Immagine che contiene logo

Descrizione generata automaticamente

Iqueue Project

UML Architectural

Software Engineering for Automation (2022-2023)

Immagine che contiene testo

Descrizione generata automaticamente

*UML Architectural* helps to understand the architectural class diagram of our Iqueue project. The architectural UML has the goal of describing the relationships among the different actors in terms of the internal software components. This means that the Iqueue app will be divided into modules, each one with its own functionalities and which communicate with other components through interfaces. For this reason, the document is divided into three main sections: first a description of the principal components of the Iqueue system is given, then the main interfaces among modules are explained and finally practical examples are shown with Sequence diagrams.

1. Components

In this section the main components of the system are analysed.

* User GUI

The User GUI represents the graphic interface for the User. It is differentiated from the Customer and Shop Owner GUI since in our app the user can be of these two types and thus he will have a different GUI according to the typology of user (Customer or Shop Owner) he chooses.

* Customer GUI

The Customer GUI represents the graphic interface for the user who has selected to be a customer. It is characterized by the id of the customer, the position and the rewards he has received. The User GUI is characterized by the methods User\_login, User\_logout and User\_registration which allow the user to login and register in the Iqueue app respectively. Focusing on the User\_registration(in Name:String, in Surname:String, in Birthday:Date, in Email:String, in Password:String), means that the user should give as input its name, surname, birthday, email address and password to register.

* Shop Owner GUI

The Shop Owner GUI represents the graphic interface for the user who has selected to be a shop owner. It has as attribute the shop owner id.

Immagine che contiene testo, linea, ricevuta

Descrizione generata automaticamente

Figure 1: Relationships among User, Shop\_owner and Customer GUI.

* Dispatcher\_Customer

The dispatcher\_customer component has the crucial role of receiving request from the other component interfaces and send them to other interfaces, managing in this way the different tasks. This dispatcher element is specific for the Customer requests and there is a dual component for the Shop Owner. Due to the meaning of this component, it is naturally connected to many other interfaces as shown in the following figure

Immagine che contiene testo, diagramma, linea, Diagramma

Descrizione generata automaticamente

Figure

* Booking\_manager

The booking manager has to goal to supervise the bookings of the time slots from the Customers. For this reason, it is strictly connected to the QRcode manager and the Queue manager, that are the other components used for the booking algorithm.

* QRcode\_manager

This component is focusing on the QR management, since the QR code is the fundamental tool with which the Iqueue app increment or decrement the queue and manages the bookings.

* Queue\_manager

This module is strongly connected to the previous ones and its role is to coordinate the queue counter and thus again the bookings.

Immagine che contiene diagramma, testo, schizzo, mappa

Descrizione generata automaticamente

Figure 3: Relationships among Booking\_manager, QRcode\_manager and Queue\_manager.

* Dispatcher\_shopowner

The dispatcher\_shopowner has the same role of the dispatcher\_customer, but it is focused on the Shop Owner activities.

* Shop\_manager

The Shop manager is a crucial component since it has functionalities both Shop Owner side and Customer side. In fact, on one hand it collects the shop registration information from the Shop Owner, and on the other hand it manages the requests on the shop made by the Customer.

* Product\_manager
* Subscription\_manager

1. Interfaces
2. Sequence diagrams

* Customer subscribe
* Shop owner subscribe
* Add/remove Shop
* Add/remove Shop product
* Add/remove Shop product discounts
* Add/remove Booking
* QR code generation
* Sent notification
* Add/remove customer from queue
* Show shop
* Compute waiting time
* Add shop review
* Generate rewards

The main functions that the shop owner should do are:

* Add/remove customer without APP in queue
* Count products
* Set waiting time
* Generate Advertisement

The main functions that the customer should do are:

* Select shop
* Select time slot
* See queue
* Add/remove from wishlist
* See purchased list

The main functions that the GPS system should do are:

* Guide the client to the desired shop
* Track the customer position