



澳門大學
UNIVERSIDADE DE MACAU
UNIVERSITY OF MACAU

Barber & Salon booking app

Course : CISC3026 Software Engineering Principles

Group number : 7

Group members :

DC026718 UN WENG HOU

DC026157 WONG KAI YUAN

DB926105 LAO CHI IAN

DC028721 IEONG UN SAN

Date : 2022/11/21

Table Of Contents

- Introduction

- Requirements model :
 - List of users
 - List of functional requirements for each user
 - List of non-functional requirements
 - Use case model

- Design model :
 - Static system model: class diagrams
 - Dynamic system model: sequence diagrams, activity diagrams, state diagrams

- Test specification :
 - Test case specifications for each functional requirement
 - Test case specifications for each non-functional requirement

- Conclusion

Introudction

In the situation of Macau, making appointments for haircuts or salons are usually made by phone calls or some messaging app such as wechat. Which is inconvenient, inefficient and easy to cause confusion for both customers and merchants. In our projects, we decided to design an online booking system, which allows users to book a haircut or beauty salon services on their own. The system will be designed as a mobile phone software. We expect this software to include all barber shops and salons in Macau. Customers could search for barber shops or salons according to their location, popularity or stylist. The software should show the schedule and available time for each stylist including their introduction and some of their works. The system also involves other services such as beauty product stores, advertisements, grading and comments, online payment etc. In order to save time, increase convenience for users, gain sales figures and reduce labor cost for merchants, easier and more efficient management. After making the appointment, customers could briefly describe their needs. Stylists can prepare early , which also saves a lot of time. In addition, there will be a login system, every customer will have their own account. All consumption records will be stored in the system. Customers can view at any time. Merchants can also clearly understand their revenue.

Requirements model

List of users

The target users will be all citizens in Macau. If every citizen gets a haircut or beauty service every 2 to 3 weeks. There will be about 800,000 reservations per month. But also consider hair salons that don't require appointments. For example, quick cut hair. There will be approximately 400,000 users.

Functional requirement

Functional requirements define the functions of a software, and are also the functions and services that a system needs to provide. The functional requirements of the software are below.

- Associate every online booking with an account
 - Users have to login before making appointment, account could provide user basic information
- One user can only have one account
 - phone number verification
- Enable users to search for the shop according to their selection
 - select by location, price, popularity, grade, fastest available for appointment
- Enable users to select different services provided by the shop
 - haircut, color, facial, hair treatment, permanent.

- Accept users to check available date and time of the stylist
 - there will be a table showing the available date and time of the stylist
- Booking confirmation should be sent to the specified contact details
 - email, messenger
- Calculate and display total amount that users need to pay
 - After making the appointment, the price that the user has to pay should be shown.
- Cancel bookings
 - Allow users to cancel booking by their own
- Advertisement
 - There should be an area displaying advertisement
- Product Attributes
 - showing product characteristics, price, size, quantity, source
- Online Payment
 - Allow user to pay for the product or services via various payment methods
- Comment and grading
 - Allow user to comment and give grade to the services

Non-functional requirement

- Non-functional requirements refer to those requirements that are not directly related to the specific functions of the system, but they are related to the overall characteristics of the system, such as reliability, response time, storage space, etc. The non-functional requirements of the software are below.
- Use captcha and encryption to avoid bots from booking
 - prevent cyber attack and secure information of users
- Performance, Usability
 - System should perform reliably and smoothly. Easy to use, efficient, and accessible
- Maintainability
 - System should be easy-to-maintain right from its launch. Reduce system complexities as possible therefore easy for engineers to make changes to the system in the future.

Use case model

A use case diagram is the simplest representation of the interaction between the user and the system, showing the relationship between the user and his related use cases. Through the use case diagram, people can understand the different types of users and use cases of the system. Use case diagrams are also often used in conjunction with other diagrams. Through the use case diagram we can know who is the system user, what can users do through this system and how does the system solve this user need?

Use case diagram of the system

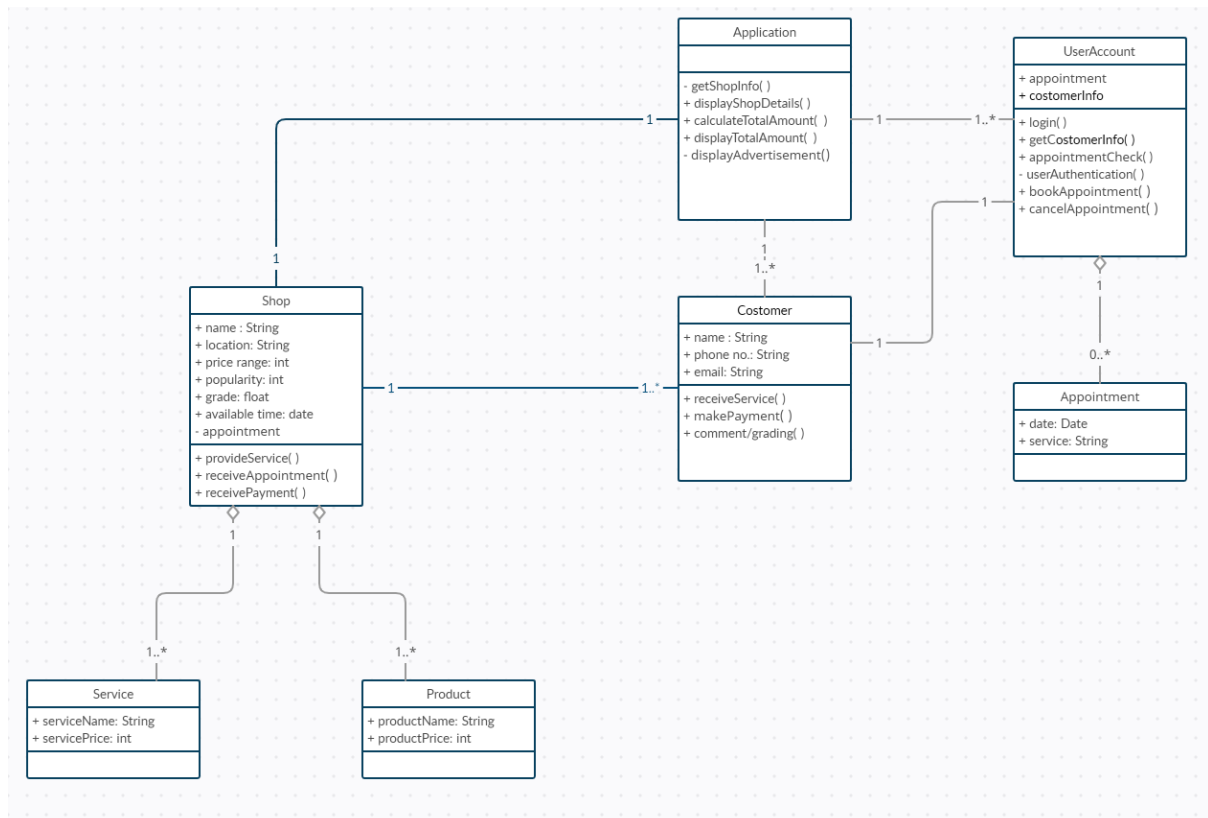


Design Model

Static System Model :

1) Class Diagram

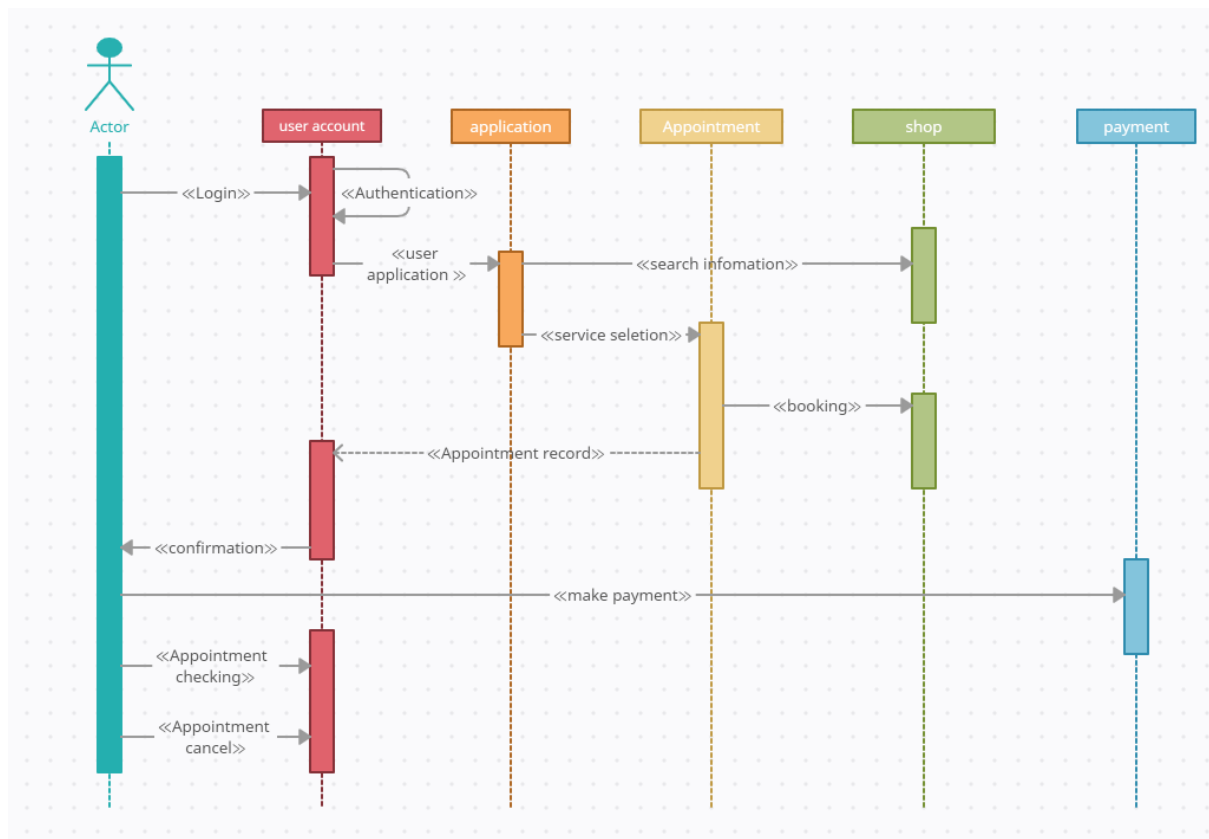
A class diagram in the Unified Modeling Language (UML) is a form of static structural diagram that depicts a system's classes, their properties, operations, and relationships between objects to explain the structure of the system.



Dynamic System Model :

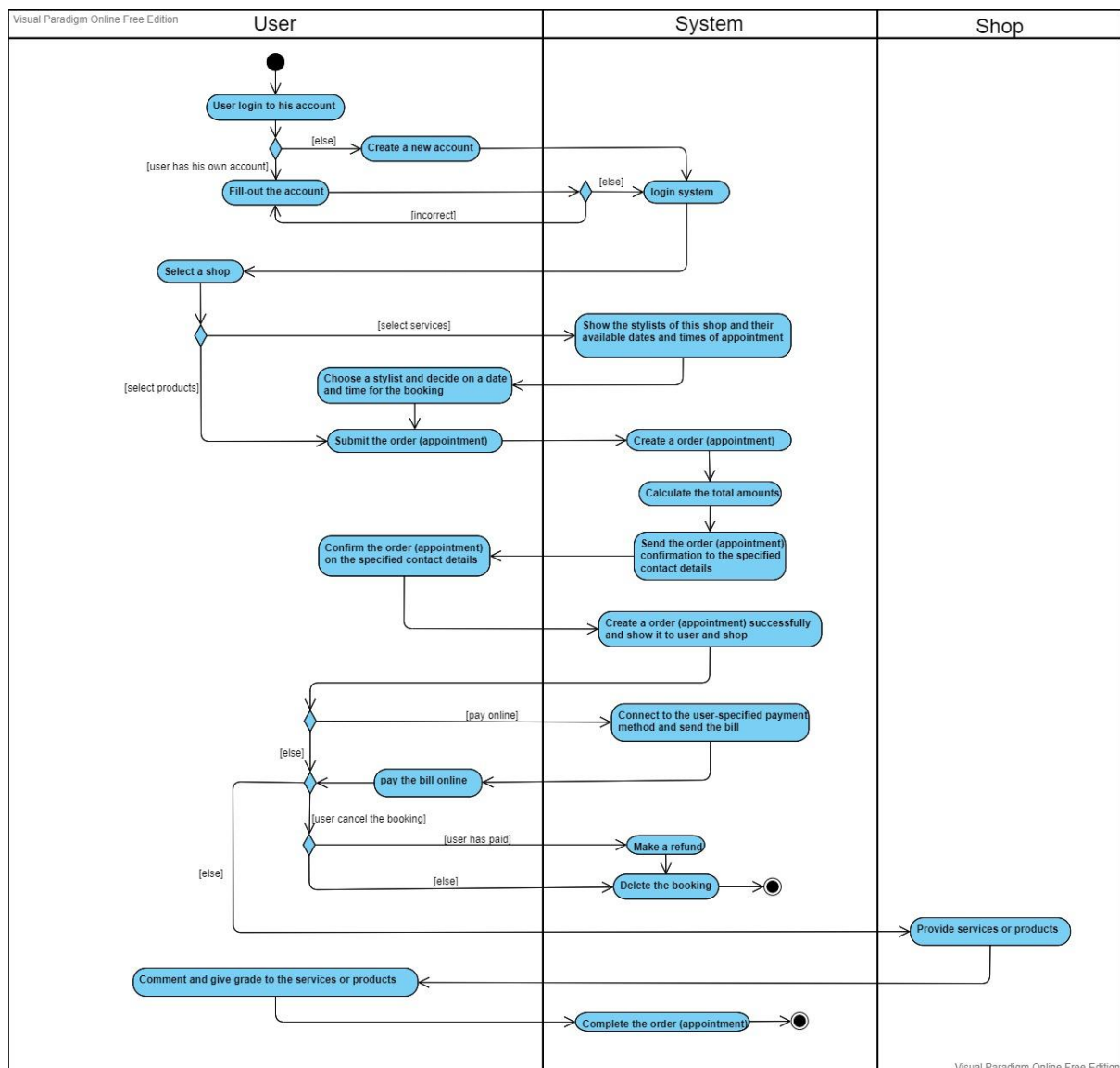
2) Sequence Diagram

UML Sequence Diagrams are interaction diagrams that describe the steps used to complete an operation. They depict how items interact within the framework of a cooperation.



3) Activity Diagrams

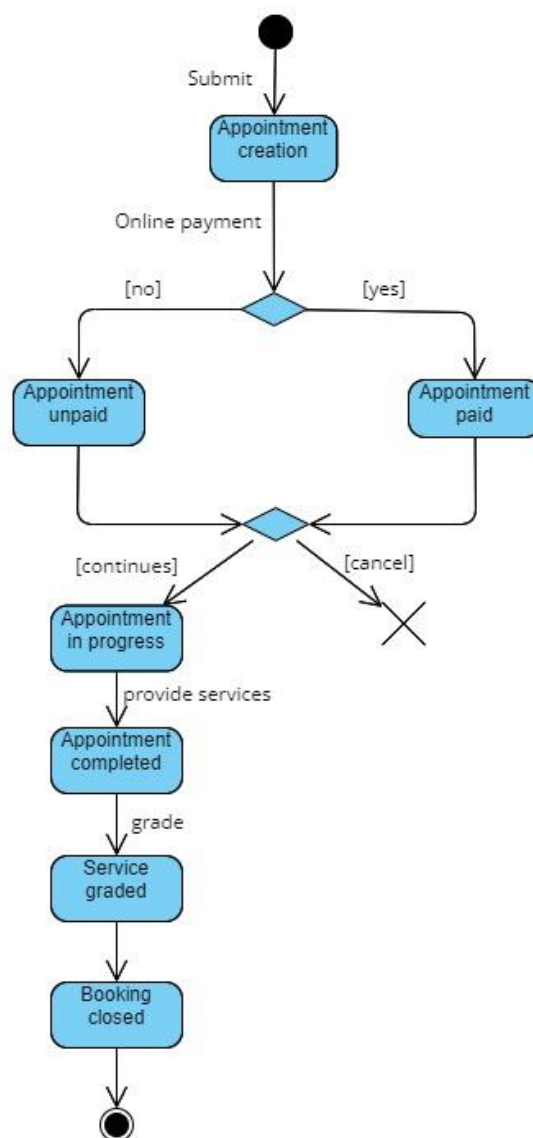
An essential behavioral diagram in a UML diagram for describing dynamic characteristics of the system is an activity diagram. An activity diagram is essentially a more complex flow chart that models the transition between activities.



4) State Diagram

State machine diagrams created using UML display an entity's various states. State machine diagrams can also demonstrate how an entity reacts to different events by switching between different states. A UML diagram called a state machine diagram is used to represent a system's dynamic nature.

State Diagram of the Appointment



Test Specification

A test case in software engineering is a description of the inputs, execution conditions, testing process, and anticipated results that define a single test to be executed to achieve a specific software testing objective, such as to exercise a specific program path or to verify compliance with a specific requirement. Testing that is deliberate rather than chaotic is supported by test cases. To achieve the desired coverage of the software under test, a battery of test cases can be created. Regression testing is efficient and reliable when the tests are run frequently against newer versions of the program thanks to formally stated test cases.

In our project, to simplify, we will only include “Input”, “Tests” and “Output” for each functional and non- functional requirements. If some inputs or outputs cannot be defined, then it will not be included in the test specification.

Test Specification of Functional Requirement

- 1) Associate every online booking with an account (so that can provide basic information)

Input :

Online booking ID, User account details

Tests :

Test for input where user account has detailed information

Test for input where user account has no information

Test for input where there is no user account (customer visit apps as guest)

Output :

OK or error message indicating that online booking is not associated with an account that is suitable.

2) One user can only have one account

Input :

User account, Phone number that bind with account

Tests :

Test for input where 1 user account with 1 unique phone number

Test for input where 1 user account with multiple phone numbers

Test for input where multiple accounts with 1 same phone number

Output :

OK or error message indicating that one user has more than one account.

3) Enable users to search for the shop according to their selection

Input :

selections by location, price, popularity, grade, fastest available for appointment

Tests :

Test for input where all field is empty

Test for input where all field has valid data

Test for input where all field has invalid data

Test for input where the "location" field has valid data.

Test for input where the "location" field has an invalid data.

Test for input where the "price" field has valid data.

Test for input where the "price" field has an invalid data.

Test for input where the "popularity" field has valid data.

Test for input where the “popularity” field has an invalid data.

Test for input where the “grade” field has valid data.

Test for input where the “grade” field has an invalid data.

Test for input where the “fastest available for appointment” field has valid data.

Test for input where the “fastest available for appointment” field has an invalid data.

Output :

List of desired search result or error message indicating that search is not successful.

4) Enable users to select different services provided by the shop

Input :

Shop details

Tests :

Test for input where services are provided by shop

Test for input where services are not provided by shop

Test for input where all services are chosen

Test for input where multiple services are chosen

Test for input where only one service is chosen

Test for input where no service is chosen

Output :

OK or error message indicating that services provided by the shop cannot be chosen.

5) Accept users to check available date and time of the stylist

Input :

Stylist timetable

Tests :

Test for input where available date and available time is chosen

Test for input where available date but not available time is chosen

Test for input where not available date but available time is chosen

Test for input where not available date and not available time is chosen

Test for input where past date is chosen

Test for input where far future is chosen (like after 10 years)

Test for input where no date is chosen but available time is chosen

Test for input where available date is chosen but no time is chosen

Test for input where no date and time is chosen

Verify that the user can not select or is not permitted to select time that are already booked or not allowed for booking.

Output :

OK or error message indicating that date and time is not applicable

6) Booking confirmation should be sent to the specified contact details

Input :

Booking ID, contact details

Tests :

Verify that booking confirmation is successfully generated.

Test to send booking confirmation where no contact details within booking ID

Test to send booking confirmation where phone contact is set as contact details.

Test to send booking confirmation where email is set as contact details.

Test to send booking confirmation where wechat (or other software) is set as contact details.

Check once the booking is confirmed, the stylist timetable is filled

Output :

OK or error message indicating that booking confirmation is not sent to the user.

7) Calculate and display total amount that users need to pay

Input :

Product/Service selected

Tests :

Test for input that the service or product selected by customer is correctly calculated

Test for input that the service or product selected by customer is correctly displayed

Test for input that the service or product selected by customer is incorrectly calculated

Test for input that the service or product selected by customer is incorrectly displayed

Check if the user can Amend an existing order by adding an additional feature or comment to seller / service provider

Output :

Price to be paid or error message indicating that calculation of total amount is not successful

8) Cancel bookings

Input :

Booking ID, User details

Tests :

Check on canceling the bookings the system does not refund the money. (If there are any such conditions applied)

Check on canceling the booking the system refunds money to the same way from which the account amount is paid.

Verify that after cancellation the seat's status is updated to available.

Test for cancel a booking that doesn't exist

Test for cancel a booking that was canceled before

Test for cancel a booking that was already past

Output :

OK or error message indicating that canceling attempt cannot be made

9) Advertisement

Tests :

Verify the maximum limit of advertisement that a page can have, having more advertisements than permitted results in error messages.

Test when there is no advertisement

Test when there is advertisement

10)Product Attributes

Input :

Product ID with attributes (characteristics, price, size, quantity, source)

Tests :

Test for product attributes that are all filled

Test for product attributes that are all empty

Test for product attributes that are negative in quantity.

Verify that the pricing of different types of attributes is displayed to the users.

Verify that the user should see real-time stock number of product.

Verify that users can successfully select single or more than one attribute.

Output :

OK or error message indicating that product attributes need to be further revised.

11) Online Payment

Input :

Payment amount, product list, Payment method

Tests :

Test for input where payment that are failed

Test for input where payment that are successful

Verify that all the different types of payment methods work fine.

Check for previous billing amount is paid

Check if the user able to check the receipt and download it

Verify that after successful payment the user selected time is updated to booked.

Output :

Receipt of order or error message indicating that payment is not successful

12)Comment and grading

Input :

Customer details, Comment, Grading

Tests :

Test for input where no grade is being input

Test for input where maximum grade is being input

Test for input where grade between 0 – max is being input

Test for input where no comments is being input

Test for input where lengthy comments is being input

Verify that only customer that have used the service is allowed to give comment and grade

Output :

OK or error message indicating that comment and grading is not valid.

Test Specification of Non- Functional Requirement

1) Use captcha and encryption to avoid bots from booking

Tests :

Test for correctly input for captcha

Test for incorrectly input for captcha

Test for many attempts on input for captcha

Output :

OK or error message indicating that unable to avoid bots

2) Performance, Usability

Tests :

Application load time should not be more than 5 secs up to 1000 users accessing it simultaneously

Software should be installable on all versions of IOS and Android

User should be able to use app after short introduction / tutorial when first got into apps

Response time of each page should be less than 1 second

Output :

OK or error message indicating that performance does not achieve pre defined metrics.

3) Maintainability

Tests :

Time to make changes on system should be less than time needed to restart the whole project

Cost to make changes on system should be less than time needed to restart the whole project

Test on different architecture to check which has the most efficient and least complexity.

Conclusion :

There is no denying that Macao is a small town, and its market is limited. On the other hand, it could be one of Macao's advantages. Small market can be combined with high efficiency and low cost. This software can make haircutting more convenient for Macau people. It is no longer necessary to call the hair salon or queue up in the shop to get a new hairstyle. People only need to make an appointment on the software, to save time they spend on haircutting. Software also provides an exposure platform for salons to help them expand the customer base, no more limit to the previous publicity model that only relied on customers. We believe it can change the hairdressing service market in Macau.