CENG3004: Software Engineering

The Restaurant and Menu Ordering System Management

Design Document

10/06/2021

Team Members:

Kıymet Deren TOY
Görkem SAVRAN
Emre TAŞKIN

Contents

1 Overview

Our project will be a software package that facilitates the ordering in traditional restaurants. Nowadays, orders are carried out by taking notes on the papers by waiters in many restaurants today and they are also transmitted to the kitchen side with papers. In addition to causing some confusion, this situation makes things difficult, causes too much paper to be consumed and even slows down the order action. Communication disorders between people such as waiter-customer, waiter-cook, waiter-waiter, etc. causes deterioration in the traditional ordering system. With this project, we aim to minimize these problems and offer a customer-friendly system.

Our customers are institutions that include the food and beverage industry, such as restaurants, cafes and hotels. Our project aims to increase customer satisfaction in these institutions and to regulate the order system. Thanks to this software, the customer (in the restaurant) can view the menu, place an order, view order status, call the waiter and pay the bill through the tablets placed on her/his table. In addition to these, customers can read the comments made by other customers about the products (food and beverage), leave comments if they wish, and view the most popular products of the restaurant.

Waiters can set up the table, manage the table remotely, confirm the order, and remotely control table functions to assist customers with tablets. In addition, staff working in the kitchen can view the approved orders from the kitchen. During preparation, they are able to let the waiter know the status of each order, and can send notifications when products are completed, through the touch-display.

The system contains full accountability and logging systems, and supports supervisor actions to account for exceptional circumstances, such as an order being refunded or walked out on. Customers are presented with an attractive and easy-to-use surface computer GUI.

Waiters are able to perform all actions that the table system normally handles via their tablet PCs, so in the event of a customer being unable to operate the surface computer, the waiter can handle orders traditionally while using retaining the accountability and logging functions of the system, and retaining the same channel of communication with food staff.

As can be understood from its functions, this software provides people with both convenience and time. We are doing this project because we aim to reduce the use of paper and provide people with an enjoyable meal that they can manage completely with our software on tablets.

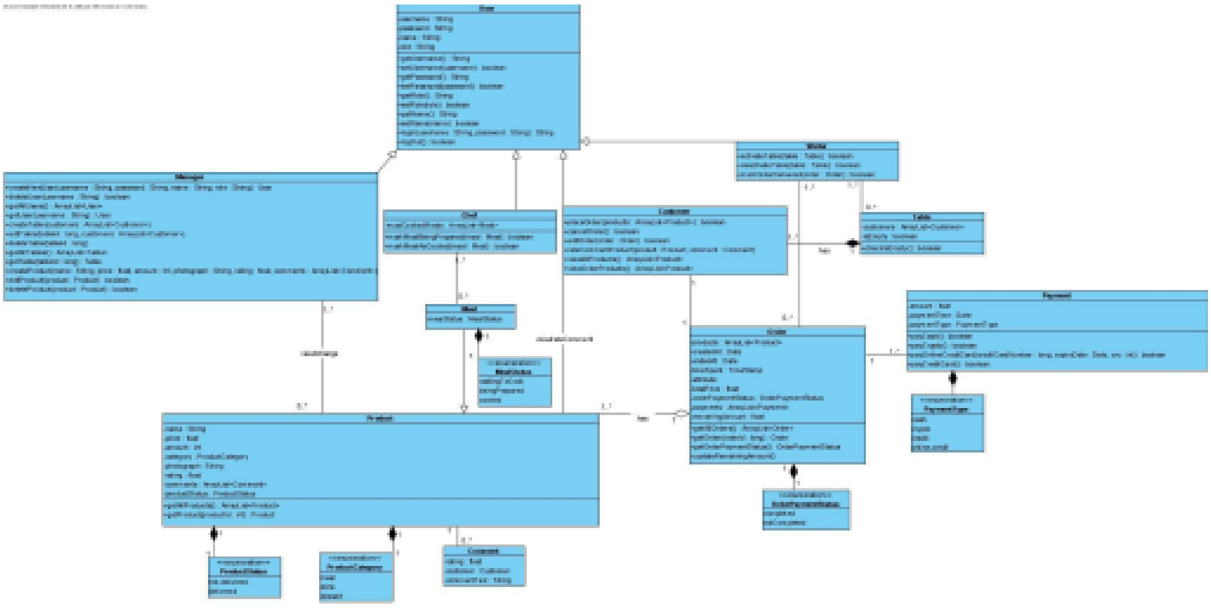
2 Design Goals

Goal's Concern	Related Requirement Identifier	Description
Reliability	NR-1	When the system shuts down, the restart time does not exceed 10 seconds, which ensures that the system is reliable.
Reliability	NR-2	During system use, the response time of the system should not exceed 10 ms seconds which ensures that the system is reliable.
Reliability	NR-3	The upper-level limit of the system to perform transactions should be at a level that can perform 1000 transactions per second which ensures that the system is reliable.
Fault tolerance	NR-4	The system shall log every state and change of state of every surface computer, tablet and display to every second to provide recovery from system failure. This ensures that the system has fault tolerance.
Fault tolerance	NR-5	In case of any failure(e.g. system crash or power loss), the system should be able to return to its previous state within 10 seconds which ensures that the system has fault tolerance.
Traceability of requirements	NR-7	The cryptocurrency payment method, one of the payment methods, can be used when the necessary government regulations are provided. This ensures the system is traceable for requirements.
Security	NR-8	The system requires a staff member to enter username and password information to log in to the system. This requirement ensures that the system is secure.
Reliability	NR-9	Each user can open an account from only one device. The system prevents user login from two different devices at the same time which ensures that the system is reliable.
Security	NR-10	The bit strength of the waiter/chef/manager password used to log in to the system must be at least 64 bits. This helps maintain the security of the system.
Security	NR-11	The passwords used for login in the system must be changed by the users after three months of use.This

		helps maintain the security of the system.
Reliability	NR-12	The system cannot interfere with any panel without the user's request which ensures that the system is reliable.
Security	NR-13	Accessing the system without user login compromises system security; this helps maintain the security of the system.
Portability	NR-14	To ensure portability, the system must be usable in devices such as tablet computers and surface computers.
Reliability	NR-15	The system imposes restrictions on users to perform certain operations. Each role has specific actions that it can perform.This limitations ensure the reliability of the system.
Reliability	NR-16	The system should be able to work as long as the restaurant is open which ensures the reliability of the system.
Security	NR-17	Wireless communication throughout the system will be encrypted using SSLv3 at the application layer and WPA2-PSK at the data link layer. This ensures the system is secure.
Security	NR-18	The WPA2-PSK password used for wireless communication must have a bit strength of at least 80 bits. This ensures the system is secure.
Security	NR-19	The WPA2-PSK password used for wireless communication must be changed every three months. This ensures the system is secure.
Robustness	NR-20	The server shall be capable of supporting no less than 500 concurrent connections from any combination of surface computers, tablets and displays. This ensures that the system is robust.

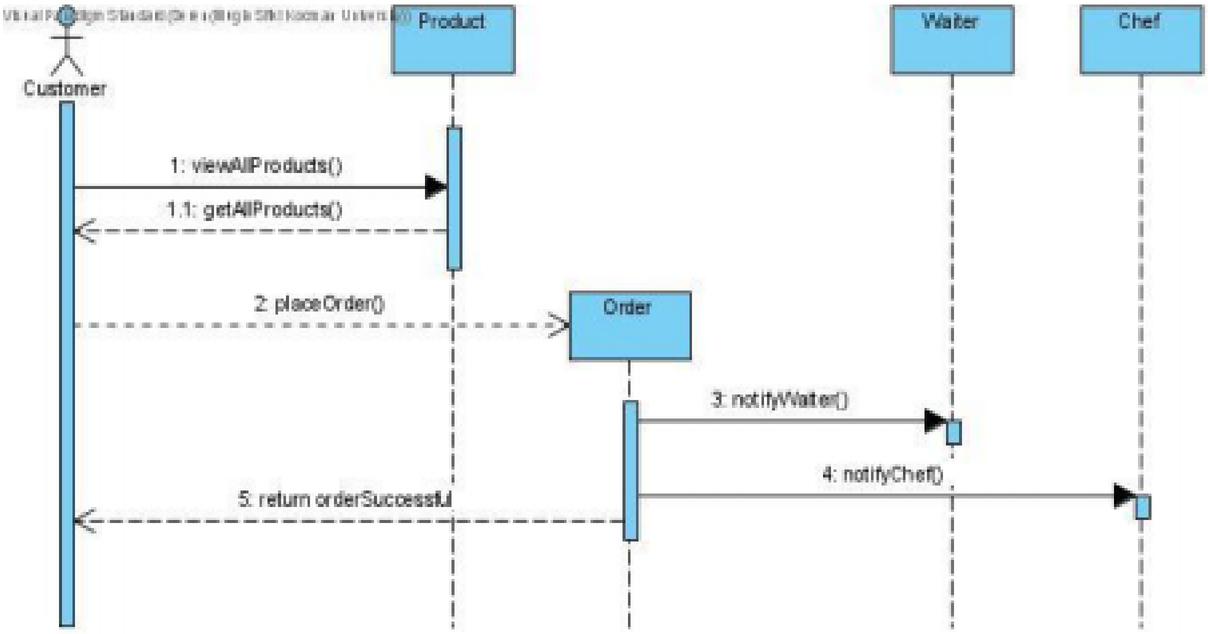
3 System Models

3.1 Class Diagrams

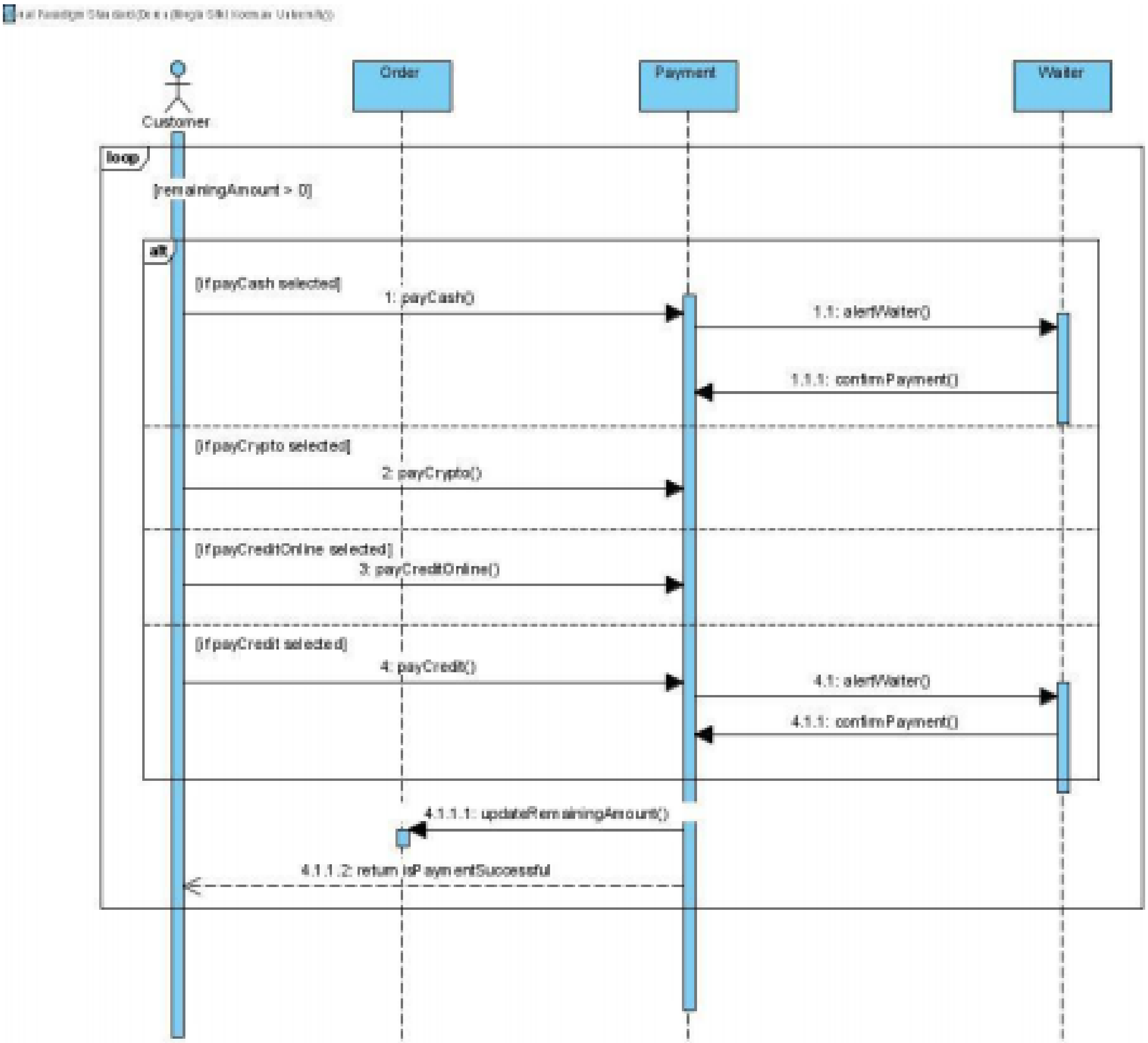


3.2 Sequence Diagrams

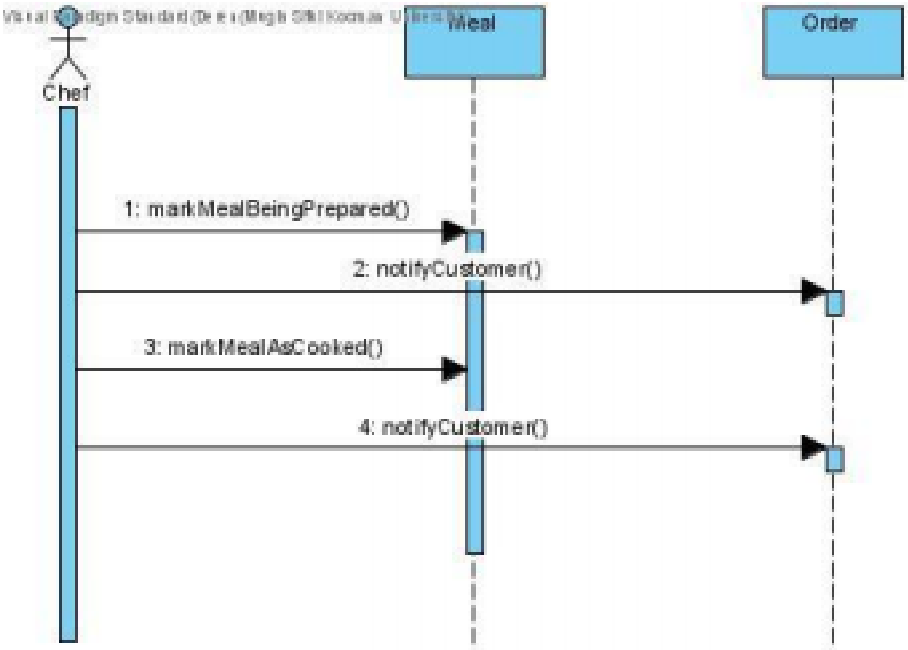
❖ Place Order Sequence Diagram



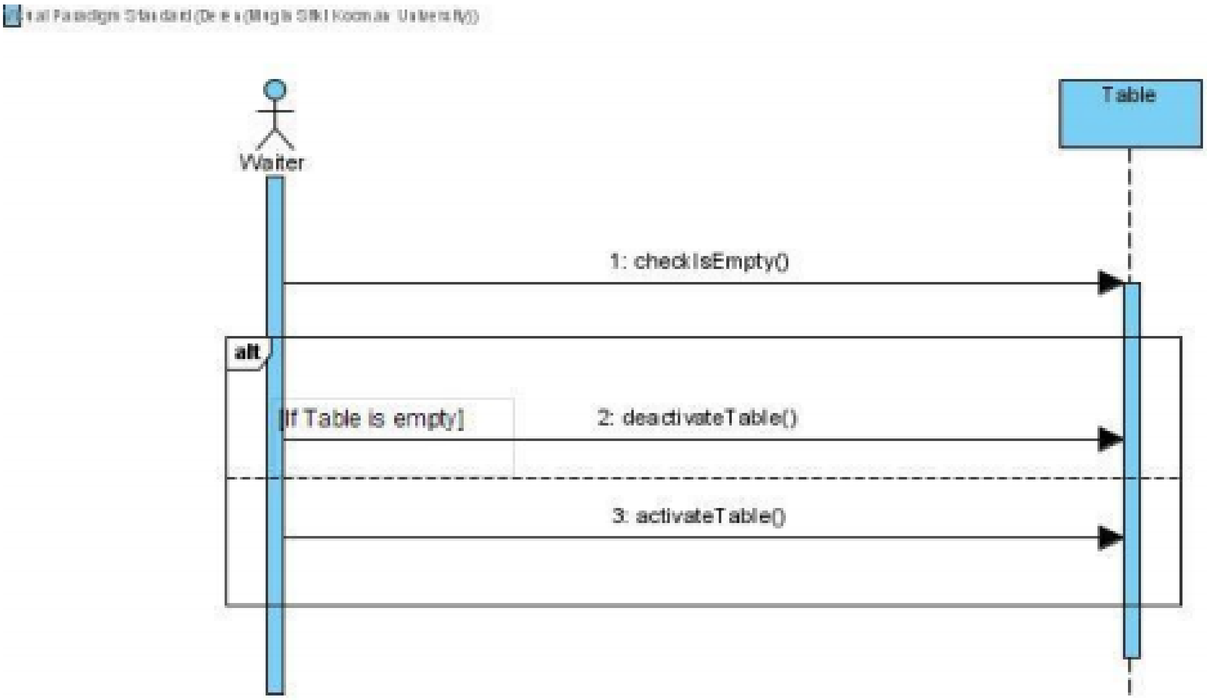
❖ Pay Bill Sequence Diagram



❖ Mark Meal as Cooked Sequence Diagram

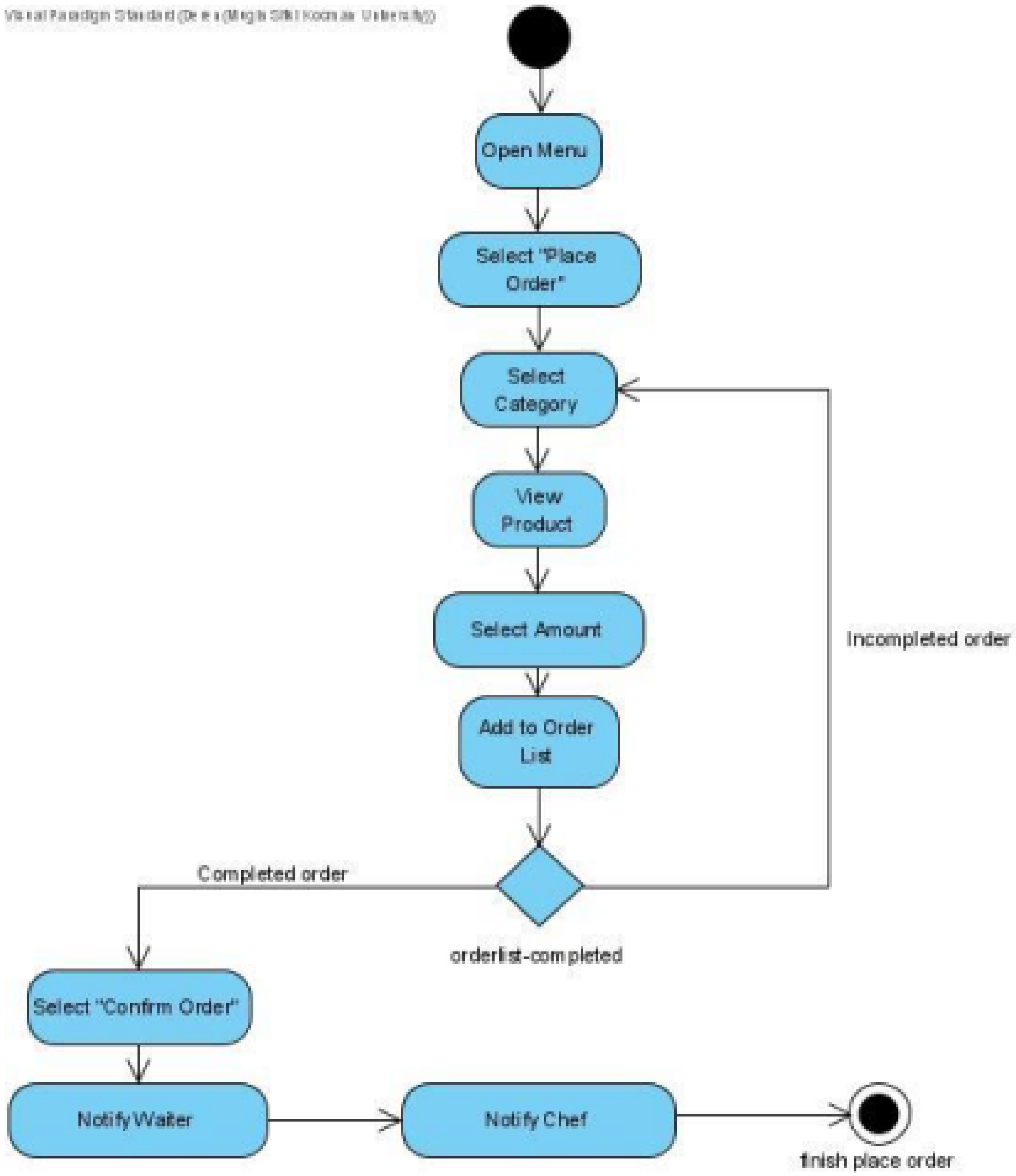


❖ Table Activation Sequence Diagram

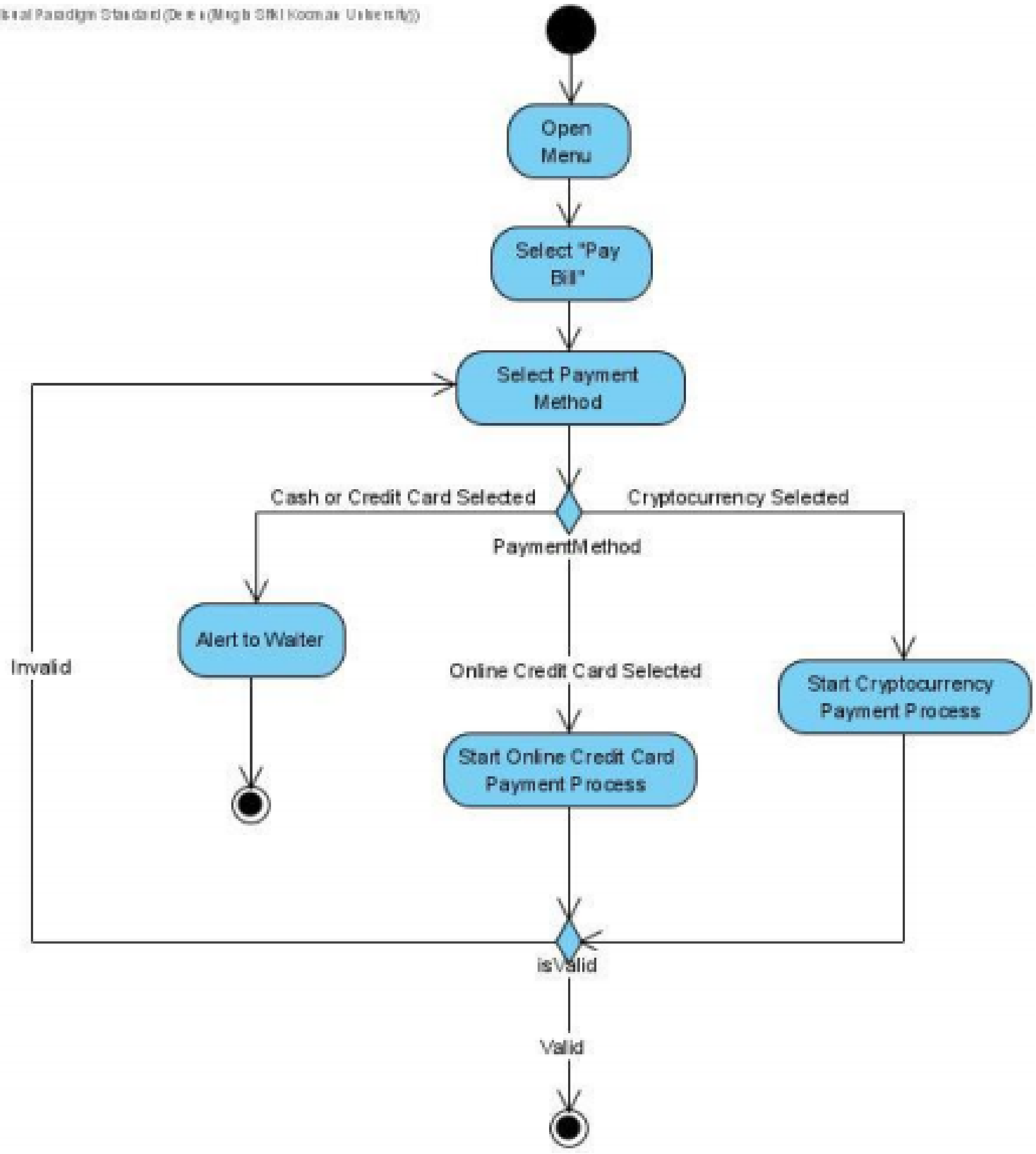


3.3 Activity Diagrams

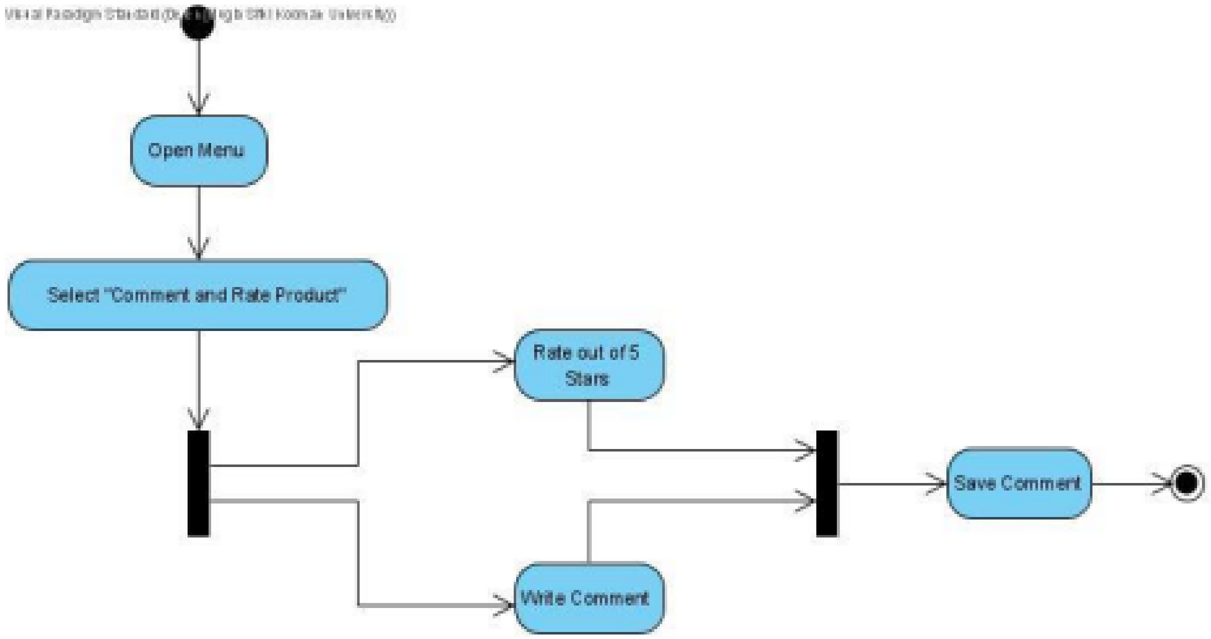
Activity Diagram



❖ Pay Bill Activity Diagram

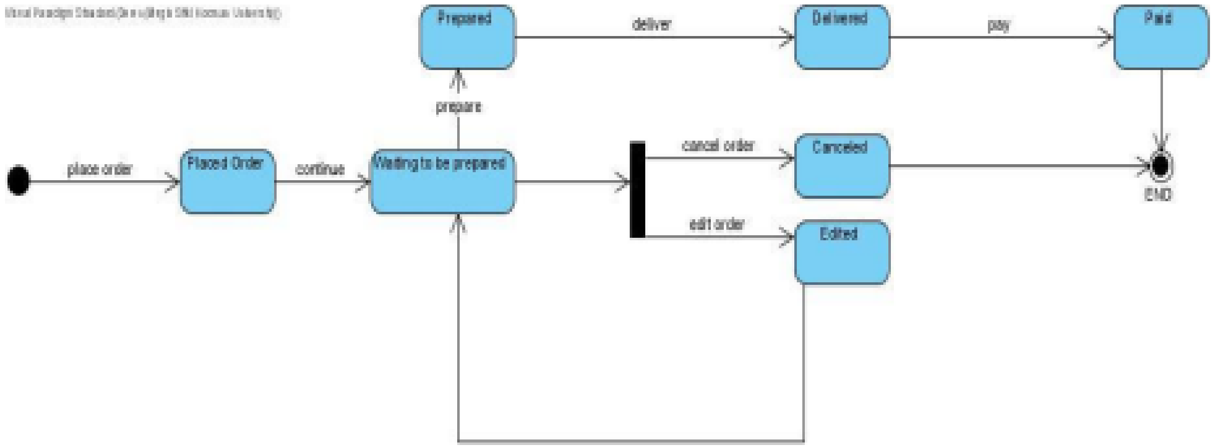

Open Menu
Select "Pay Bill"
Select Payment Method
Cash or Credit Card Selected
Cryptocurrency Selected
PaymentMethod
Alert to Waiter
Invalid
Online Credit Card Selected
Start Cryptocurrency Payment Process
Start Online Credit Card Payment Process
isValid
Valid

❖ Comment/Rate Product Activity Diagram

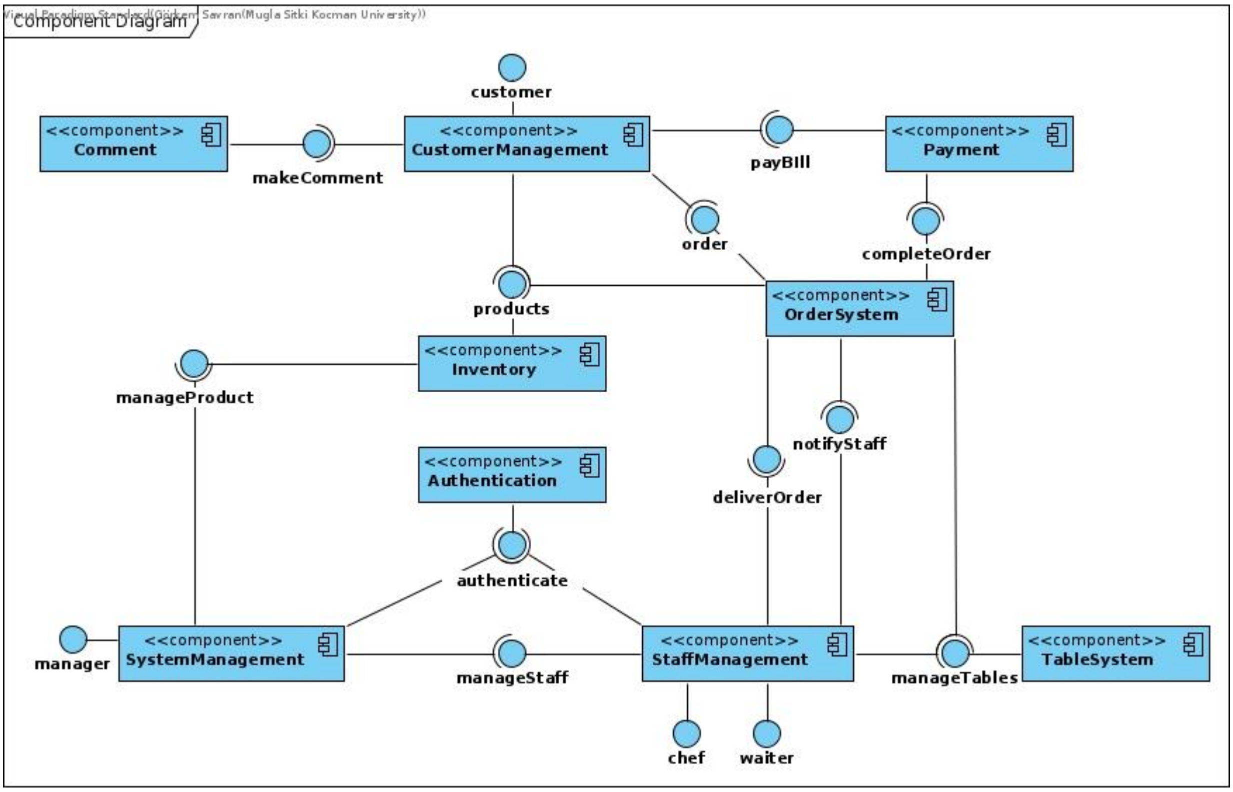


3.4 Statechart Diagrams

❖ Order Class



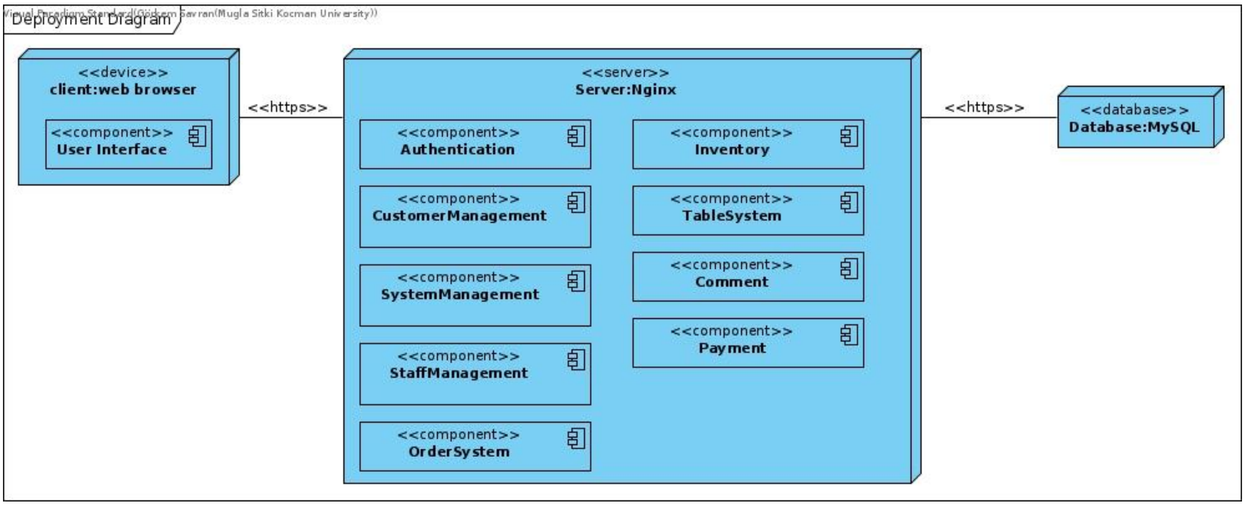
4 Subsystem Decomposition


Component Diagram
Visual Paradigm Standard(Görkem Savran(Mugla Sitki Kocman University))
customer
<<component>> Comment
makeComment
<<component>> CustomerManagement
payBill
<<component>> Payment
order
completeOrder
products
<<component>> OrderSystem
<<component>> Inventory
manageProduct
notifyStaff
<<component>> Authentication
deliverOrder
authenticate
manager
<<component>> SystemManagement
manageStaff
<<component>> StaffManagement
manageTables
<<component>> TableSystem
chef
waiter

Components and List of Classes

- **Comment:** Comment
- **SystemManagement:** Manager
- **CustomerManagement:** Customer
- **OrderSystem:** Order, OrderPaymentStatus
- **Inventory:** Product, Meal, MealStatus, ProductStatus, ProductCategory
- **Authentication:** User
- **StaffManagement:** Chef, Waiter
- **TableSystem:** Table
- **Payment:** Payment, PaymentType

5 Hardware / Software mapping



6 Other Design Concerns (use relevant subsections)

6.1 Concurrency

For 'Place-Order' Sequence Diagram:

Case: If there are two customers sitting at different tables and they want to order at the same time.

Method: Placing two orders from different tables at the same time is simultaneous.The system puts the orders in random order that are placed at the same time. The system adds it to the chief's queue in this order.

For 'Pay Bill' Sequence Diagram:

Case: If there are two customers sitting at different tables and they want to pay at the table at the same time.

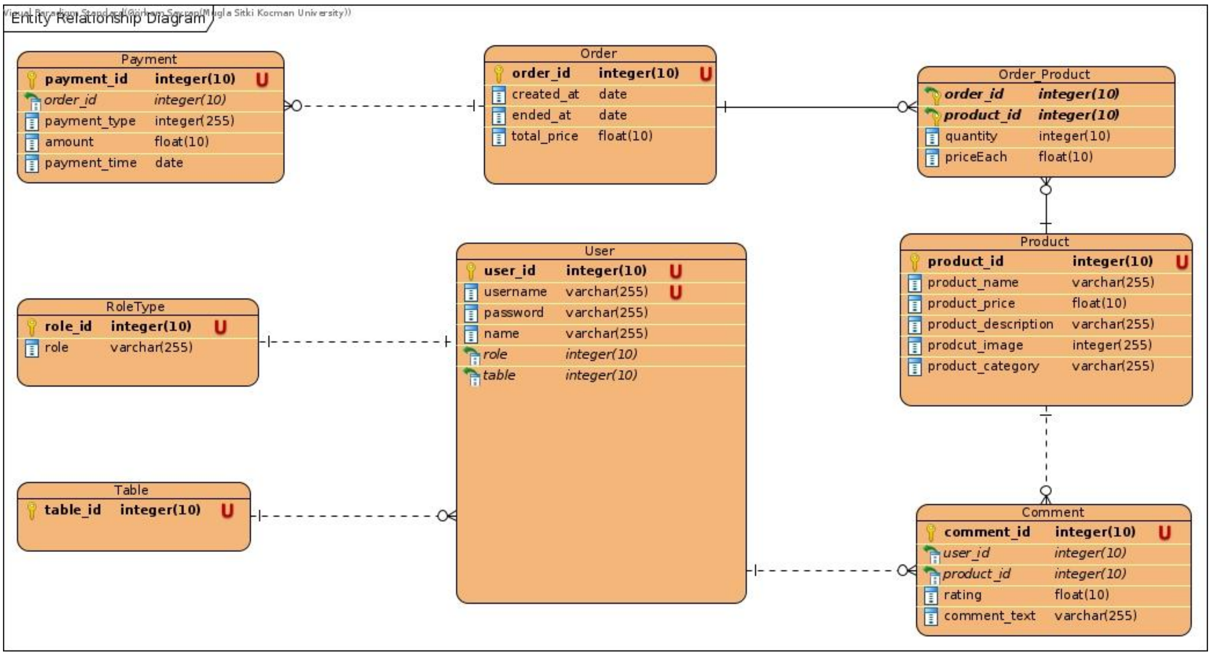
Method: Payment of bills from different tables at the same time is simultaneous. The system queues these processes and then sends alerts to the appropriate waiters.

For 'Table Activation' Sequence Diagram:

Case: If a waiter deactivates a table and a customer orders from that table at the same time.

Method: In this case, priority will be given to the deactivation process, no order will be given and the waiter warns customers to sit at another table.

6.2 Data Management



6.3 Global Resource Handling

Authentication/Authorization Mechanisms

1. **Does the system need authentication?**

   Yes, it is necessary because different people can do different jobs in our project. For example, manager and waiter. While the manager can perform actions such as adding and deleting users to the system, the waiter can manage the tables. Waiter is not supposed to do what the manager does.

2. **If yes, what is the authentication scheme?**

   It will be authenticated using username and password. We will use the Access control list. Thus, users will be able to access or not access functions according to their roles.

3. **What is the user interface for authentication?**

   Each user will be asked for a username and password.

4. **Does the system need a network-wide name server?**

   No, it does not. Because our server will work in the local area network.

5. **What Security Framework is in use?**

   We will use Spring Security for Authentication and Authorization security.

   We will also use the Payment Card Industry Data Security Standard (PCI-DSS) for credit card payment transactions.

Access Matrix

Class/Actor	Customer	Manager	Waiter	Chef

Order	<<create>>
	placeOrder()
	cancelOrder()
	editOrder()
	viewOrderStatus()
	getOrderPaymentStatus()
		getAllOrders()
		getOrder()
		getOrderPaymentStatus()
			getAllOrders()
			getOrder()
			getOrderPaymentStatus()
			markOrderDelivered()
			notifyWaiter()
				notifyChef()
				getAllOrders()
				getOrder()

Product	rateCommentProduct()
	viewAllProducts()
	viewOrderProducts()
		<<create>>
		createProduct()
		editProduct()
		deleteProduct()
		getProduct()
		getAllProducts()		

Table		<<create>>
		createTable()
		editTable()
		deleteTable()
		getTable()
		getAllTables()
			getTable()
			getAllTables()
			checkisEmpty()
			activateTable()
			deactivateTable()	

Customer				

User		<<create>>
		logIn()
		logOut()
		getUsername()
		setUsername()
			logIn()
			logOut()
				logIn()
				logOut()

		getPassword()
		setPassword()
		getRole()
		setRole()
		getName()
		setName()
		createNewUser()
		deleteUser()
		createUser()
		getUser()		

Chef		<<create>>		
Waiter	callWaiter()	<<create>>		

Payment	payCash()
	payCrypto()
	payOnlineCreditCard()
	payCreditCard()
		issueRefund()	alertWaiter()
			receiveCash()
			receiveCreditCard()	

Comment	<<create>>
	rateCommentProduct()
		getAllComments()
		getComment()
		editComment()
		deleteComment()		

Meal				markMealAsCooked()
				markMealBeingPrepared()

Manager		<<create>>		

6.4 Boundary Conditions

Discuss boundary conditions initialization, termination and failure. See below for some questions relevant:

Initialization

- What data needs to be accessed at startup time?
- We need the data "User", "Product" and "Table".

- What services have to be registered?
- User information, information of products and information of tables have to be registered.

- What does the user interface do at start up time?

- User Interface opens menu for Customers and staff panels for Staff members.

Termination

- Are single subsystem is allowed to terminate?

- No, our system has required two subsystems to terminate.

- Are subsystems notified if a single subsystem terminates?

- Yes, if a system other than the user interface is terminated, this situation appears in the user interface.

- How are updates communicated to the database?

- Updates are transmitted to the database in real time via https protocol.

Failure

- How does the system behave when a node or communication link fails?

- Our system transmits the error report to the error report system. The user is prompted to retry the operation.

- How does the system recover from failure?

- When it fails, the system reboots and the backed up data is reloaded into the system.

7 Glossary

Crypto-currency: Crypto-currency is a digital asset designed to work as a medium of exchange wherein individual coin ownership records are stored in a ledger existing in a form of a computerized database using strong cryptography to secure transaction records, to control the creation of additional coins, and to verify the transfer of coin ownership.

WPA2-PSK: WPA2-PSK is the wi-fi security protocol.

8 References

[1]https://www.slideshare.net/Aurnob0071/software-requirements-specification-for-restaurant-management-system

[2]https://www.academia.edu/11259743/Software_Requirements_Specification_Restaurant_Menu_and_Ordering_System

[3]https://github.com/harismuneer/Restaurant-Management-System/blob/master/documents/SRS%20(Latest).pdf

[4]https://spring.io/projects/spring-security

9 Appendix