ASSIGNMENT #03

SOFTWARE CONSTRUCTION

GROUP MEMBERS:

Ali Muhammad (Group lead)

Muhammad Faizan Kiani

Muhammad Ali Sultan

Submitted to :Mam Onaiza Maqbool

Contents

[1 Domain Model 3](#_Toc181039132)

[2 Data Dictionary 4](#_Toc181039133)

[3 System Sequence Diagram 7](#_Toc181039134)

[3.1 Muhammad Faizan Kiani (Create Event) 7](#_Toc181039135)

[3.2 Muhammad Ali Sultan 8](#_Toc181039136)

[3.3 Ali Muhammad 9](#_Toc181039137)

[4 Usecases 10](#_Toc181039138)

[4.1 Muhammad Faizan Kiani 10](#_Toc181039139)

[4.1.1 Create Event 10](#_Toc181039140)

[4.1.2 Use Case: Announce Payment Deadline 12](#_Toc181039141)

[4.1.3 Select Items 14](#_Toc181039142)

[4.2 Muhammad Ali Sultan 16](#_Toc181039143)

[4.2.1 View Event 16](#_Toc181039144)

[4.2.2 Select Gifts 18](#_Toc181039145)

[4.3 Ali Muhammad 20](#_Toc181039146)

[4.3.2 Modify Selected Items 20](#_Toc181039147)

[4.3.2 Make Payment 22](#_Toc181039148)

# 1 Domain Model

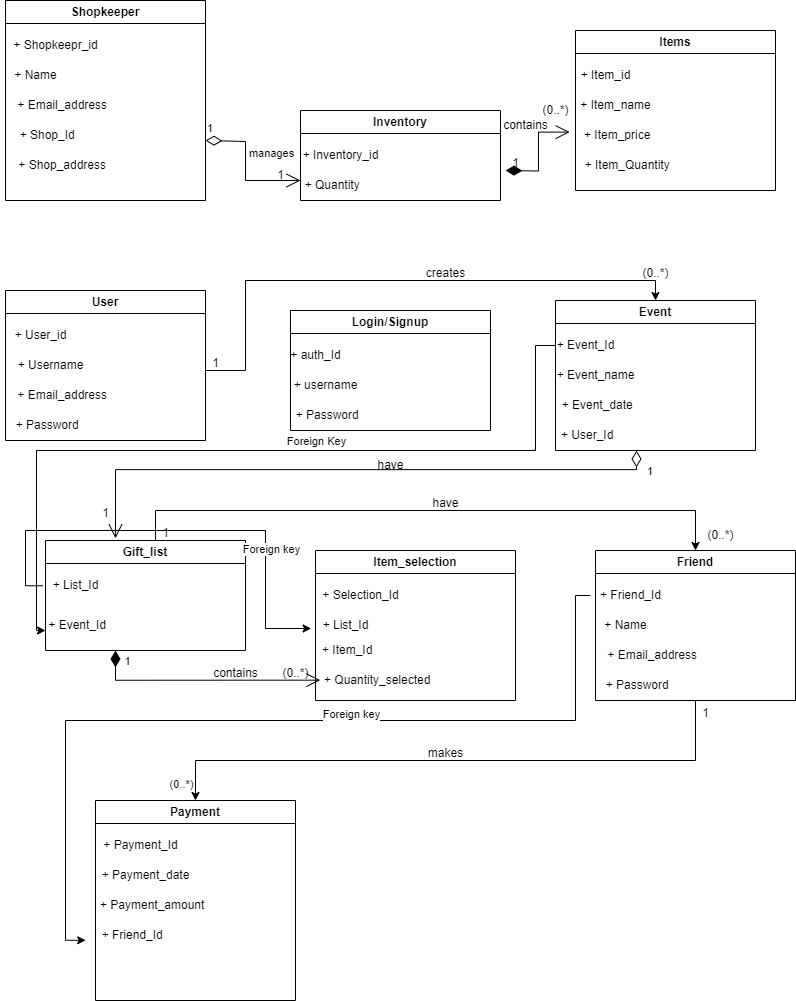


Figure 1

# 2 Data Dictionary

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field Name** | **Description** | **Data Type** | **Size** | **Constraints** | **Default Value** |
| **Shopkeeper** |  |  |  |  |  |
| shopkeeperID | Unique identifier for the shopkeeper | Integer |  | Primary Key, Not Null | 1 |
| name | Name of the shopkeeper | Varchar | 100 | Not Null |  |
| email | Email address of the shopkeeper | Varchar | 150 | Not Null, Unique |  |
| contactNumber | Contact number of the shopkeeper | Varchar | 15 |  |  |
| shopID | Foreign key referencing the Shop entity | Integer |  | Foreign Key | Not Null |
| **Inventory** | Centralized inventory for the shop |  |  |  |  |
| Inventory\_ID | Unique identifier for the central inventory | Integer |  | Primary Key, Not Null | 1 |
| **Item** |  |  |  |  |  |
| Item\_ID | Unique identifier for each gift item | Integer |  | Primary Key, Not Null |  |
| Item\_Name | Name of the gift item | Varchar | 150 | Not Null |  |
| price | Price of the gift item | Decimal |  | Not Null, Check >= 0 | 0.00 |
| Item\_quantity | Available quantity of the gift item | Integer |  | Not Null, Check >= 0 | 0 |
| **User** |  |  |  |  |  |
| User\_ID | Unique identifier for each registered user | Integer |  | Primary Key, Not Null |  |
| name | Username of the registered user | Varchar | 50 | Not Null, Unique |  |
| email | Email address of the registered user | Varchar | 150 | Not Null, Unique |  |
| password | Password of the registered user | Varchar | 100 | Not Null |  |
| **Login/Signup** |  |  |  |  |  |
| Auth\_ID | Unique identifier for each login/signup | Integer |  | Primary Key, Not Null |  |
| name | Username used for login/signup | Varchar | 50 | Not Null |  |
| password | Password used for login/signup | Varchar | 100 | Not Null |  |
| **Event** |  |  |  |  |  |
| Event\_ID | Unique identifier for each event | Integer |  | Primary Key, Not Null |  |
| Event\_Name | Name of the event | Varchar | 100 | Not Null |  |
| Event\_Date | Date of the event | Date |  | Not Null |  |
| User\_ID | Foreign key referencing the Registered User | Integer |  | Foreign Key | Not Null |
| **Gift List** |  |  |  |  |  |
| List\_ID | Unique identifier for each gift list | Integer |  | Primary Key, Not Null |  |
| Event\_ID | Foreign key referencing the Event | Integer |  | Foreign Key | Not Null |
| **Item Selection** |  |  |  |  |  |
| Selection\_ID | Unique identifier for each item selection | Integer |  | Primary Key, Not Null |  |
| listI\_D | Foreign key referencing the Gift List | Integer |  | Foreign Key | Not Null |
| Item\_ID | Foreign key referencing the Gift Item | Integer |  | Foreign Key | Not Null |
| Quantity\_Selected | Quantity of the gift item selected | Integer |  | Not Null, Check >= 0 | 1 |
| **Friend** |  |  |  |  |  |
| friendID | Unique identifier for each friend | Integer |  | Primary Key, Not Null |  |
| name | Name of the friend | Varchar | 100 | Not Null |  |
| email | Email address of the friend | Varchar | 150 | Not Null, Unique |  |
| password | Password for the friend's account | Varchar | 100 | Not Null |  |
| **Payment** |  |  |  |  |  |
| Payment\_ID | Unique identifier for each payment | Integer |  | Primary Key, Not Null |  |
| Payment\_Date | Date when the payment was made | Date |  | Not Null |  |
| amount | Amount paid | float |  | Not Null | 0.00 |
| Friend\_ID | Foreign key referencing the Friend who made payment | Integer |  | Foreign Key | Not Null |
| List\_ID | Foreign key referencing the Gift List from where item was selected | Integer |  | Foreign Key |  |

# 3 System Sequence Diagram

## 3.1 Muhammad Faizan Kiani (Create Event)

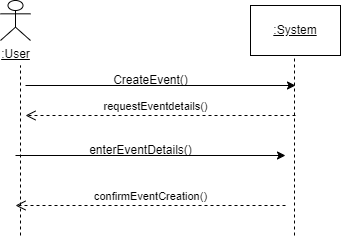


Figure 2

## 3.2 Muhammad Ali Sultan

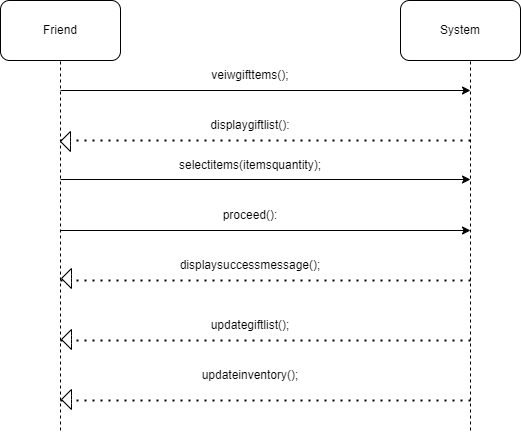


Figure 3

## 3.3 Ali Muhammad

# 4 Usecases

# 4.1 Muhammad Faizan Kiani

### 4.1.1 Create Event

**Use Case ID**

UC-04

**Use Case Name**

Create Event

**Actors**

**Primary Actor:** Event Owner

**Secondary Actor:** None

**Stakeholders & Interests:**

**User (Event Owner):** Wants to create an event and set up a gift registry.

**Friends:** Will be selecting gifts from the event owner's registry.

**Shopkeeper:** Provides items for selection.

**Preconditions**

The event owner is logged into the system.

The items list is available for the user to view and select items from.

**Postconditions**

A new event and registry are created.

Friends can view the event/registry and choose gifts.

**Main Success Scenario**

The user chooses the option to create a new event.

The system asks for event details (name, date).

The event owner enters the required details.

The system saves the event and confirms the creation.

**Alternate Scenarios**

* **Scenario 1:** (Occurs after Step 3)  
  **Condition:** If the event owner doesn’t enter all required information (like event name or date),  
  **Action:** The system asks the event user to provide the missing information and prevents them from moving to the next step.
* **Scenario 2:** (Occurs after Step 2)  
  **Condition:** If the event owner clicks on "Cancel Event,"  
  **Action:** The system does not save any information and returns the user to the dashboard/homepage or previous page.

**Input Section**

Event\_Id

Event\_name

Event\_date

User\_Id

**Output Section**

**Confirmation Message** (success or failure)

**Special Requirements**

The event name must be unique within the user's account.

The event date should be valid (cannot be in the past).

**Frequency of Occurrence**

This use case occurs frequently as people regularly celebrate events.

### 4.1.2 Use Case: Announce Payment Deadline

**Use Case ID**

UC-05

**Use Case Name**

Announce Payment Deadline

**Stakeholders and Interests**

**User:** Wants to inform freinds of the payment deadline.

**Primary Actor**

**Event Owner** (User)

**Secondary Actors**

**System**

**Frequency of Occurrence**

This use case occurs when the shop keeper needs to set or change the payment deadline.

**Preconditions**

1. The user is logged into the system.
2. The event is already created.

**Postconditions**

1. The payment deadline is set for the event.
2. friends are notified of the payment deadline.

**Main Flow Success Scenario**

1. The user selects the event for which to announce the payment deadline.
2. The system prompts for the payment deadline details.
3. The user enters the payment deadline date.
4. The user submits the payment deadline.
5. The system saves the payment deadline

**Alternative Scenarios**

* **Scenario 1:** (Occurs after Step 3)  
  **Condition:** If the user does not enter a valid date,  
  **Action:** The system prompts the user to provide a correct date before proceeding.
* **Scenario 2:** (Occurs after Step 4)  
  **Condition:** If the user tries to set a payment deadline that is after the event date,  
  **Action:** The system warns the user that the payment deadline must be set before the event date

**Input Section**

**Payment Deadline Date**

**Output Section**

Confirmation Message

Notification Sent to Friends

**Special Requirements**

1. The payment deadline must be set before the event date.
2. The system must send notifications to all registered freinds.

### Select Items

**Use Case ID**

UC-06

**Use Case Name**

Select Items

**Stakeholders and Interests**

**User**: Wants to choose items from the shop's inventory to add to an event registry for his freinds to view and select.

**Primary Actor**

**User (Event Owner)**

**Secondary Actors**

**None**

**Frequency of Occurrence**

This use case occurs whenever the user is setting or updating an event registry.

**Preconditions**

The user is logged into the system.

The event is already created.

Items are available in the shop's inventory.

**Postconditions**

The selected items are successfully added to the event registry.

Freinds can view and select the chosen items in the registry.

**Main Success Scenario**

The user selects the event for which to choose items.

The system displays the available items from the inventory.

The user selects items for the registry.

The user submits the selected items.

The system saves the selected items and confirms the update to the event registry.

**Alternative Scenarios**

**Scenario 1:(At point 3)**

**Condition:** The user does not select any items.

**Action:** The system asks the user to select at least one item.

**Input Section**

**List\_Id**

**Event\_Id**

**Selected Items List**

**Output Section**

**Confirmation Message** (success or failure)

**Gift\_list**

**Special Requirements**

All selected items must be currently available in the shop's inventory.

## Muhammad Ali Sultan

## 4.2.1 View Event

**Use Case ID:** UC-07

**Use Case :** View Event

**Primary Actor:**

- Friend

**Secondary Actors:**

- System

**Stakeholders:**

- User(the person who has created the event)

- Shop Owner(the person in whose shop or site the event is registered)

**Preconditions:**

- Friend has an access to the good internet Internet connection.

- Event deadline has not reached yet.

- Friend is authorized for the gifts selection by the event creator.

**Main Flow:**

1. Friend goes through authorization process to access the event.

2. System shows event name, date, and description.

3. System displays items list created by user(event organizer) along with the price of item.

4. Friend clicks on each item to check their quantity required .

5. System fades the items that are fully selected and otherwise display the item details.

**Alternate Flow:**

1. Events deadline has reached during viewing.

2. Their can be traffic on event site causing errors.

3. Friend accidentally exit the website.

**Postconditions:**

- Friend can view event details , available or remaining items and their quantity.

**Inputs:**

Event id

User login id

**Outputs:**

List of available shop items (with names, descriptions, prices, quantities)

Stock availability confirmation

Details about the remaining required gift items.

### 4.2.2 Select Gifts

**Use Case ID:** UC-08

**Use Case:** Select Gifts

**Primary Actor:**

Friend (Guest User)

**Secondary Actors:**

System

**Stakeholders:**

User (the person who created the event)

Shop Owner (the person providing the items in the shop)

**Preconditions:**

The friend has a stable internet connection.

The event is active and has not passed its deadline.

The friend has permission from the event creator to select gifts.

**Main Flow:**

The friend accesses the event registry through the system.

The system displays the event name, date, and details.

The system shows the list of available gifts for selection.

The friend chooses the gifts they wish to select.

The friend submits their gift selections.

The system confirms the gift selections and updates the registry accordingly.

**Alternate Flow:**

The event deadline has passed during the selection process.

The friend accidentally navigates away from the website.

**Postconditions:**

The friend can view the selected gifts and their status in the event registry.

**Inputs:**

Event ID

friend login ID

Selection\_Id

List\_Id

Item\_Id

**Outputs:**

Confirmation message of selected gifts

Updated gift selection list reflecting the friend’s choices

Notification of available or unavailable gifts

## Ali Muhammad

### 4.3.2 Modify Selected Items

**Use Case ID:** UC-09

**Use Case:** Modify Selected Items

**Primary Actor:**

Friend

**Secondary Actors:**

System

**Stakeholders:**

**User**

**Shop Keeper**(the person providing the items)

**Preconditions:**

The event is active and has not passed its payment deadline.

The friend has permission from the event creator to modify their selections.

**Main Flow:**

The friend logs into the system and navigates to the event registry.

The system displays the friend’s current selections.

The friend adds or removes items from their selected list.

The friend reviews the updated list of selected items.

The friend confirms the changes.

The system saves the modified selections and updates the event registry.

**Alternate Flow:**

If the friend tries to modify selections after the payment deadline, the system prevents changes and notifies them.

If the friend attempts to add an item that is out of stock, the system alerts them and suggests alternatives.

**Postconditions:**

The modified selections are saved in the event registry.

**Inputs:**

Event\_ID

User login\_ID

Selected items (to add or remove)

**Outputs:**

Confirmation message of successful modification

Updated list of selected items

### 4.3.2 Make Payment

**Use Case ID:** UC-10

**Use Case:** Make Payment

**Primary Actor:**

Friend Secondary Actors:

System

**Stakeholders:**

User (the event owner)

Shop Owner (the person providing the items)

**Preconditions:**

The friend has selected items in the event registry.

The event is active and has not passed its payment deadline.

The friend is authorized to make a payment.

**Main Flow:**

The friend navigates to the payment section after modifying their selected items.

The system displays the total cost of selected items.

The friend enters payment details (credit card information, billing address, etc.).

The friend reviews the payment information.

The friend submits the payment.

The system processes the payment and confirms the transaction.

The system updates the event registry with the finalized selections and sends a confirmation message.

**Alternate Flow:**

If the payment process fails (e.g., due to insufficient funds or incorrect information), the system prompts the friend to re-enter payment details.

If the friend navigates away from the payment page before completing the transaction, the system prompts them to confirm before leaving.

**Postconditions:**

Payment is processed, and a receipt is generated.

The friend receives confirmation of the transaction.

**Inputs:**

Event ID

Login\_ID (if applicable)

Payment\_Id

Payment\_date

Payment\_amount

**Outputs:**

Confirmation message of payment success or failure

Receipt of the transaction for the friend