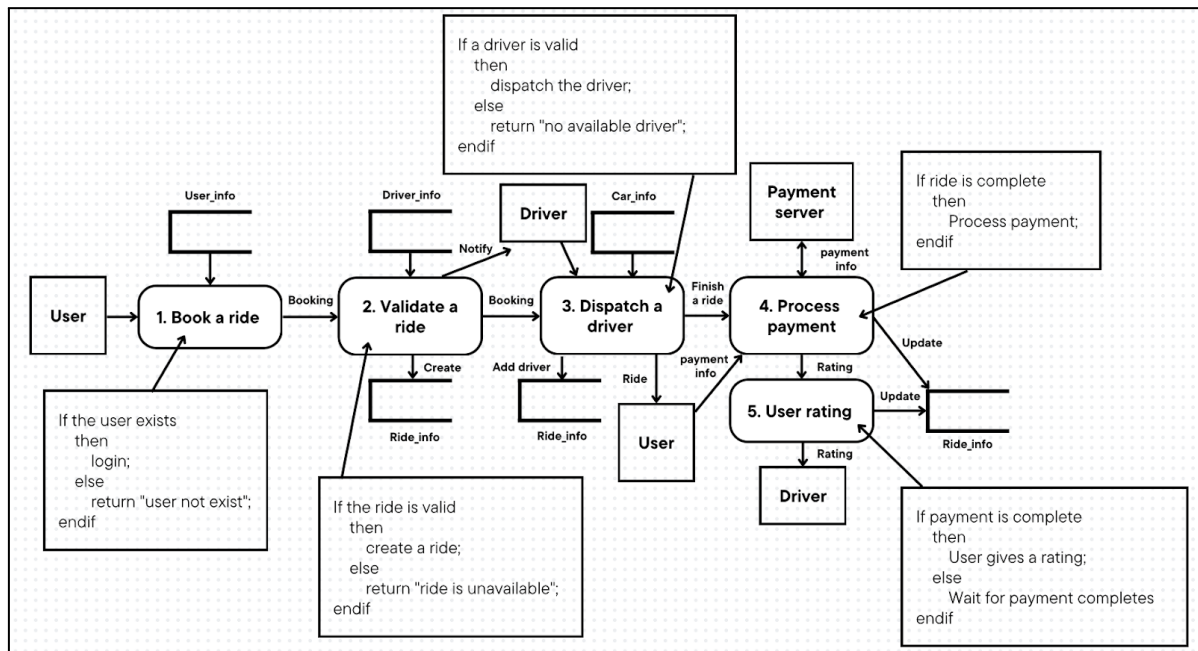
(i)





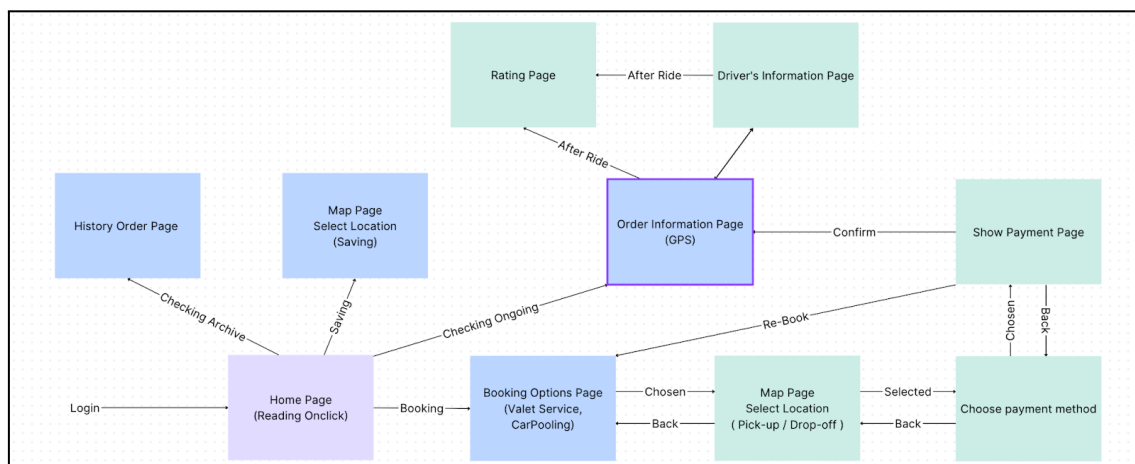
(ii)

(iii)



## C. CSPEC

(i) STD



## (ii) Decision Table

Conditions	Valet Service	Y	Y	Y	Y	Y	N	N	N	N	N
	accidental car problem	Y	Y	N	N	N	Y	Y	N	N	N
	Manmade(Y)/Natural(N) car problem			Y	Y	N			Y	Y	N
	Driver-caused car problem	Y	N	Y	N		Y	N	Y	N	
Actions	Call 110/119	✓	✓				✓	✓			
	Search New Car and Driver			✓	✓	✓			✓	✓	✓
	Apply compensation (Small amount)		✓						✓		✓
	Apply compensation (Medium amount)			✓			✓				
	Apply compensation (Large amount)	✓									

## D. Data Dictionary

### 1. Ride\_info

Name:	Ride_info
Aliases:	Ride information
Where/How used:	Validate a ride (input) Dispatch a driver (input) Process payment (input) User rating (input) Ride history (output)
Description:	Ride_info = id + user_id + driver_id + license plate number + Pick-up location + Drop-off location + fee + ride time + ride rating + (valet service) + (carpooling) + (user2_id) + (user3_id) + (user4_id)
Format:	id = *unique key generated by the server* user_id = *unique key of a user's User_info* driver_id = *unique key of a user's Driver_info* license plate number = *unique key of a user's Car_info* Pick-up location = *address of pick-up location* Drop-off location = *address of drop-off location* fee = *ride fee* ride time = *total ride time from pick-up to drop-off* ride rating = *user gives a rating to the driver*

	valet service = *whether it's a valet service or not, default is false* carpooling = *whether it's a carpooling ride or not, default is false* user2_id = *second user in the ride, non-null when carpooling is true* user3_id = *third user in the ride, non-null when carpooling is true* user4_id = *fourth user in the ride, non-null when carpooling is true*
--	--

## 2. User\_info

Name:	User_info
Aliases:	Users' personal information
Where/How used:	Login (input) Save location (input) Check saved locations (output) Ride history (input) Book a ride (output)
Description:	User_info = id + username + password + (saved locations) + (ride history)
Format:	id = *unique key generated by the server* username = *user's user name* password = *login password* saved locations = *select on map or text address to save* ride history = *show every Ride_info that user_id in it is equal to id*

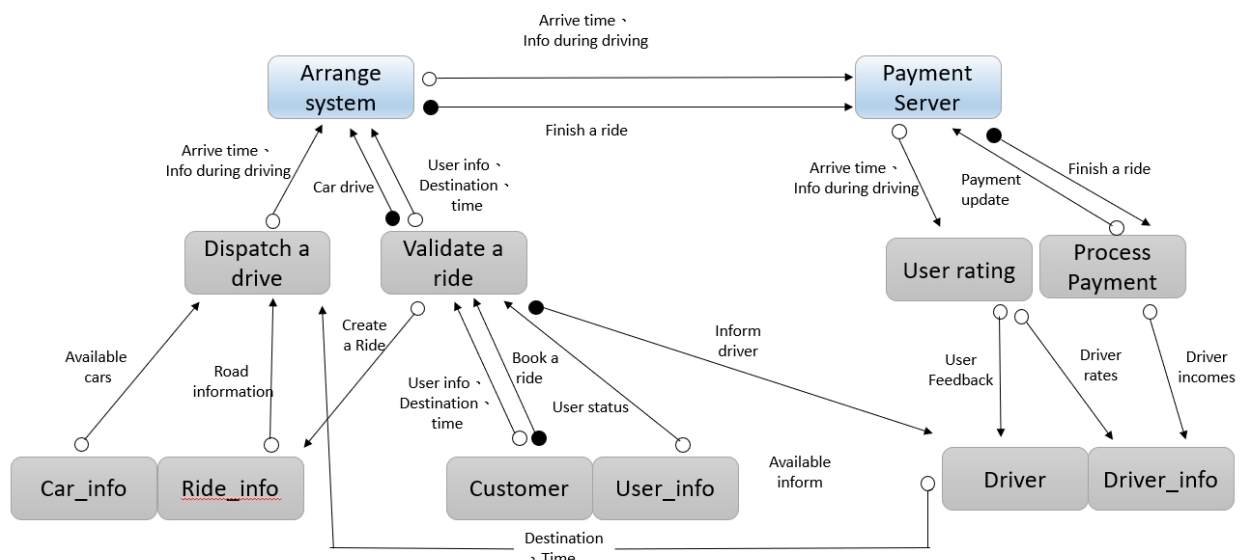
## 3. Driver\_info

Name:	Driver_info
Aliases:	Drivers' personal information
Where/How used:	Login (input) Add/Remove a car (output) Ride history (input) Validate a ride (output)
Description:	Driver_info = id + driver license id + name + username + password + rating + ride history
Format:	id = *unique key generated by the server* driver license id = *unique key generated by the government* name = *driver's real name* username = *driver's user name* password = *login password* rating = *average value of ride rating in ride history* ride history = *show every Ride_info that driver_id in it is equal to id*

#### 4. Car\_info

Name:	Car_info
Aliases:	Car information
Where/How used:	Add/Remove a car (input) Dispatch a driver (output)
Description:	Driver_info = id + license plate number + driver id + brand + type id = *unique key generated by the server* license plate number = *unique string generated by the government* driver id = *unique key of a user's Driver_info* brand = *car brand* type = *car type*
Format:	

#### E. Structure Chart



#### F. Coupling & Cohesion

##### 1. Coupling

Arrange System, Payment Server, User Rating之間有Data Coupling

Driver, Driver\_info之間有Data Structure Coupling

##### 2. Cohesion

Dispatch a Drive屬於Functional Cohesion

Car\_info, Ride\_info, Customer, User\_info屬於Communicational Cohesion

Validate a Ride, Process Payment屬於Sequential Cohesion

#### G. ChatGPT

ChatGPT 擁有下述特性可以在軟體系統分析與設計階段中輔助軟體架構分析/設計、模組設計和介面設計等工作。

特性之一為知識基礎，ChatGPT 可以提供豐富的軟體開發知識和相關概念，這對軟體架構分析/設計非常有價值。開發人員可以向 ChatGPT 提問有關軟體架構設計的問題，如適用的設計模式、可擴展性考量、模組間的交互作用等。ChatGPT 可以提供指引並分享最佳實踐，以幫助開發人員做出明智的設計決策。

另一項特性為設計建議，開發人員可以向 ChatGPT 提供特定的需求和限制，並獲取有關軟體架構、模組設計和介面設計的建議。例如，開發人員可以描述系統的特定需求，然後詢問 ChatGPT 如何設計適應這些需求的模組化結構。ChatGPT 可以提供相關的建議，包括模組之間的邏輯結構、資料流程和互動方式。

最後一項特性為原型驗證，開發人員可以使用 ChatGPT 生成的程式碼片段來驗證軟體架構、模組設計和介面設計的構想。例如，開發人員可以描述一個軟體系統的組件，並要求 ChatGPT 提供相關的程式碼示例。這可以幫助開發人員快速評估設計選擇的可行性，並確保系統的基本結構符合預期。

然而，開發人員在使用 ChatGPT 生成的程式碼片段時需要謹慎。ChatGPT 生成的程式碼僅為參考，可能不具有最佳效能、可靠性或安全性。開發人員應該仔細審查並進行適當的測試，以確保生成的程式碼符合項目需求，並遵循軟體工程的最佳實踐。此外，開發人員應該具備足夠的專業知識，以理解並運用 ChatGPT 提供的資訊。