**Project Use Case Analysis**

*Date : 04-27-2022*

*Project Name -*

***Restaurant Management System***

CS 6359: Object-Oriented Analysis and Design

Instructor: R. Z. Wenkstern

TA: Sultan Alsarra

*Rohit Rajkumar Kriplani*

*Sirisha Satish*

*Chandana Sindhu Gadipudi*

*Greeshma Naga Lakshmi Palisetty*

***Table of Contents***

**I. DOMAIN MODEL** 4

Revision History 4

**II. USE CASE MODELS** 8

Revision History 8

1. Use Case Diagrams 8

1.1. UCD Level 0 9

1.2. UCD Level 1 10

1.3. UCD Level 2 10

1.4. UCD Prioritization (at level 0) 11

2. Use Case Descriptions 12

2.1. UC 1 12

2.2.1 UC1 Description 12

2.2.2 UC 1 «Includes» 13

2.2.3 UC 1 «Extends» 14

2.2.4 UC 1 Non-Functional Requirements 14

2.2. UC 2 15

2.2.1 UC 2 Description 15

2.2.2 UC 2 «Includes» 15

2.2.3 UC 2 «Extends» 15

2.2.4 UC 2 Non-Functional Requirements 15

2.3. UC 3 15

2.3.1 UC 3 Description 16

2.3.2 UC 3 «Includes» 16

2.3.3 UC 3 «Extends» 16

2.3.4 UC 3 Non-Functional Requirements 16

**III. SYSTEM SEQUENCE DIAGRAMS** 18

Revision History 18

1. UC1 System Sequence Diagram v1.0 19

2. UC1 System Sequence Diagram v1.1 20

**III. USE CASE REALIZATION** 21

Revision History 21

1. UC1 REALIZATION V1.0 22

2. UC1 REALIZATION V1.1 23

**IV. DESIGN CLASS DIAGRAM** 24

Revision History 24

1. DESIGN CLASS DIAGRAM V1.0 25

# **I. Domain Model**

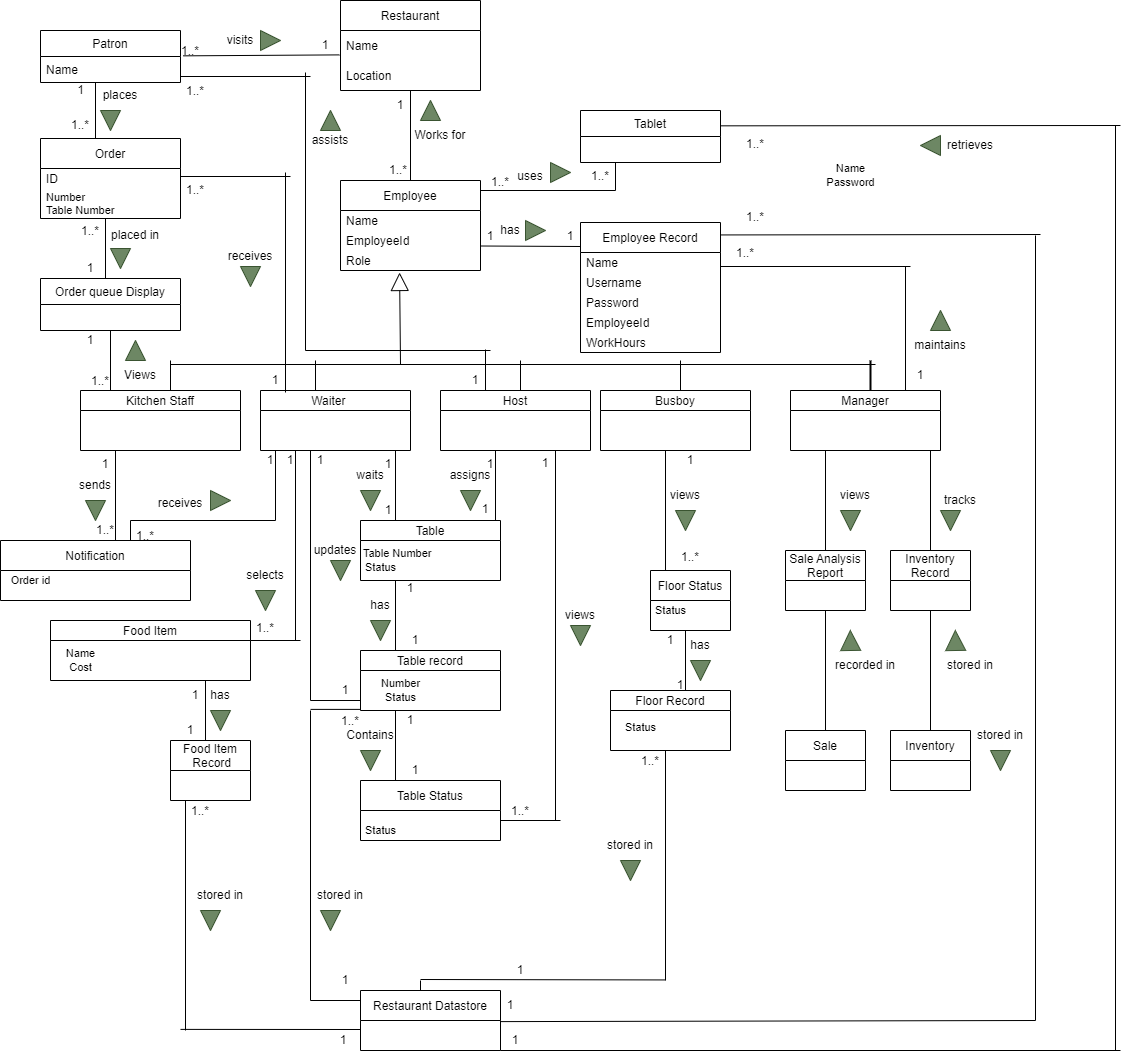
Version 1.0

Date : 04-13-2022

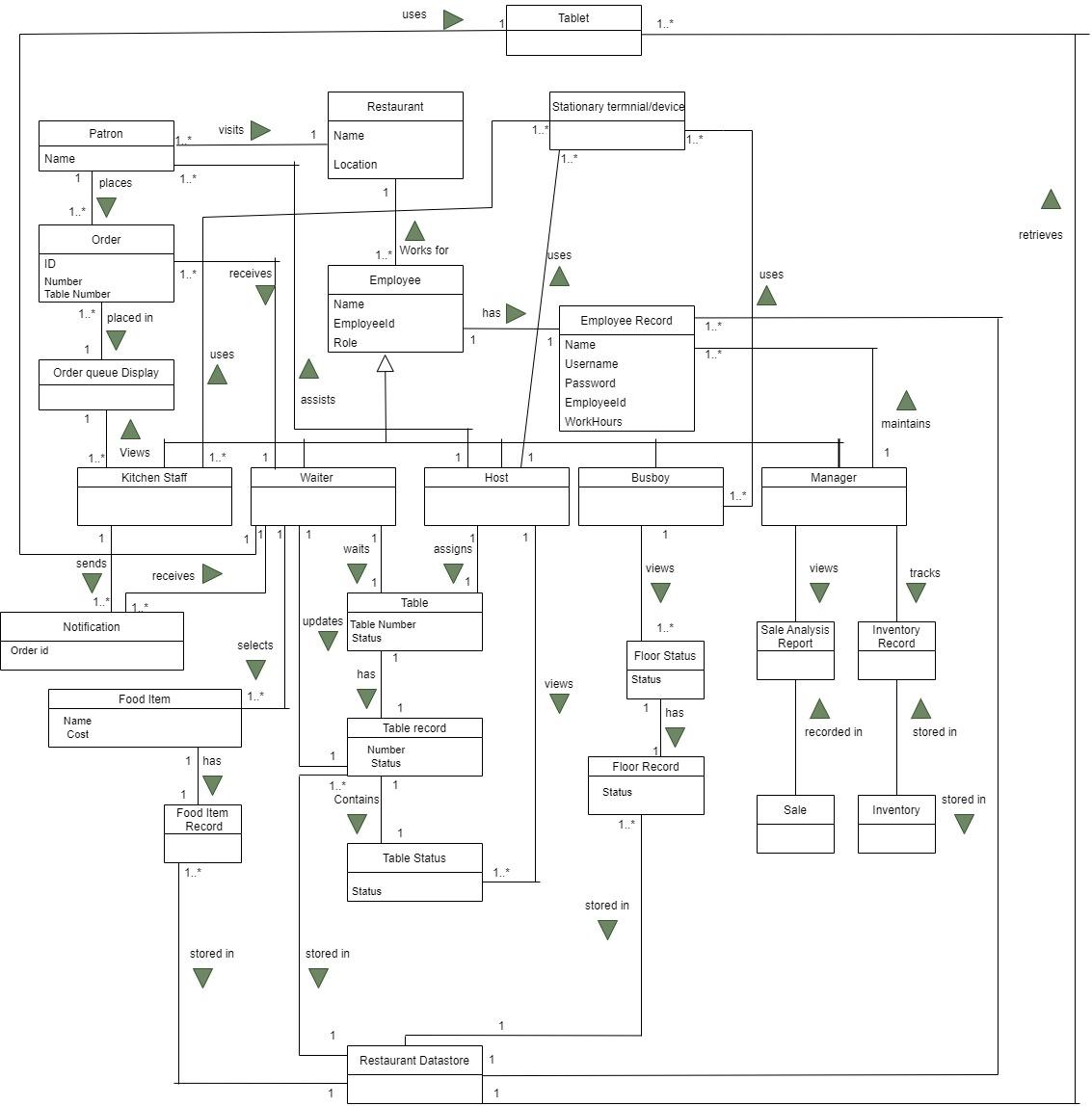
## ***Revision History***

| **Version - description** | **Date** | **Description** | **Author** |
| --- | --- | --- | --- |
| vs 1.0 - Inception draft | 04-12-2022 | First draft. To be refined primarily during elaboration. | … |
| Elaboration 1 draft 1 | 04-20-2022 | Second draft. To be refined during elaboration | … |
| Elaboration 1 draft 2 | … |  | … |
|  |  |  |  |

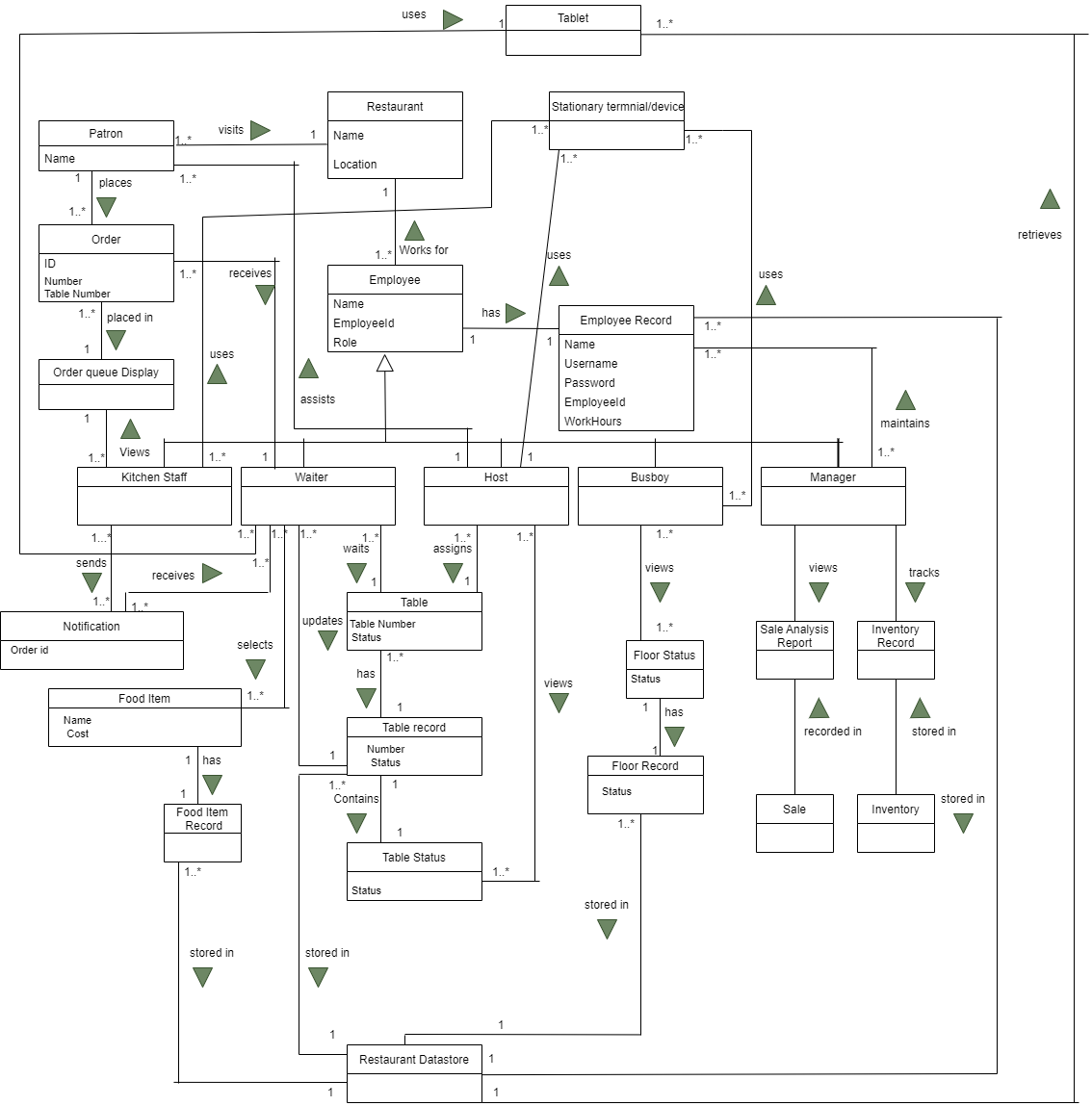
*Domain Model v1.0*



*Domain Model v1.1*

**

*Domain Model v1.2*

**

# **II. Use Case Models**

Version 1.0

Date 04-13-2022

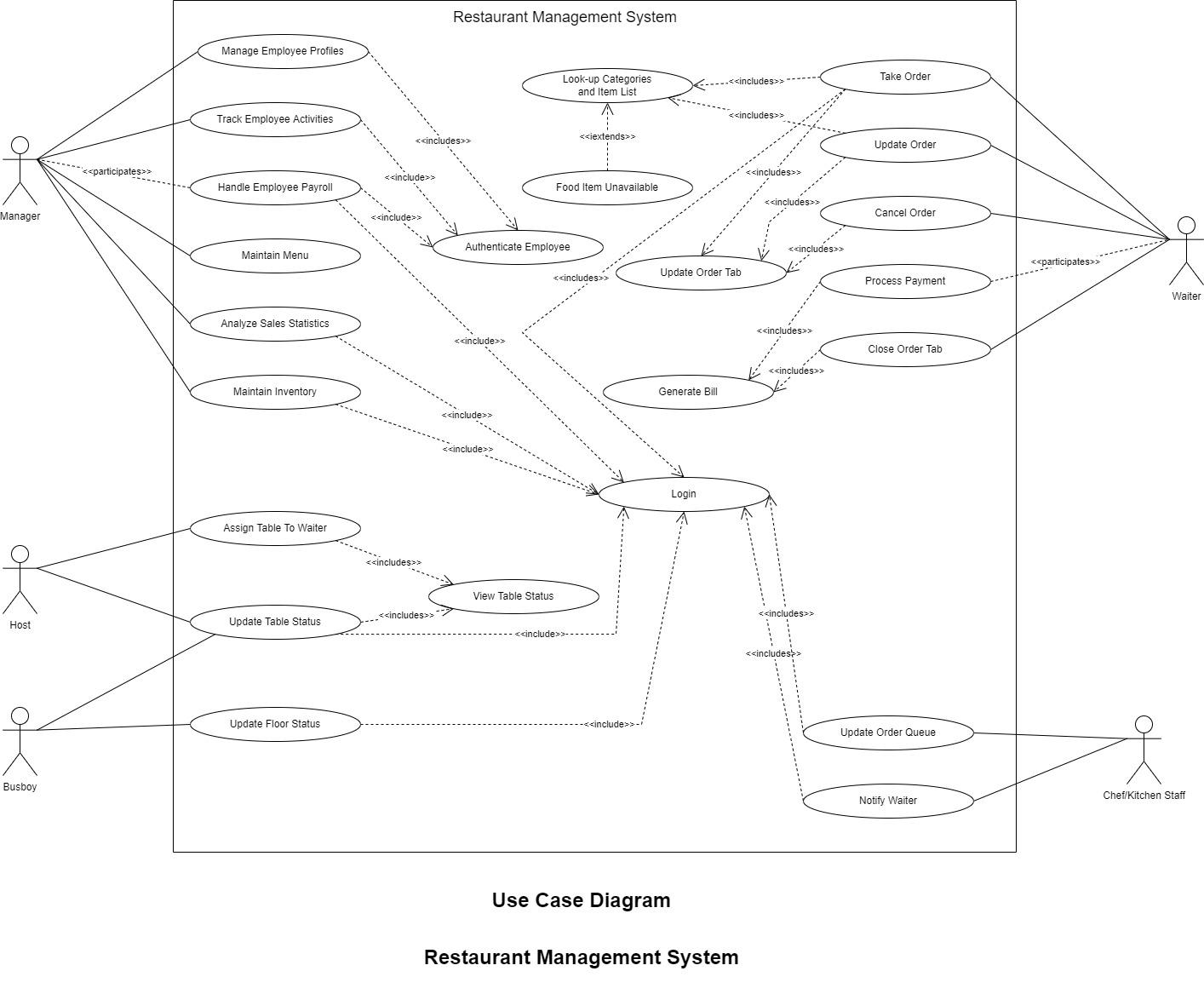
## ***Revision History***

| **Version - description** | **Date** | **Description** | **Author** |
| --- | --- | --- | --- |
| vs 1.0 - Inception draft | 04-12-2022 | First draft. To be refined primarily during elaboration. | … |
| Elaboration 1 draft 1 | 04-20-2022 | Second draft. To be refined during elaboration | … |
| Elaboration 1 draft 2 | … |  | … |
|  |  |  |  |

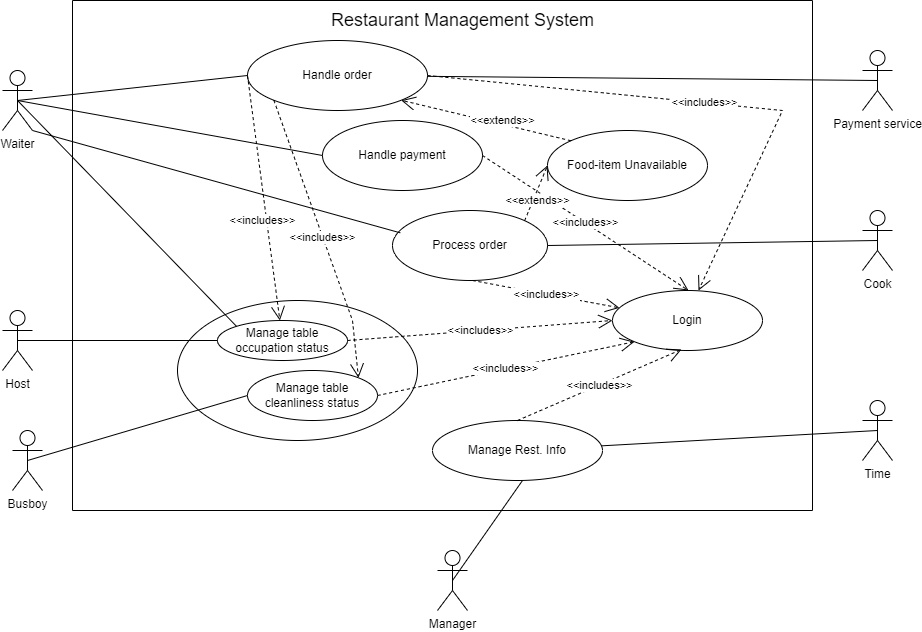
## ***Use Case Diagrams***

### **UCD**

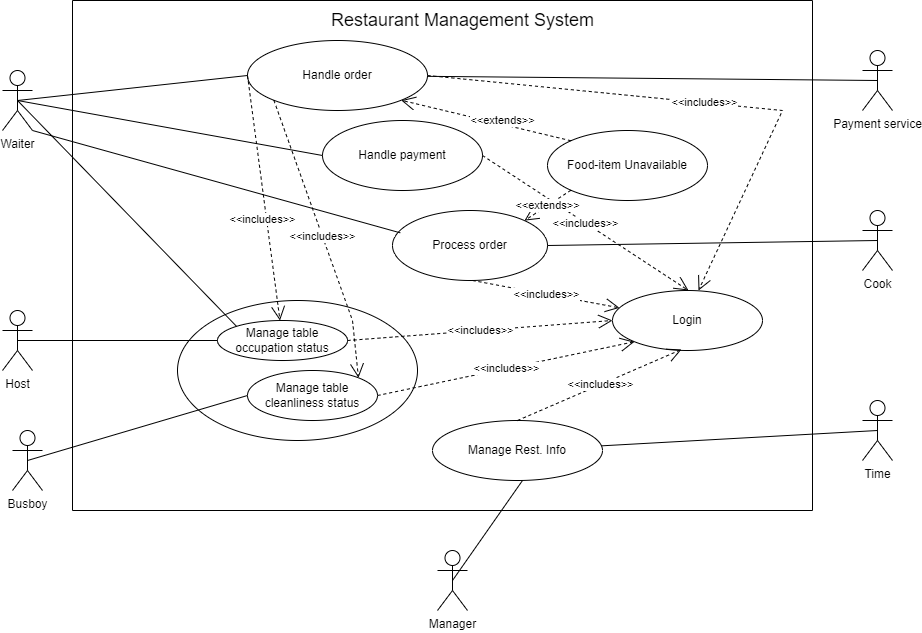
*Use case Diagram v1.0*



*Use case Diagram v1.1*

**

*Use case Diagram v1.2*



### **UCD Prioritization**

| **Fully dressed** | **Casual** | **Brief** |
| --- | --- | --- |
| Handle order | Process order | Handle payment |
|  |  | Manage table status |
|  |  | Manage cleanliness status |
|  |  | Login |

## 

## ***Use Case Descriptions***

### **UC 1 *(Handle order)***

### **UC1 Description**

| Use Case: Handle Order |
| --- |
| ID: 1 |
| Brief Description:  Handling of order by waiter and kitchen staff. It includes all the activities right from taking the order to processing the payment. |
| Primary Actors:  Waiter |
| Secondary Actors:  Kitchen Staff |
| Preconditions:  Host has assigned the table to the waiter |
| Main Flow:  Waiter logs in the system  System greets and displays the tables assigned to the waiter  Waiter views the floor status of those tables.  Waiter selects the  table tab  System opens the table tab  Waiter selects Add item option  System displays the list of item categories available  Waiter selects a category from the list.  System displays the food items offered in that category  Waiter selects the desired food item  **Extension Point:** Selected food-item unavailable  System adds the selected food item to cart.  12.System redirects to table screen        13.Waiter repeats steps from 4 to 11, until patron is done ordering  14.System sends the order list to the kitchen computer  15.Kitchen staff views the order list from kitchen computer |
| Postconditions:  The order and food-items ordered are updated in the restaurant datastore for inventory and sales analysis. |
| Alternative Flows:  None |

#### 

#### 

#### **UC 1 «Includes» *(Manage table status)***

| Use Case: Manage Table Status |
| --- |
| ID: 1.1 |
| Brief Description:  Handling of order by waiter and kitchen staff. It includes all the activities right from taking the order to processing the payment. |
| Primary Actors:  Host, Busboy |
| Secondary Actors:  Waiter |
| Preconditions:  The user must be logged in the system |
| Main Flow:  Host checks the floor status to see what tables are vacant  System displays the floor status by showing all occupied and vacant tables  Host provides the table to the patron  Host changes the table status from Vacant to Occupied  Host assign the table to a waiter  Once the patron leaves, either waiter or manager will change the table status  System changes the occupancy status of table from occupied to vacant  System also changes the cleanliness status of table to dirty  Busboy views the table statuses to see which tables are dirty  Busboy cleans the table and changes the status to clean  System now shows the table to be clean and vacant |
| Postconditions:  Table status is updated which will reflect in floor status, which in turn will be viewed by actors to clean and assign table to new patron |
| Alternative Flows:  None |

#### 

#### **UC 1 «Extends»**

| Use Case: Food-item Unavailable |
| --- |
| ID: 1.2 |
| Brief Description:  The desired food-item that user wants to add is not available |
| Primary Actors:  Waiter |
| Secondary Actors:  None |
| Preconditions:  The waiter selected a category  The waiter tried adding desires item |
| Main Flow:  Waiter enters the category  System displays food-item list in that category  Waiter selects food-item from the list  System displays an exception that the food-item is unavailable due to certain reasons (out of stock, etc.)  Waiter views the error  System redirects waiter to table screen  Waiter asks patron and selects category again for another order |
| Postconditions:  System tells the patron that the food-item is unavailable, and the Waiter notifies the patron of the same. |
| Alternative Flows:  None |

**UC 1 Non-Functional Requirements**

* Accessibility
* Availability
* Reliability
* Operability
* Integrability

### **UC 2 *(Process order)***

### **UC 2 Description**

This use case involves processing of the food-items ordered by patrons. This processing of order includes steps like:

Waiter adding the item to the order tab in order for kitchen staff to view, system updating the order queue that is viewed and maintained by kitchen staff, kitchen staff viewing the order queue for new orders, kitchen staff picking up the order to be prepared in first-in, first-out basis, when the order is prepared/ready, wait staff will be notified as soon as he/she logs in

For the process order use case, Waiter and Kitchen Staff are the actors involved.

Waiter’s duties include - adding items ordered by patron into the system, checking the notifications when the order is ready (and serving).

Kitchen Staff/Chef’s duties include - viewing the newly added orders in the order queue, picking up and preparing orders from the order queue on a first-in, first-out basis.

The system handles tasks like - adding the food-item order to the order queue, notifying the correct waiter when the food is prepared

#### **UC 2 «Includes»**

Login

#### **UC 2 «Extends»**

Food-item unavailable

#### **UC 2 Non-Functional Requirements**

* Accessibility
* Availability
* Reliability
* Operability
* Integrability

### **UC 3 *(Handle payment)***

### **2.3.1 UC 3 Description**

System handles all the payment processing, wait staff will participate in this and will perform duties like - check the total order value from the system and provide the bill to the patron, ask the patron for the payment mode, complete the payment. Since the payment mode is electronic, there will be involvement of third party payment service, for example, to validate credit/debit cards for payment.

#### **2.3.2 UC 3 «Includes»**

Login

#### **2.3.3 UC 3 «Extends»**

Food item unavailable

#### **2.3.4 UC 3 Non-Functional Requirements**

* Accessibility
* Availability
* Reliability
* Operability
* Integrability

### **2.3 UC 3 *(Manage table status)***

### **2.3.1 UC 3 Description**

This use-case includes managing/maintaining/updating table status. Table status can be categorized further as Occupancy Status and Cleanliness Status. Occupancy Status denotes if the table is occupied by any patron or it is vacant at the given instance. Cleanliness Status denotes if the table is dirty or clean and ready to be assigned to a patron. A host or waiter is responsible for updating the occupancy status as soon as it is assigned to a patron or the patron leaves and it is ready for a new customer. A busboy is the only one responsible for updating the cleanliness status of a table

#### **2.3.2 UC 3 «Includes»**

Login

#### **2.3.3 UC 3 «Extends»**

#### **2.3.4 UC 3 Non-Functional Requirements**

* Accessibility
* Availability
* Reliability
* Operability
* Integrability

### **UC 3 *(Manage restaurant data)***

### **2.3.1 UC 3 Description**

This use-case includes all the activities involved with the restaurant datastore, like managing and tracking inventory, tracking employee activities, generating and viewing sales analysis reports in order to track revenue, other analysis like Revenue and revenue percentage per menu item, Menu item popularity, Personnel efficiency, etc. The actors involved in this use case are manager and the time.

#### **2.3.2 UC 3 «Includes»**

Login

#### **2.3.3 UC 3 «Extends»**

#### **2.3.4 UC 3 Non-Functional Requirements**

* Accessibility
* Availability
* Reliability
* Operability
* Integrability

### **UC 3 *(Login)***

### **2.3.1 UC 3 Description**

Login is handled by the host, waiter, manager, busboy, and the kitchen staff.It involves logging in to the system with their respective credentials. Once every employee logs in, each of them is displayed with their respective role personalized home screens and the screen will be refreshed automatically when needed. By logging in unauthorized activities are restricted depending on employee/user roles.

#### **2.3.2 UC 3 «Includes»**

#### **2.3.3 UC 3 «Extends»**

#### **2.3.4 UC 3 Non-Functional Requirements**

* Accessibility
* Availability
* Reliability
* Operability
* Integrability

# **III. System Sequence Diagrams**

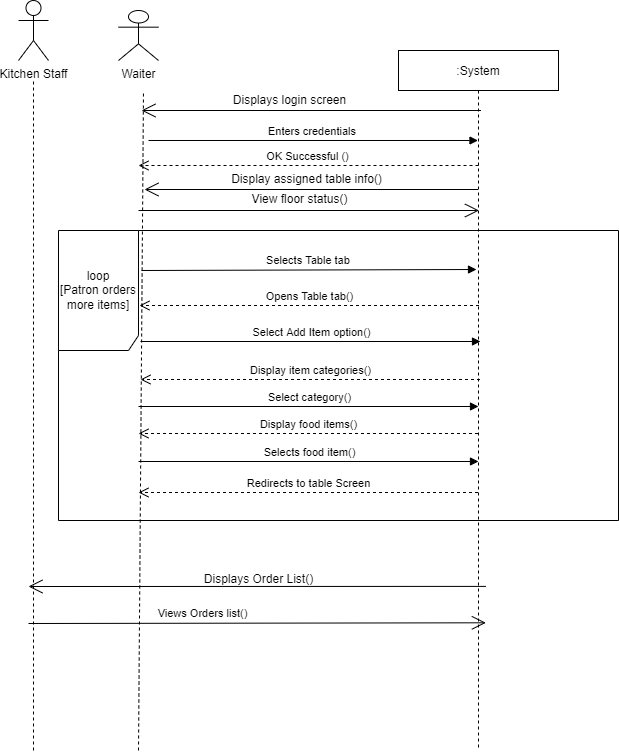
Version 1.0

Date – 04-19-2022

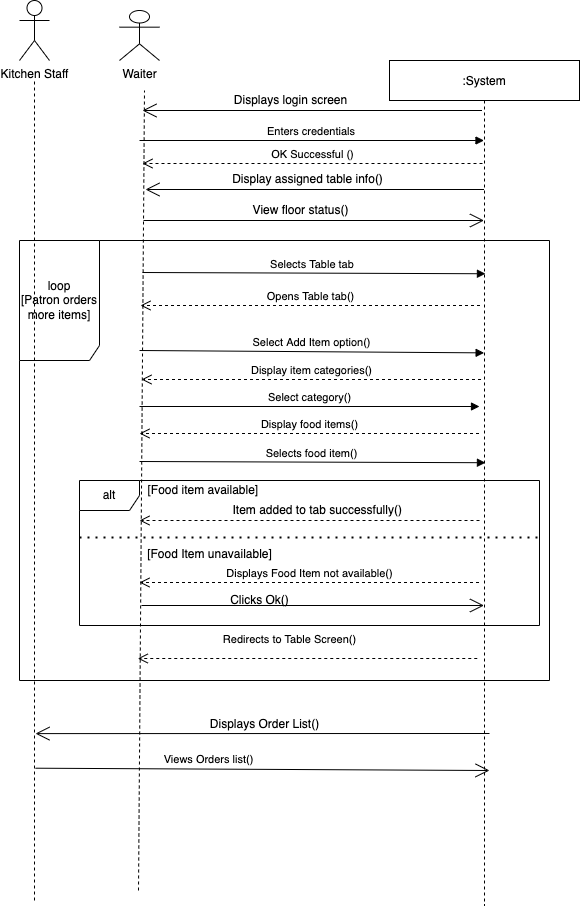
## ***Revision History***

| **Version - description** | **Date** | **Description** | **Author** |
| --- | --- | --- | --- |
| vs 1.0 - Inception draft | 04-20-2022 | First draft. To be refined primarily during elaboration. | … |
| Elaboration 1 draft 1 | 04-25-2022 | … | … |
| Elaboration 1 draft 2 | … | … | … |
|  |  |  |  |

## ***1.UC1 System Sequence v1.0***



## ***2.UC1 System Sequence v1.1***



# IV. Use Case Realizations

# 

# Version 1.0

# Date 04/27/2022

# 

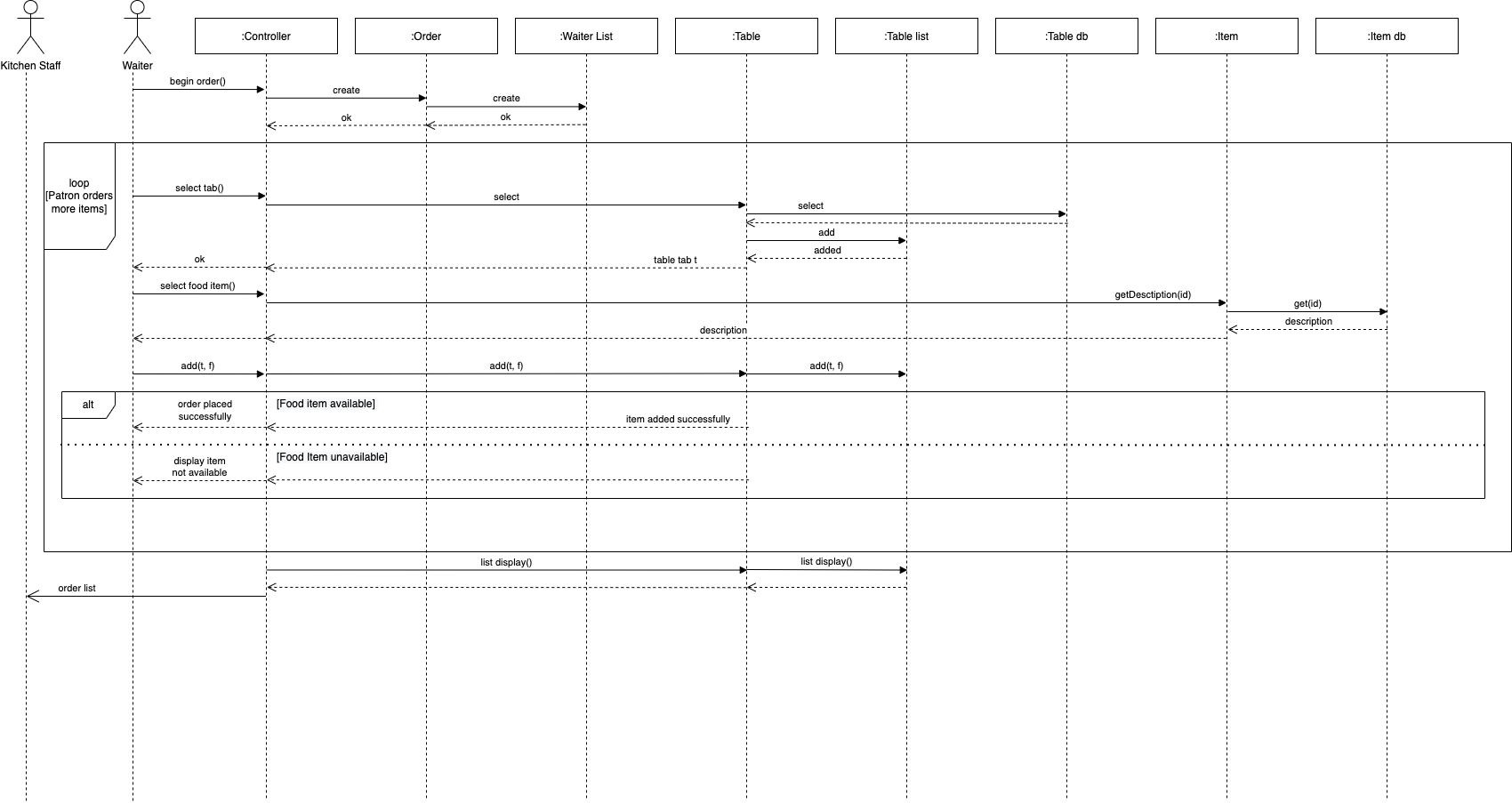
## Revision History

| Version - description | Date | Description | Author |
| --- | --- | --- | --- |
| vs 1.2 Inception draft | 04-27-2022 | first draft to be defined primarily during elaboration | … |
| Elaboration draft 1 | 05-12-2022 | … | … |
| Elaboration draft 2 | … | … | … |
|  |  |  |  |

# 

# 

# *1.* *UC1 Realization v1.0*



# *2.* *UC1 Realization v1.1*

# 

# V. Design Class Diagram

# 

# Version 1.0

# Date 05/12/2022

# 

## Revision History

| Version - description | Date | Description | Author |
| --- | --- | --- | --- |
| vs 1.1 Inception draft | 05-12-2022 | first draft to be defined primarily during elaboration | … |
| Elaboration draft 1 | … | … | … |
|  |  |  |  |

# *1.* *Design Class Diagram v1.0*

# 

# **Project Participation Report**

Team: 2

Submission Date: 05-12-2022

Deliverable: Updated Deliverables 1, 2 and 3, Design Class Diagram, Project Participation Report

**Areas/questions for discussion**

**Team Members participation**

Name 1: 25 %

Name 2: 25 %

Name 3: 25 %

Name 4: 25 %

**Member 1 (Rohit Kriplani)**

| **Tasks Completed** | **Hours** |
| --- | --- |
| Initially discussed the problem statement and requirements | 3 |
| Made a rough draft of Design Class Diagram individually | 2 |
| Discussed, analyzed and finalized the DCD among all the rough drafts made by other team members | 2.5 |
| Assisted in final drafting of DCD | 0.5 |
| Coordinated the updations to be made in previous deliverables to team members responsible for same | 1.5 |
| Went over the final draft of DCD and updations in previous deliverables and made or coordinated necessary changes | 1 |
| Created the final report | 1.5 |

Sign: Rohit Rajkumar Kriplani

**Member 2 (Sirisha Satish)**

| **Tasks Completed** | **Hours** |
| --- | --- |
| Initially discussed the problem statement and requirements | 3 |
| Made a rough draft of Design Class Diagram individually | 2 |
| Discussed, analyzed and finalized the DCD among all the rough drafts made by other team members | 2.5 |
| Made the final draft of DCD | 2 |
| Reviewed the final changes of deliverable 3 | 1 |
| Made the final necessary changes in DCD draft after review from peers | 1 |
| Assisted in report documentation | 0.5 |

Sign: Sirisha Satish

**Member 3 (Greeshma Naga Lakshmi Palisetty)**

| **Tasks Completed** | **Hours** |
| --- | --- |
| Initially discussed the problem statement and requirements | 3 |
| Made a rough draft of Design Class Diagram individually | 2 |
| Discussed, analyzed and finalized the DCD among all the rough drafts made by other team members | 1 |
| Updated previous deliverable i.e. deliverable 3 (Use Case realization) | 3 |
| Reviewed the final draft of DCD and communicated the changes required | 1 |

Sign: Greeshma Naga Lakshmi Palisetty

**Member 4 (Chandana Sindhu Gadipudi)**

| **Tasks Completed** | **Hours** |
| --- | --- |
| Initially discussed the problem statement and requirements | 3 |
| Made a rough draft of Design Class Diagram individually | 2 |
| Discussed, analyzed and finalized the DCD among all the rough drafts made by other team members | 1 |
| Updated previous deliverable i.e. deliverable 3 (Use Case realization) | 3 |
| Reviewed the final draft of DCD and communicated the changes required | 1 |

Sign: Chandana Sindhu Gadipudi