North South University

Report on

Software Engineering Project : Wedding Bridge

November 25, 2024

Prepared by
2211486042, Syeed Mahmud
2111599642, Adiba Sarker Adar

Course Instructor:
Mrs. Sarnali Basak

GitHub: Github Repository

Contents

List of Figures

1 Project Title

WEDDING BRIDGE

2 Project Description

"WEDDING BRIDGE" is a unique web-based platform designed to connect local communities with foreign visitors through the vibrant celebration of weddings. The platform serves as a bridge, enabling cultural exchange while fostering financial contributions to local events. Unlike conventional event platforms, "WEDDING BRIDGE" is designed to showcase the rich traditions and customs of our country, allowing foreigners to experience the authenticity of local weddings.

The platform simplifies the process for local hosts to share their wedding details, including dates, venues, and cultural highlights, while enabling interested foreigners to RSVP and contribute financially to the events. It features a user-friendly interface where users can easily browse weddings by location, date, or cultural significance. Each wedding profile includes rich descriptions, photographs, and optional engagement packages to enhance the visitor's experience.

To ensure smooth operations, "WEDDING BRIDGE" incorporates a robust backend system powered by a database management system (DBMS). This system efficiently manages wedding information, user profiles, and transactions. The platform also includes an intuitive admin panel that allows administrators to oversee user accounts, wedding listings, and payment processes. Admins can add or update event details, approve user registrations, and maintain platform security with role-based access control.

Through "WEDDING BRIDGE", the project aims to provide a seamless experience for both local hosts and foreign guests, creating an unforgettable cultural exchange that benefits all parties involved.

3 Vision Statement

> "WEDDING BRIDGE" is a web-based platform **For** both Local People and Foreigners **Who** seek to experience local cultures. **Unlike** Join My Wedding, "WEDDING BRIDGE" is **[our]** a custom design platform **that** ensures the smooth management of wedding information, user profiles, and RSVPs.

4 Project Objective

The objectives of this projects are:

1. Design a real life Software Project.
2. Showcasing Multiple Software Design structure.

Requirements

5 Functional Requirements

FR1. User should be able to register, log in, and update their profile details.

FR2. There should be a Search option to find the specific wedding using the bride and groom's name.

FR3. There should be a filter option to find all the weddings on specific dates and a list of the upcoming weddings.

FR4. Wedding host users can create and edit their wedding details like bride-groom info, date, venue, event list, food item list, and bank info to receive payment.

FR5. Users should be able to cancel any registered wedding listed in their profile

FR6. There should be automated notifications and mail systems for hosts and guests (event reminders, updates, etc.).

FR7. Only the admin should be able to log in to the admin control center.

FR8. Admin users can manage accounts, monitor system activity, and manage wedding event data.

FR9. The system has to support online payments, track transactions, and generate receipts.

FR10. Guests should be able to leave reviews about their wedding experience, and hosts can provide feedback on the guests.

FR11. An optimized database management system should be created with efficient storage and retrieval of wedding details, user profiles, transaction histories, and reviews.

6 Non-Functional Requirements

NFR1. There should be an Intuitive and user-friendly interface for all the users.

NFR2. The website should be capable of handling 5 million users at the same time without affecting its performance.

NFR3. Need to ensure high uptime and reliability, especially during peak event times.

NFR4. Ensure data security and privacy (SSL encryption for transactions).

NFR5. Role-based access control to differentiate between normal users and admins.

NFR6. Fast loading times for pages and efficient query handling from the database.

NFR7. Minimized latency in delivering notifications or transaction confirmations.

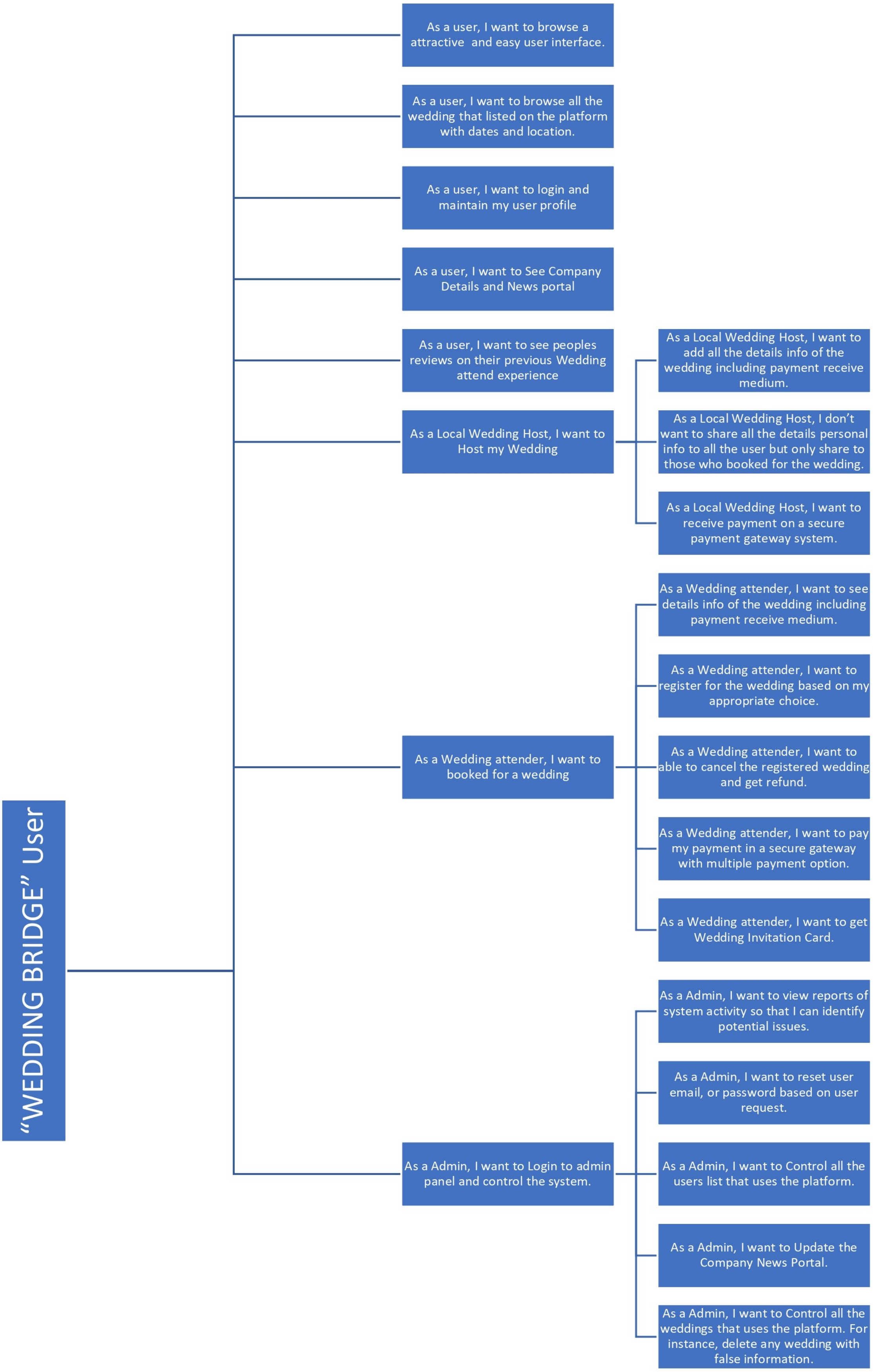
NFR8. Site visitors, any user should be able to browse the homepage, all listed weeding lists, company details, and the company news portal without login.

NFR9. Users should not be able to view specific wedding details and register for weddings without logging in.

NFR10. The user won't be able to give a review before visiting the wedding.

NFR11. The wedding will listed in the user's main profile after confirming payment and successful registration for the wedding.

7 User Story



“WEDDING BRIDGE” User

As a user, I want to browse a attractive and easy user interface.

As a user, I want to browse all the wedding that listed on the platform with dates and location.

As a user, I want to login and maintain my user profile

As a user, I want to See Company Details and News portal

As a user, I want to see peoples reviews on their previous Wedding attend experience

As a Local Wedding Host, I want to Host my Wedding

As a Local Wedding Host, I want to add all the details info of the wedding including payment receive medium.

As a Local Wedding Host, I don’t want to share all the details personal info to all the user but only share to those who booked for the wedding.

As a Local Wedding Host, I want to receive payment on a secure payment gateway system.

As a Wedding attender, I want to booked for a wedding

As a Wedding attender, I want to see details info of the wedding including payment receive medium.

As a Wedding attender, I want to register for the wedding based on my appropriate choice.

As a Wedding attender, I want to able to cancel the registered wedding and get refund.

As a Wedding attender, I want to pay my payment in a secure gateway with multiple payment option.

As a Wedding attender, I want to get Wedding Invitation Card.

As a Admin, I want to Login to admin panel and control the system.

As a Admin, I want to view reports of system activity so that I can identify potential issues.

As a Admin, I want to reset user email, or password based on user request.

As a Admin, I want to Control all the users list that uses the platform.

As a Admin, I want to Update the Company News Portal.

As a Admin, I want to Control all the weddings that uses the platform. For instance, delete any wedding with false information.

Figure 1: User Story

8 Project Design

8.1 Use Case Diagram

9

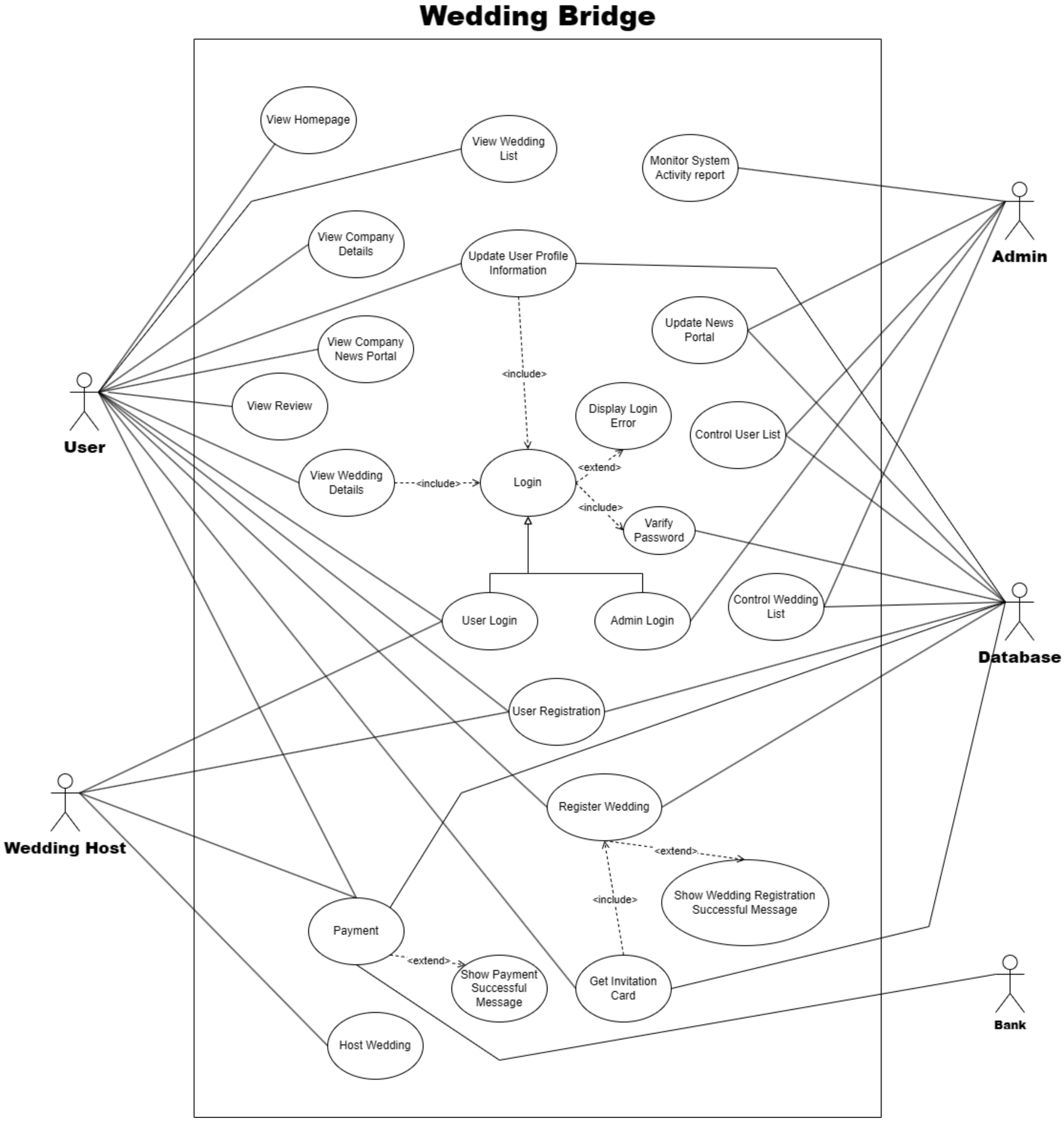
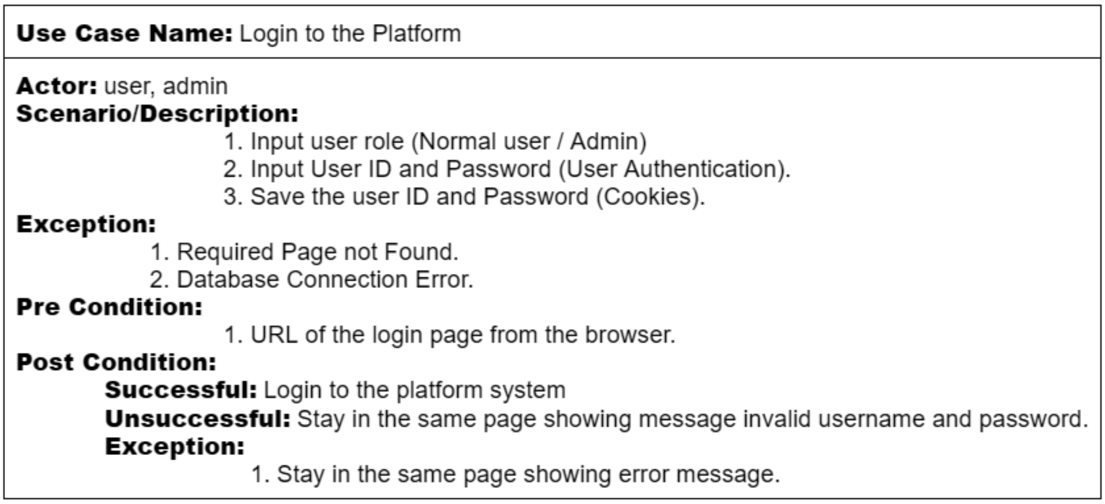


Figure 2: Use Case Diagram

8.2 Use Case Specification

9



Use Case Name: Login to the Platform

Actor: user, admin
Scenario/Description:

1. Input user role (Normal user / Admin)
2. Input User ID and Password (User Authentication).
3. Save the user ID and Password (Cookies).

Exception:

1. Required Page not Found.
2. Database Connection Error.

Pre Condition:

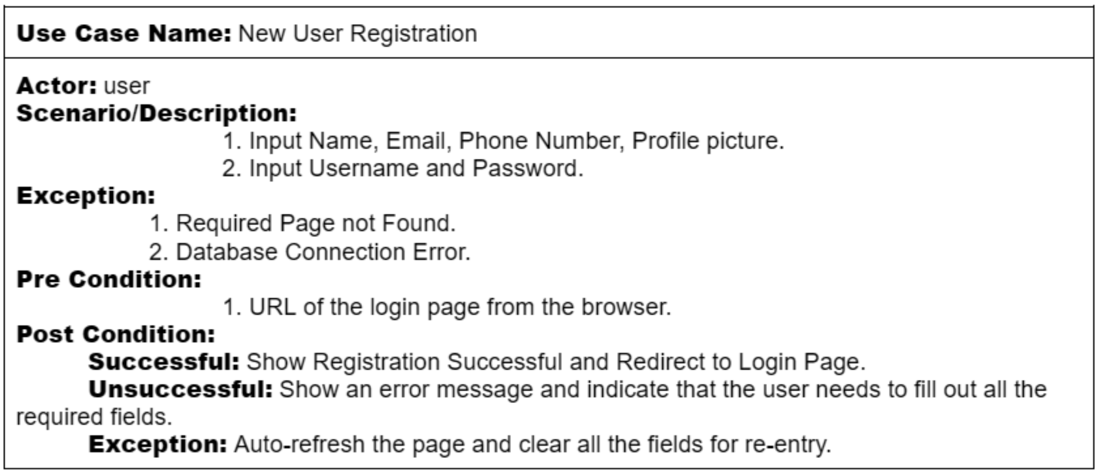
1. URL of the login page from the browser.

Post Condition:

Successful: Login to the platform system
Unsuccessful: Stay in the same page showing message invalid username and password.
Exception:

1. Stay in the same page showing error message.

Figure 3: Use Case Specification 1



Use Case Name: New User Registration

Actor: user
Scenario/Description:

1. Input Name, Email, Phone Number, Profile picture.
2. Input Username and Password.

Exception:

1. Required Page not Found.
2. Database Connection Error.

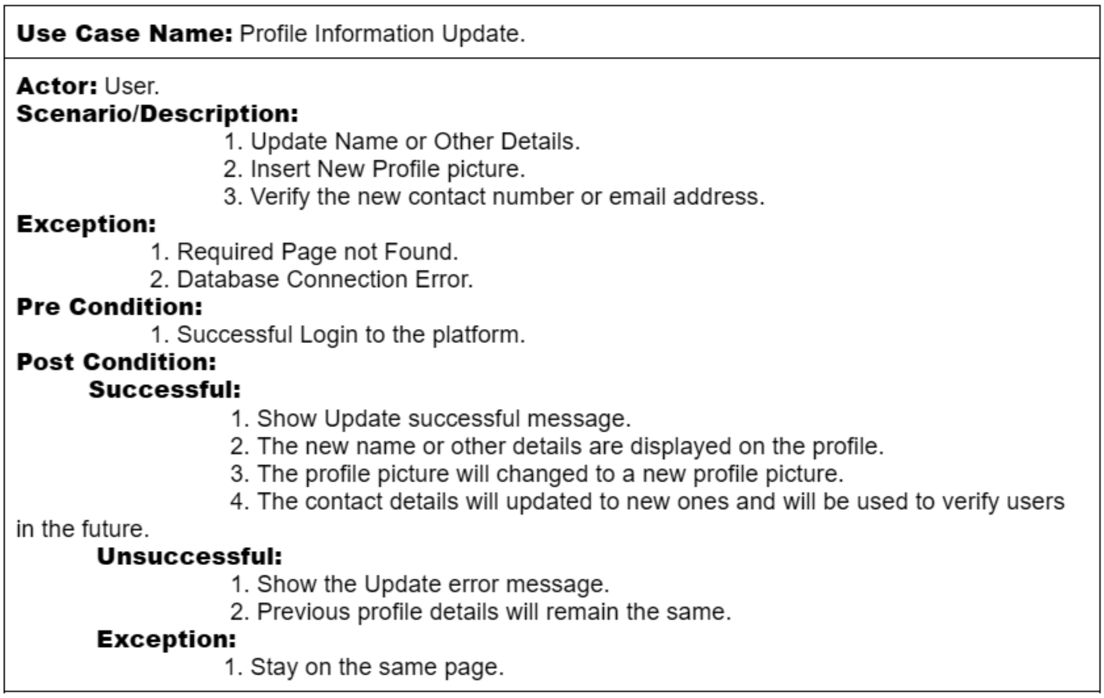
Pre Condition:

1. URL of the login page from the browser.

Post Condition:

Successful: Show Registration Successful and Redirect to Login Page.
Unsuccessful: Show an error message and indicate that the user needs to fill out all the required fields.
Exception: Auto-refresh the page and clear all the fields for re-entry.

Figure 4: Use Case Specification 2



Use Case Name: Profile Information Update.

Actor: User.
Scenario/Description:

1. Update Name or Other Details.
2. Insert New Profile picture.
3. Verify the new contact number or email address.

Exception:

1. Required Page not Found.
2. Database Connection Error.

Pre Condition:

1. Successful Login to the platform.

Post Condition:

Successful:

1. Show Update successful message.
2. The new name or other details are displayed on the profile.
3. The profile picture will changed to a new profile picture.
4. The contact details will updated to new ones and will be used to verify users in the future.

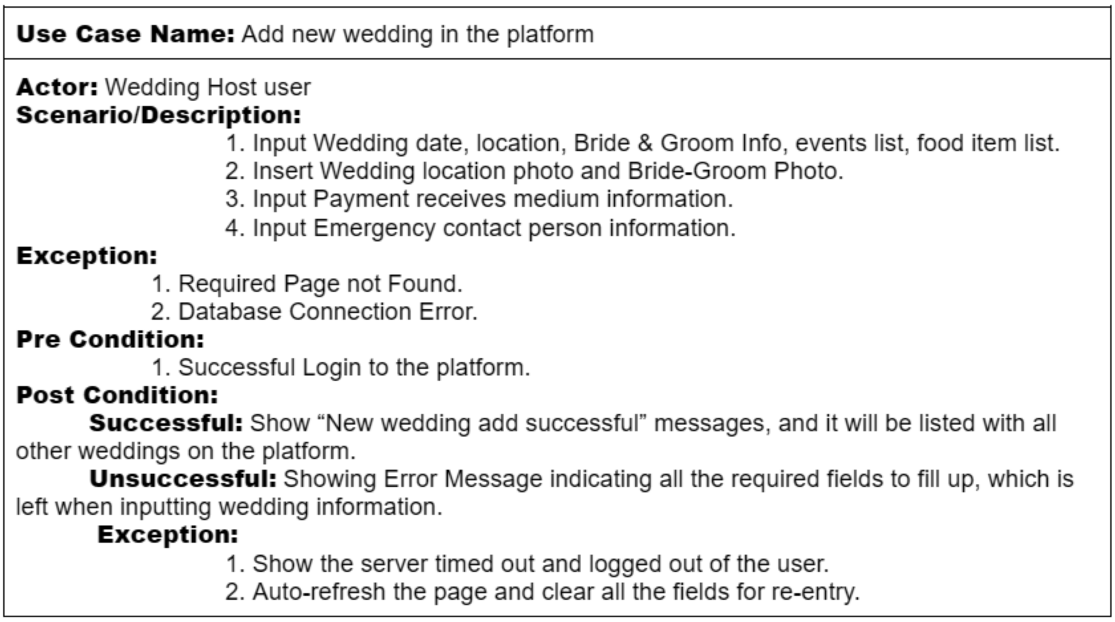
Unsuccessful:

1. Show the Update error message.
2. Previous profile details will remain the same.

Exception:

1. Stay on the same page.

Figure 5: Use Case Specification 3



Use Case Name: Add new wedding in the platform

Actor: Wedding Host user
Scenario/Description:

1. Input Wedding date, location, Bride & Groom Info, events list, food item list.
2. Insert Wedding location photo and Bride-Groom Photo.
3. Input Payment receives medium information.
4. Input Emergency contact person information.

Exception:

1. Required Page not Found.
2. Database Connection Error.

Pre Condition:

1. Successful Login to the platform.

Post Condition:

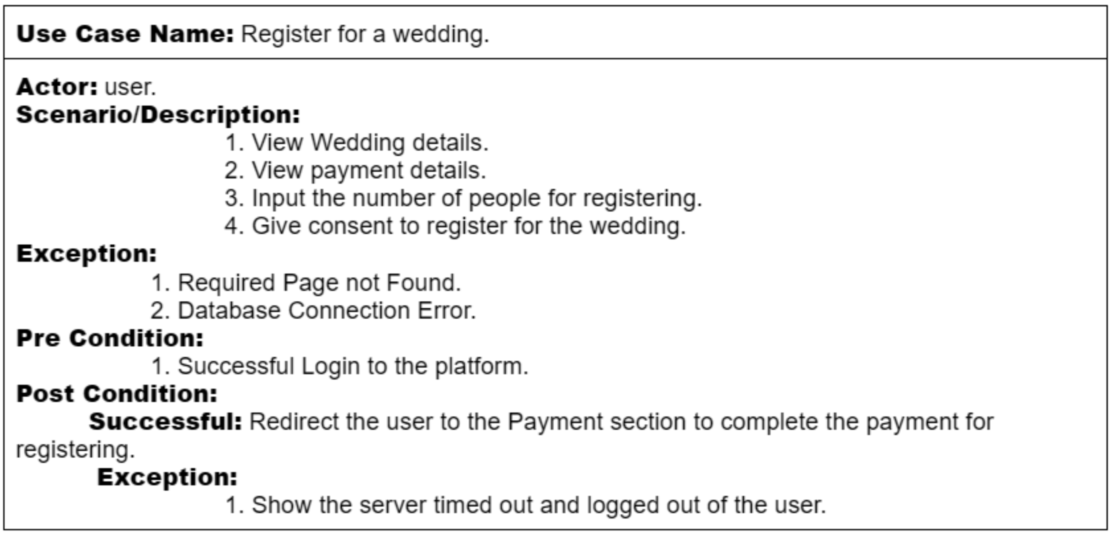
Successful: Show “New wedding add successful” messages, and it will be listed with all other weddings on the platform.

Unsuccessful: Showing Error Message indicating all the required fields to fill up, which is left when inputting wedding information.

Exception:

1. Show the server timed out and logged out of the user.
2. Auto-refresh the page and clear all the fields for re-entry.

Figure 6: Use Case Specification 4



Use Case Name: Register for a wedding.

Actor: user.

Scenario/Description:

1. View Wedding details.
2. View payment details.
3. Input the number of people for registering.
4. Give consent to register for the wedding.

Exception:

1. Required Page not Found.
2. Database Connection Error.

Pre Condition:

1. Successful Login to the platform.

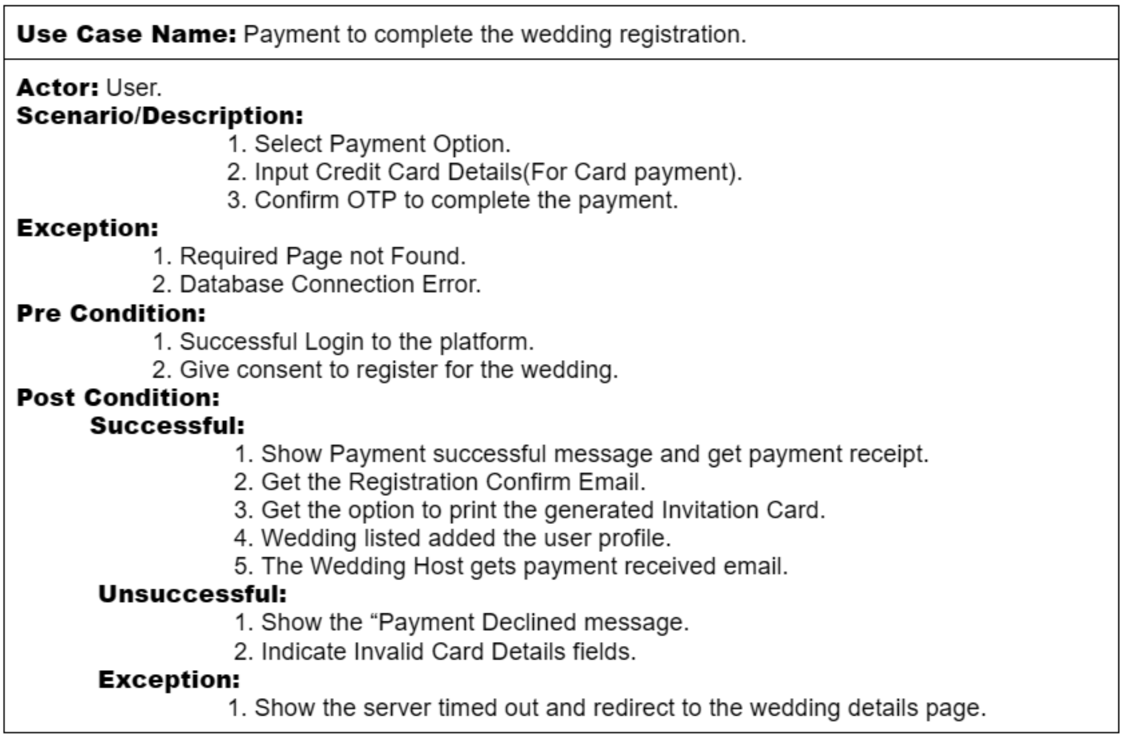
Post Condition:

Successful: Redirect the user to the Payment section to complete the payment for registering.

Exception:

1. Show the server timed out and logged out of the user.

Figure 7: Use Case Specification 5



Use Case Name: Payment to complete the wedding registration.

Actor: User.

Scenario/Description:

1. Select Payment Option.
2. Input Credit Card Details(For Card payment).
3. Confirm OTP to complete the payment.

Exception:

1. Required Page not Found.
2. Database Connection Error.

Pre Condition:

1. Successful Login to the platform.
2. Give consent to register for the wedding.

Post Condition:

Successful:

1. Show Payment successful message and get payment receipt.
2. Get the Registration Confirm Email.
3. Get the option to print the generated Invitation Card.
4. Wedding listed added the user profile.
5. The Wedding Host gets payment received email.

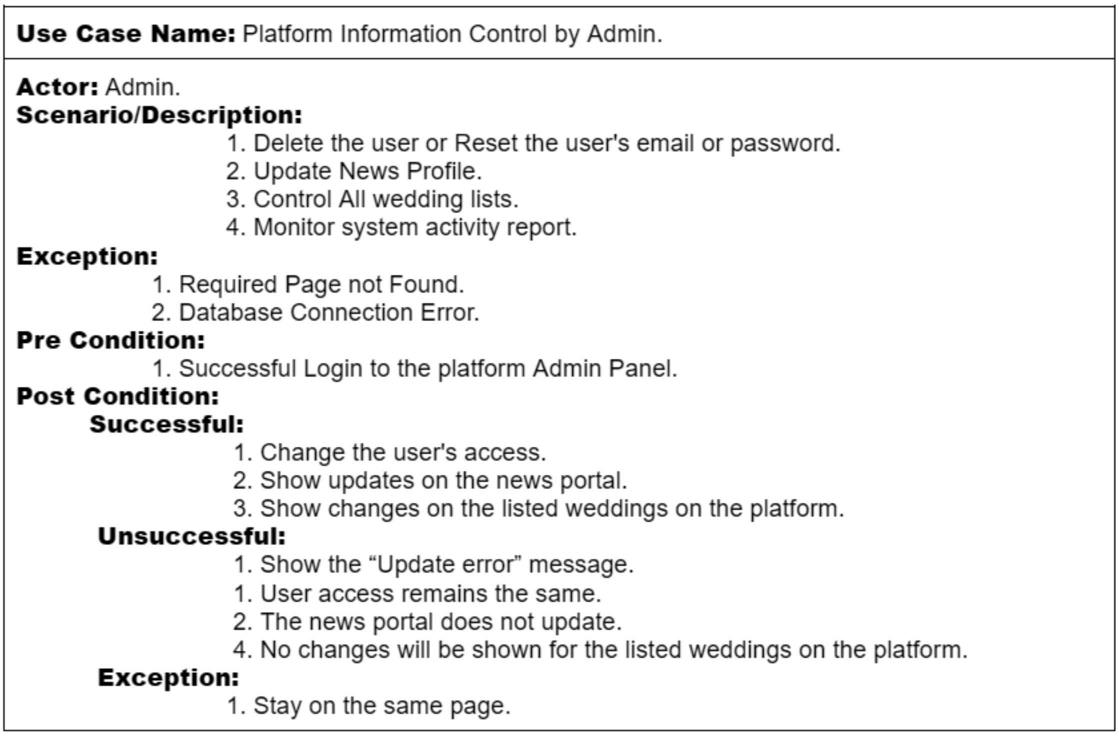
Unsuccessful:

1. Show the “Payment Declined message.
2. Indicate Invalid Card Details fields.

Exception:

1. Show the server timed out and redirect to the wedding details page.

Figure 8: Use Case Specification 6



Use Case Name: Platform Information Control by Admin.

Actor: Admin.

Scenario/Description:

1. Delete the user or Reset the user's email or password.
2. Update News Profile.
3. Control All wedding lists.
4. Monitor system activity report.

Exception:

1. Required Page not Found.
2. Database Connection Error.

Pre Condition:

1. Successful Login to the platform Admin Panel.

Post Condition:

Successful:

1. Change the user's access.
2. Show updates on the news portal.
3. Show changes on the listed weddings on the platform.

Unsuccessful:

1. Show the “Update error” message.
1. User access remains the same.
2. The news portal does not update.
4. No changes will be shown for the listed weddings on the platform.

Exception:

1. Stay on the same page.

Figure 9: Use Case Specification 7

8.3 UML Class Diagram

10

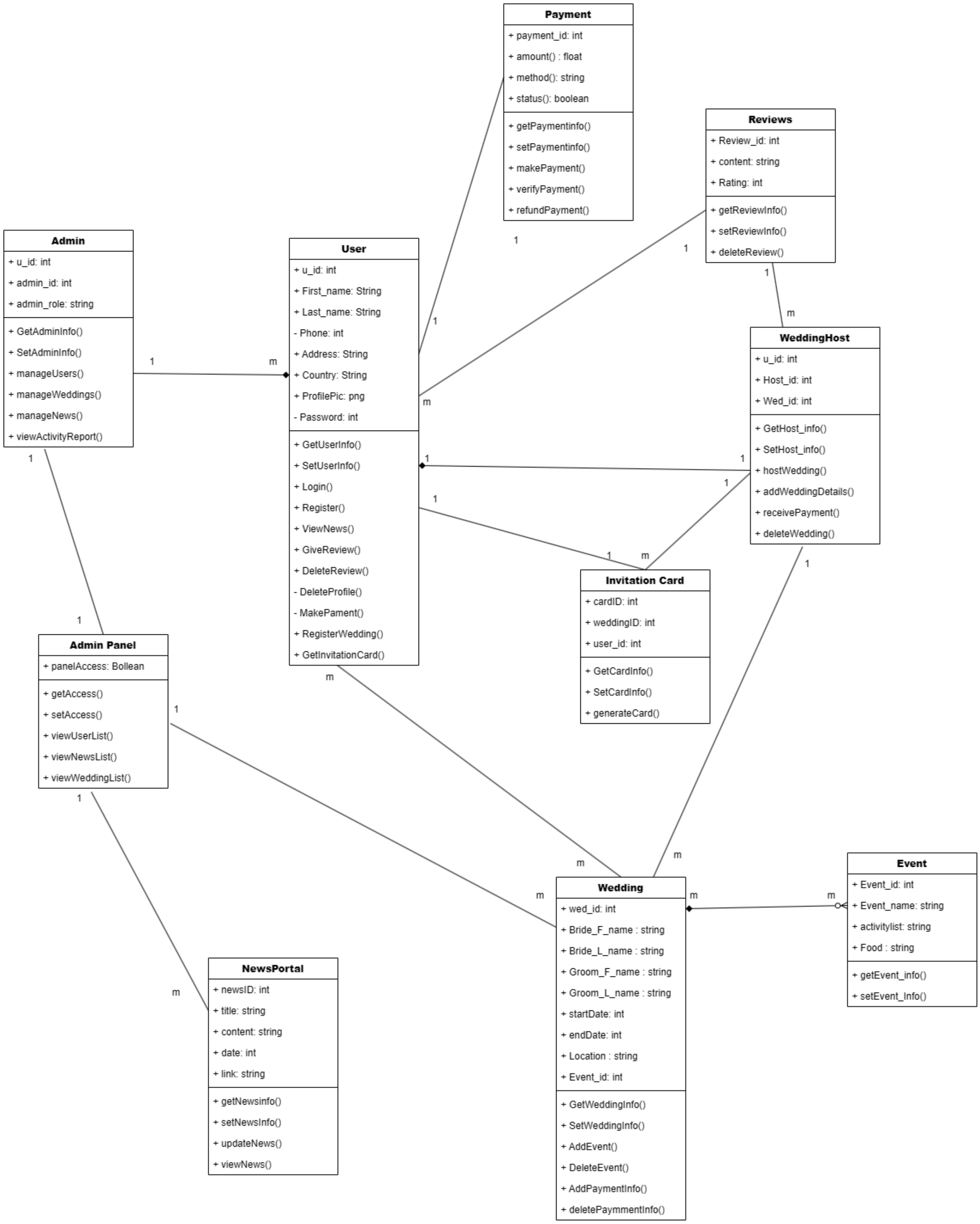


Figure 10: UML Class Diagram

8.4 Database Schema Diagram

12

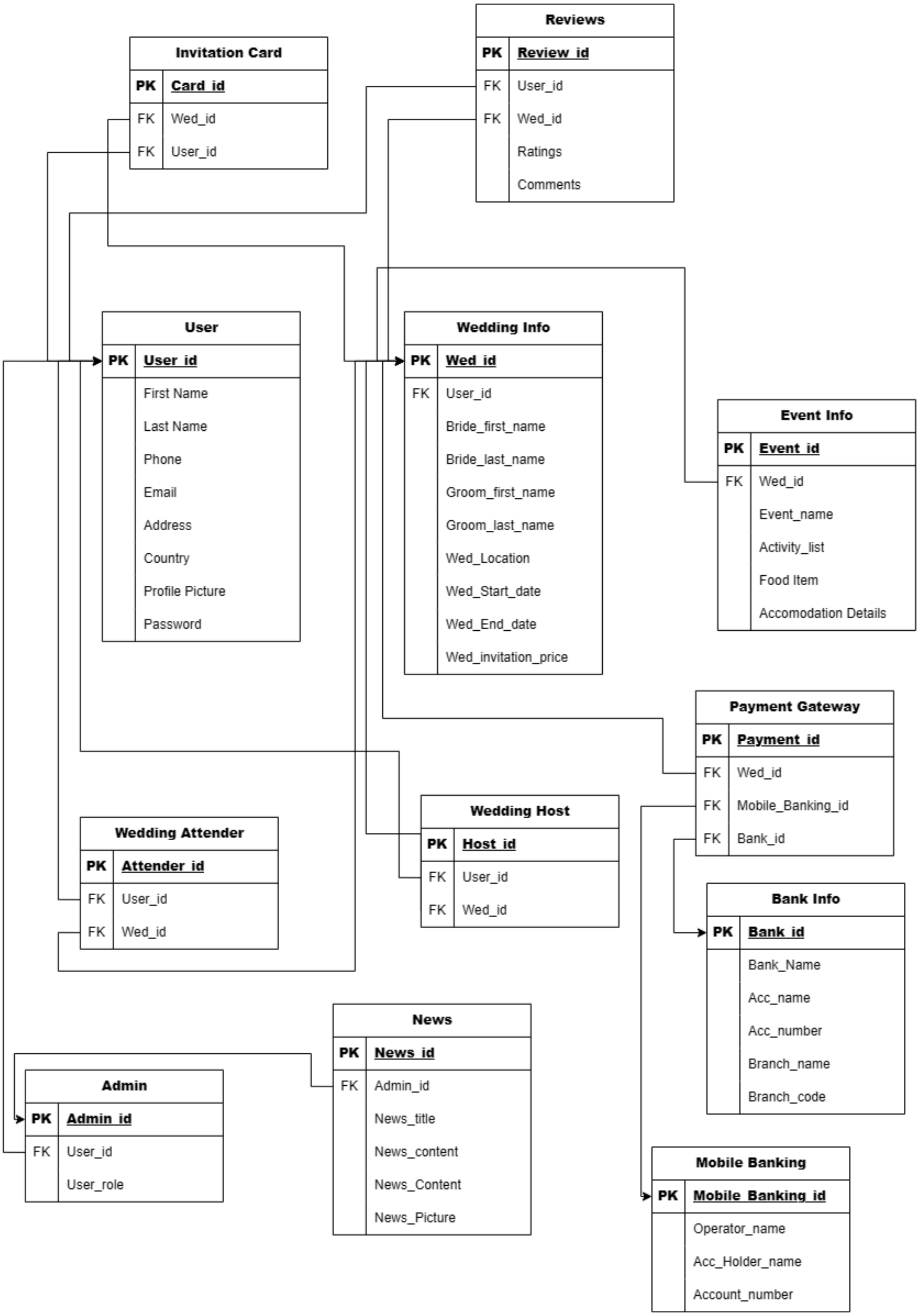


Figure 11: Database Schema Diagram

8.5 Sequence Diagram

12

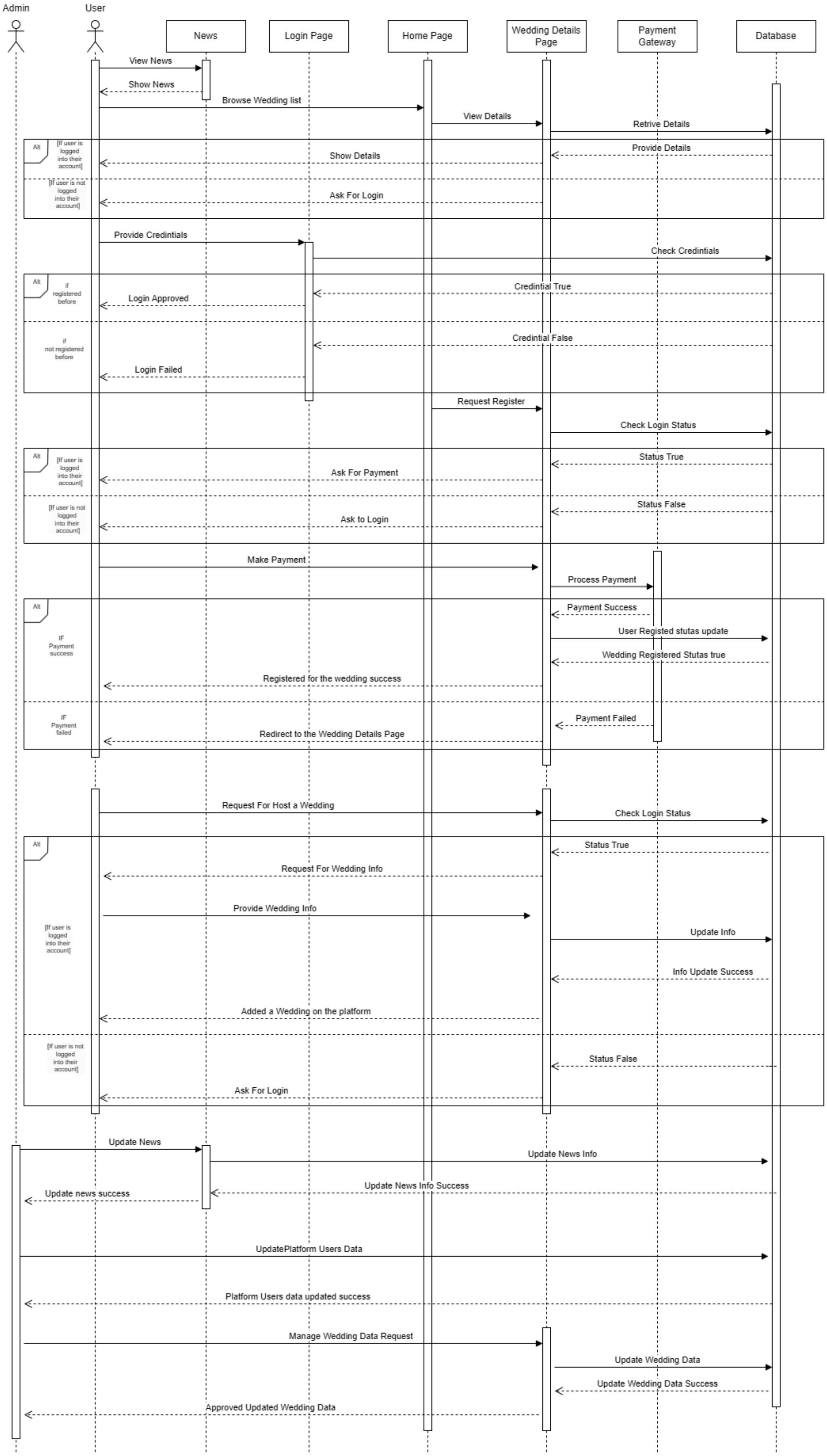


Figure 12: Sequence Diagram

9 Technologies

9.1 Frontend Development

- HTML: For structuring the web pages.
- CSS: For styling and enhancing the visual appeal of the website.
- Tailwind: Tailwind CSS enables more customization without writing custom CSS, whereas Bootstrap has a set of design defaults. Tailwind CSS can lead to smaller CSS file sizes since we only include the classes we use, potentially improving performance.
- DaisyUI is a plugin for Tailwind CSS that adds component classes to help you design your UI straight from your HTML.

9.2 Backend Development

- PHP: A server-side scripting language commonly used for web development. PHP will be employed for specific functionalities that complement the Node.js backend.

9.3 Database

- MySQL: A relational database management system. MySQL will efficiently store user data, product information, and plant records.