Documentation Technical Test Software Enginner - Dealls (YC W22)

A. Github URL
https://github.com/triadmoko/Dealls

B. Requirements
The requirements when using the application so that it can run well are as follows :
a. Golang version 1.21 or higher
b. PostgreSQL
c. Docker (Optional)
d. Docker Compose (Optional)
e. Makefile
f. Golang Migrate
g. Golang Mockery
h. Buf Protoc

C. Tech Stack

a. Connect RPC

Connect RPC is a library for building RPC servers and clients in Go. It's built on the gRPC framework, and it's designed to be easy to use and highly performant.

Connect RPC support multiple protocols, such as HTTP/1.1, HTTP/2, and gRPC.

b. Gin Gonic

Gin is a web framework written in Go, Gin support RESTful API and http2 it's easy to use.

c. Gorm

Gorm is a library for building ORM in Go, it's easy to use and support multiple databases. Using gorm we can build a database model easily and fast, then focus for the business logic.

d. Go Cron

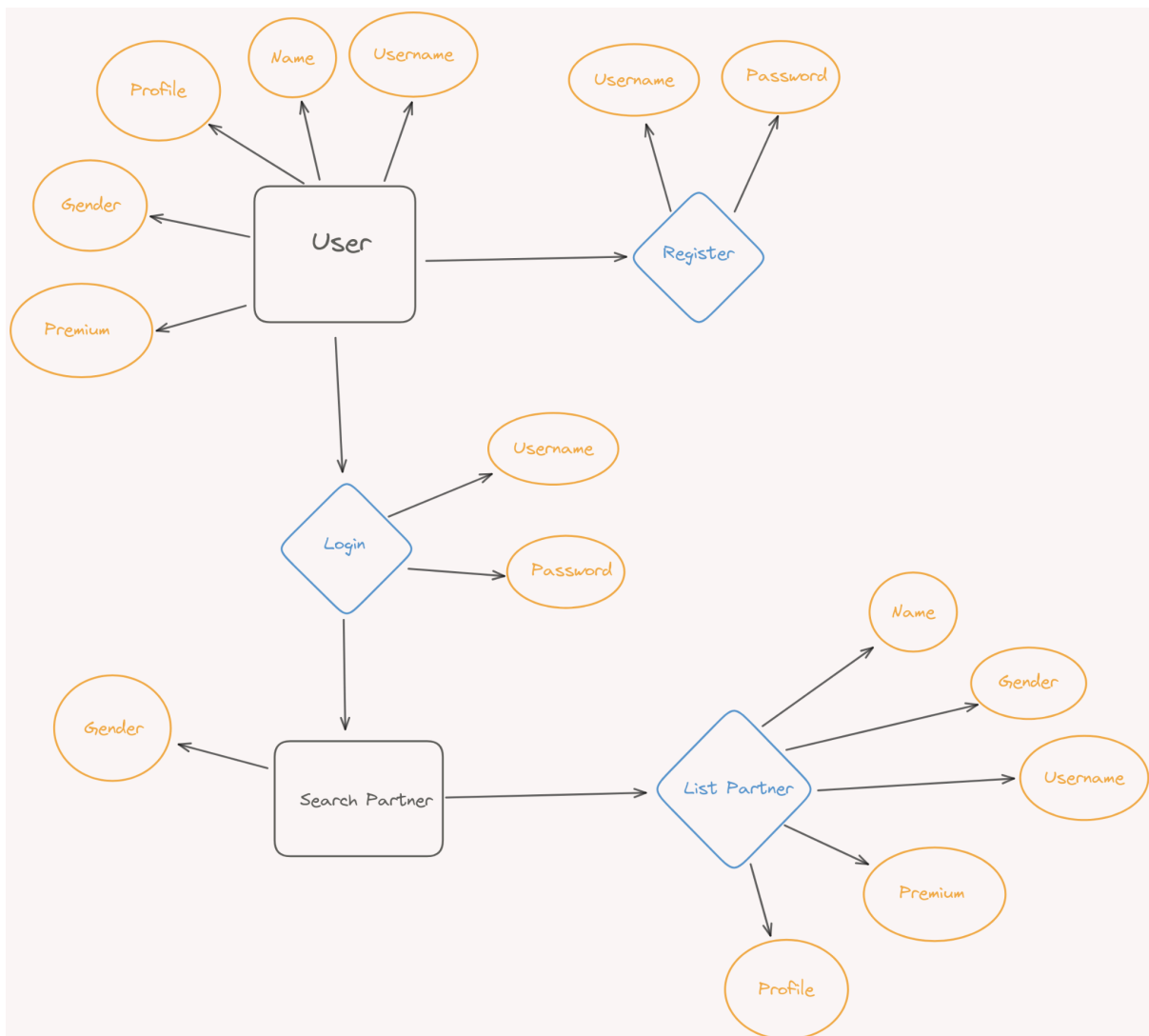Go Cron is a library for building cron jobs in Go, it's easy to use and support multiple time format.

I use this library to build a cron job for deleting the table interest that has been show for yesterday.
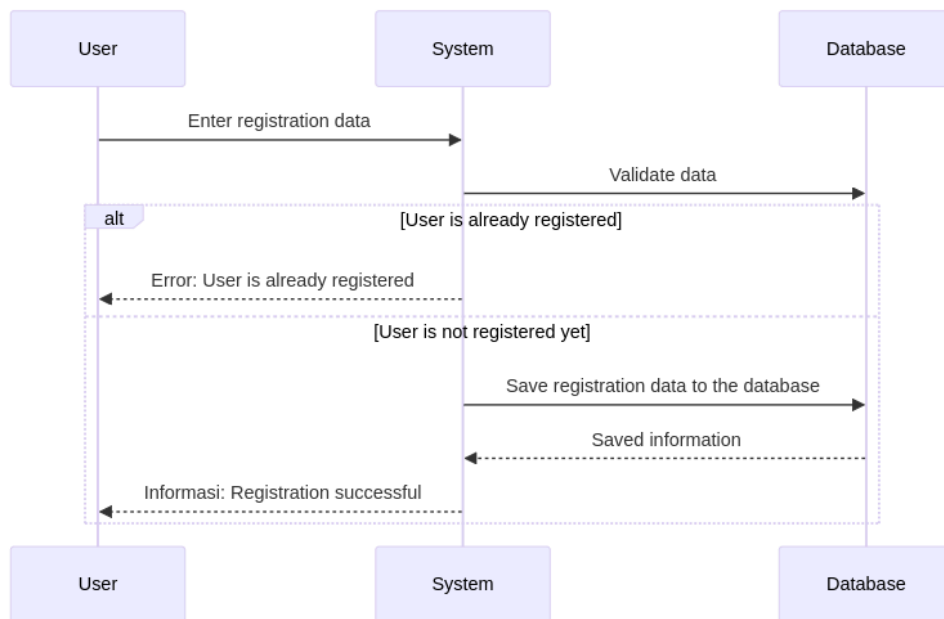
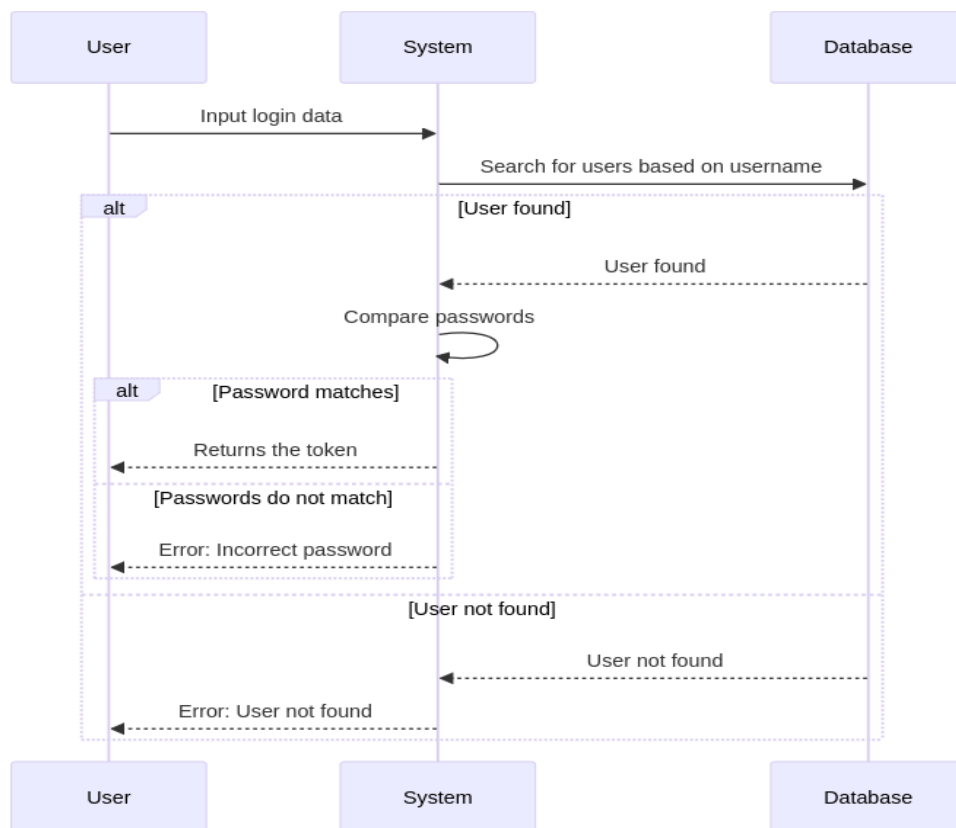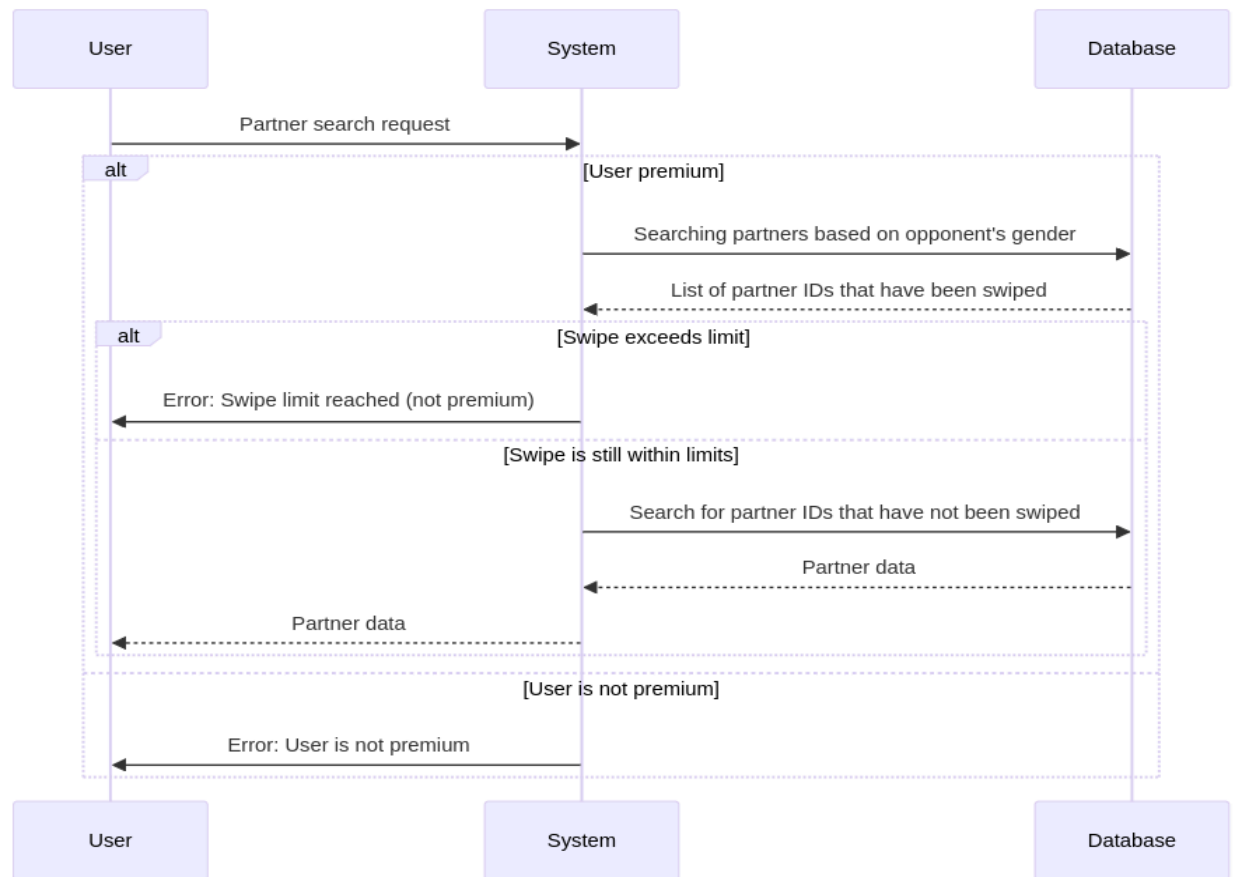c. Github workflows to run unitest
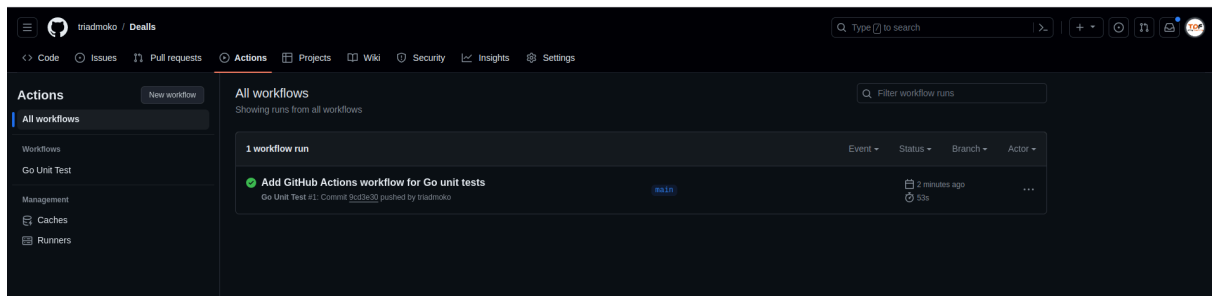
D. Design System
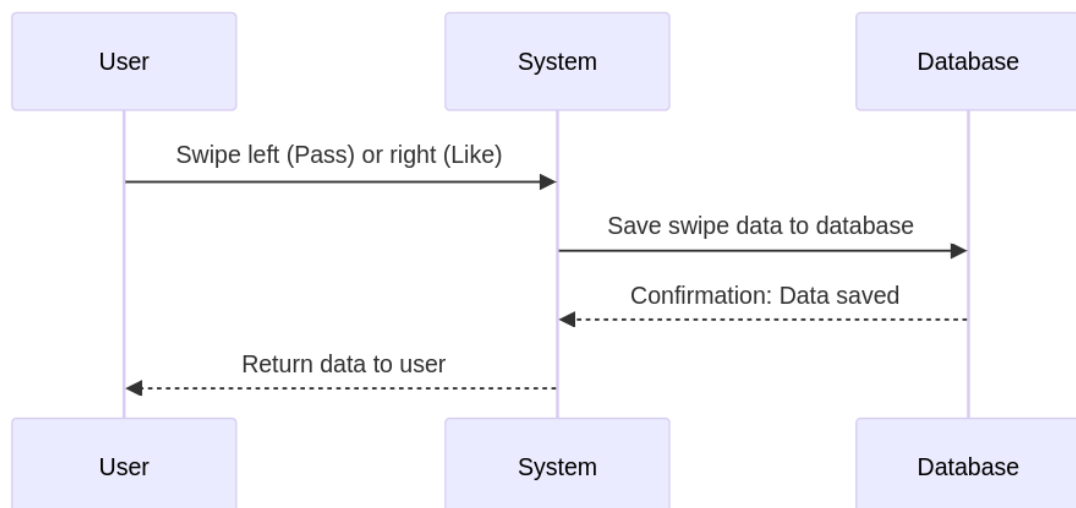
a. ERD

b. Sequence diagram
   1. register

2. Login

3. Searching Partner

4. Swipe Partner

5. Payment
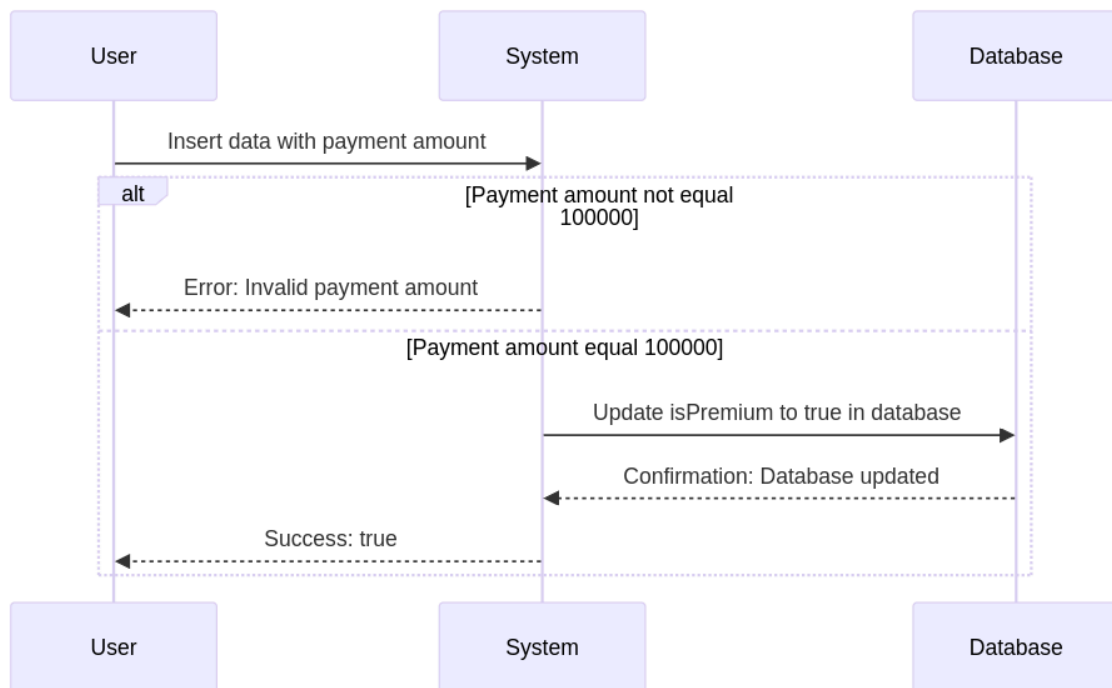
E. Test Case

#Test Case	Description Test Case	Test Data	Expected results	Actual Results	Pass/Fail

Register	Success Register	{
		    "username":"testuser",
		
		"password":"testpassword",
		"gender":"1",
		"name:"testname",
		"profile":"testprofile",
		"premium":false
		}
			No Error	No Error	Pass

	User Already Exist	{
		 "username":"testuser",
		
		"password":"testpassword"
		
		}
			username already exist	username already exist	Pass

	Test Case	Input	Expected	Actual	Result
	Internal Server Error, mock data for internal server error, example bug query to database	{ "username":"testuser", "password":"testpassword" }	Internal Server Error	Internal Server Error	Pass
Login	Success Login	{ "username":"testuser", "password":"1" }	No Error	No Error	Pass
	Wrong Password	{ "username":"testuser", "password":"2" }	password not match	password not match	Pass
	User Not Found	{ "username":"testuser1", "password":"1" }	user not found	user not found	Pass
	Internal Server Error, mock data for internal server error, example bug query to database	{ "username":"testuser1", "password":"1" }	Internal Server Error	Internal Server Error	Pass
Search Partner	Success SearchPartner	{ "per_page":10, "page":1 }	No Error	No Error	Pass
	User Limit Not Premium	{ "per_page":10, "page":1 }	user not premium	user not premium	Pass
	User Premium Not Limit	{ "per_page":10, "page":1 }	No Error	No Error	Pass
Swipe Partner	Success SwipePartner	{ "partner_id":"partner_id", "is_interest":true }	No Error	No Error	Pass

			No Error	No Error	Pass

Payment Premium	Success Purchase Premium Package	{
		   "payment_amount":100000
		}
			No Error	No Error	Pass

	Invalid Payment Amount	{
		   "payment_amount":10000
		}
			No Error	No Error	Pass