



MILESTONE 4

GROUP 1
EzPeazy
Solutions

TEAM DEVELOPMENT
DOCUMENTATION AND
DESIGN SPECIFICATIONS

Prepared by:

Abdullah Ali - 2309318

Mikaeel Suffla - 2453876

Mohammed Bangie - 2610990

Raeesa Lorgat - 2686242

Ismaaeel Shaikh Ebrahim - 2435817

Mohammed Essof - 2706657

Table of Contents

EZVenues	1
INTRODUCTION TO EZVENUES.....	1
Summarized Requirements:	1
SYSTEM REQUIREMENTS.....	1
USER REQUIREMENTS	2
USE CASE DIAGRAM.....	4
USE CASE DESCRIPTION.....	5
Use Case : Student creates a booking	5
ENTITY RELATIONSHIP DIAGRAM	6
DOMAIN MODEL CLASS DIAGRAM.....	7
USE CASE REALIZATION (SEQUENCE DIAGRAM).....	8
Student creates booking:	8
Lecturer updates booking:	9
Admin updates lecturer profile:.....	1
REPORT STRUCTURE	1
1. Introduction:	1
2. Report Design:	1
3. Data Elements:.....	1
4. Reason for choosing DataGridView:	3
5. Conclusion:	3
TEST CASES:.....	4
Student Creates a Booking:	4
Lecturer Updates Booking:	5
Update Lecturer Profile:	6

Group Number: 1

Group Name: EZPeazySolutions

EZVENUES

INTRODUCTION TO EZVENUES

EzVenues is a venue booking system designed specifically for university environments. The application aims to streamline the process of scheduling venues for lectures, events, and other university activities, enhancing efficiency and user satisfaction. By automating and simplifying the booking process, EzVenues reduces administrative burdens and mitigates common issues such as double bookings and repetitive scheduling tasks.

Summarized Requirements:

- User Management: Users, such as lecturers and students, can easily register, log in, and manage their accounts. Administrators have the ability to assign roles, permissions, and manage user access.
- Venue Booking: The system enables users to search for available venues based on specific criteria like capacity and location, with real-time updates on availability. Automated conflict detection prevents double bookings.
- Recurring Bookings: Users can set up recurring bookings for regularly scheduled events, reducing the need for repetitive scheduling.
- Report Generation: Administrators can generate comprehensive reports on venue utilization, conflicts, and booking statistics, assisting in resource optimization.
- Security and Performance: EzVenues employs authentication mechanisms to ensure authorized access and data security. The system is designed for optimal performance, supporting multiple users simultaneously without noticeable delays.

SYSTEM REQUIREMENTS

- Hardware:
 - A server to run database management system
 - User Devices: Computers with internet access for users to access the booking system.
- Software:
 - Operating System: It is required to be able to run the program and should be compatible with common operating systems such as Windows, macOS, or Linux.
 - Database Management System (DBMS): A local database system used to store, retrieve, and run queries on data.

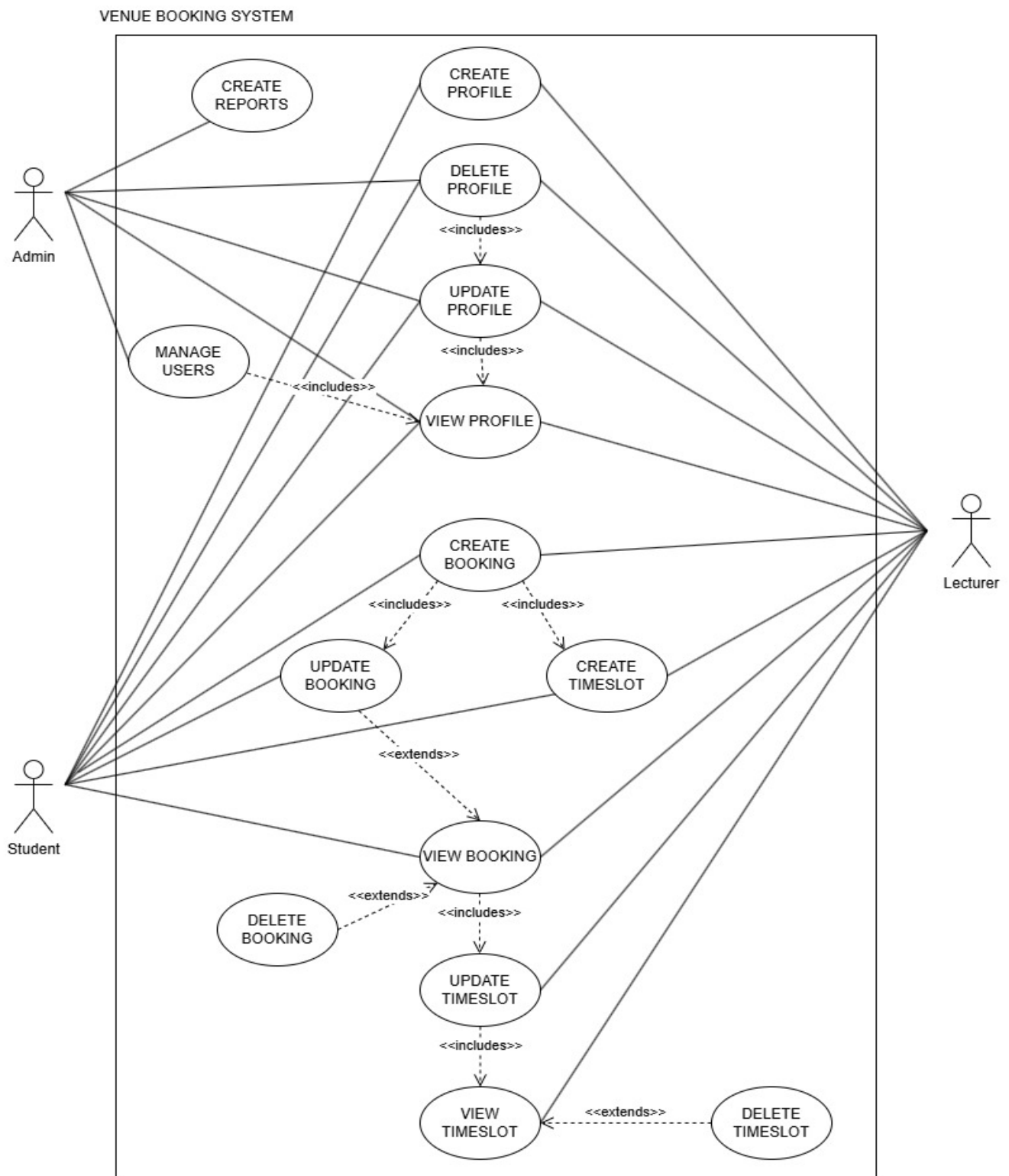
- Booking Algorithm: Solutions Software Algorithm built to allow users to book venues without double booking in an easy and automated way.

USER REQUIREMENTS

- Functional Requirements:
 - Users must be able to register, log in, and manage their accounts in a reliable and secure manner.
 - Administrators should be able to assign roles and permissions to any user on the platform.
 - Users should view comprehensive venue details and be able to filter these criteria to aid in booking for a required event. (Details include: Capacity, Location etc.)
 - System should be able to book a venue while automatically preventing the possibility of double bookings.
 - Administrators should have access to generate reports on venue bookings, including utilization statistics and conflicts.
 - Allow users to modify or cancel their booking within a reasonable time frame before the scheduled event.
 - Administrators should have the ability to block/unblock users.
- Non-Functional Requirements:
 - Performance:
Booking confirmations should be completed in a timely manner. The system must be able to handle multiple users at any given time without any noticeable performance issues.
 - Safety:
User data must be securely stored and protected from unauthorized access.
 - Security:
The system should employ authentication mechanisms to ensure only authorized users can access and modify booking information.
 - Software Quality Attributes:
 - Availability:
The system should always be readily available with minimal downtime to perform maintenance or upgrades.
 - Correctness:
The system should provide up to date information about venue availability and booking. Data entered into the system should be accurately processed and stored.
 - Maintainability:
The system should be modular, readable and well-documented allowing for easy maintenance and platform updates.
 - Usability:
The user interface should be intuitive and user-friendly, requiring minimal training for users to navigate and utilize effectively.

With these features, EzVenues enhances coordination between different academic units and faculties, ensuring that venue bookings are efficiently handled, reducing errors, and improving resource management across the university.

USE CASE DIAGRAM

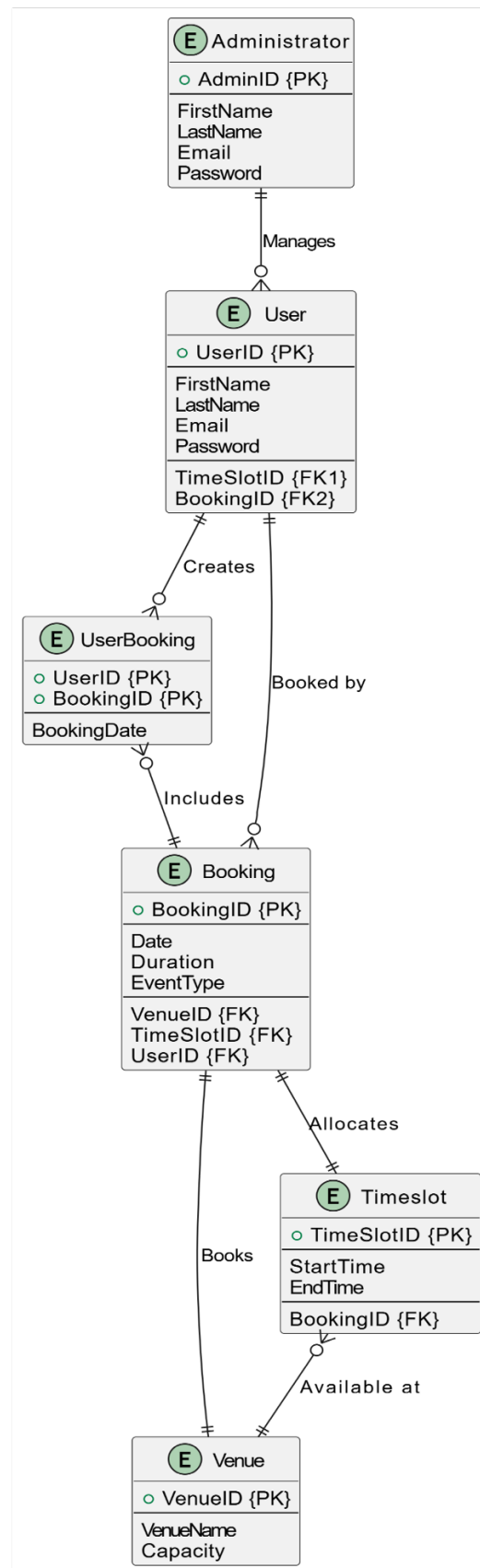


USE CASE DESCRIPTION

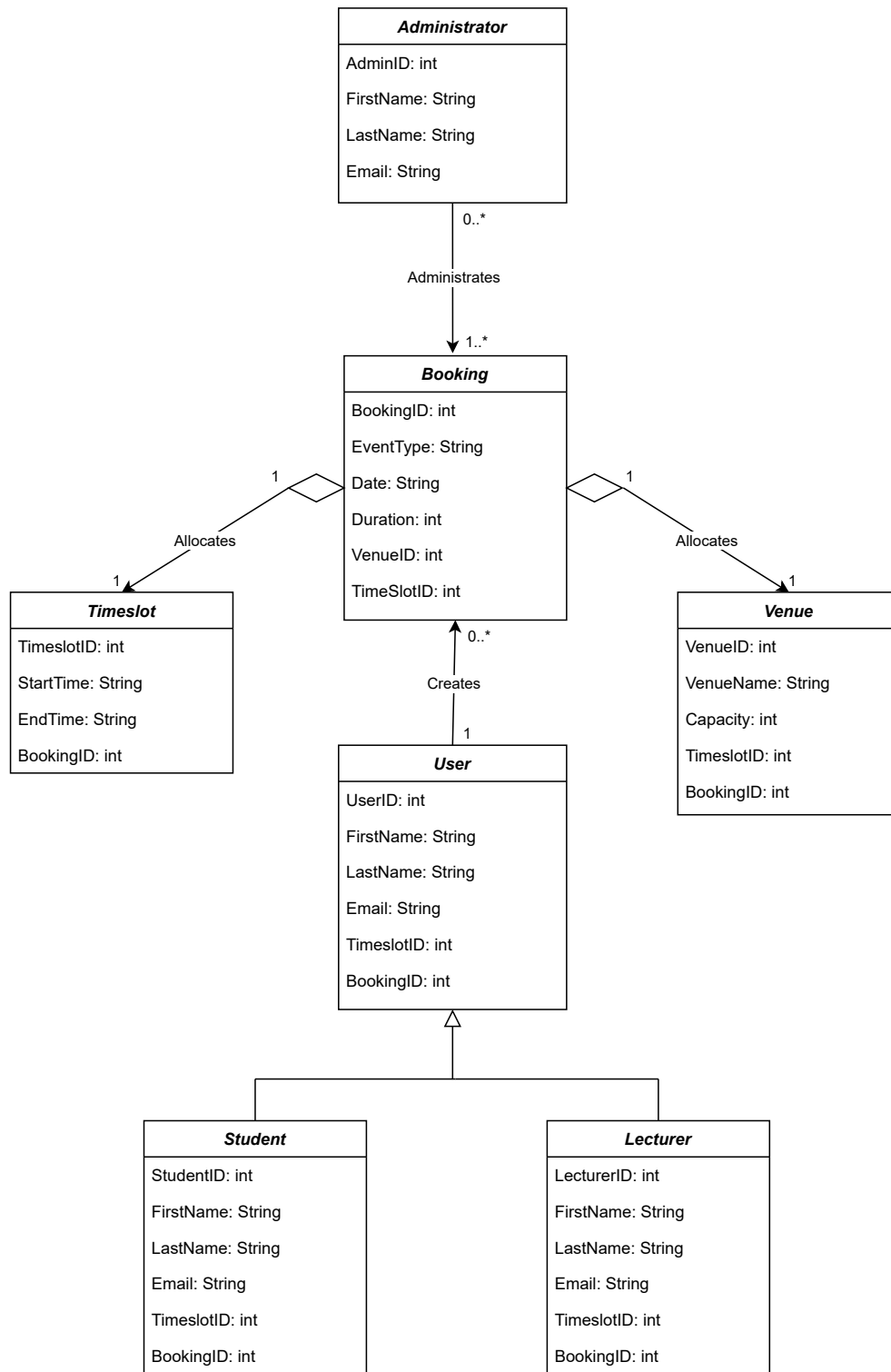
Use Case : Student creates a booking

Name of Use Case	Student creates a booking	
Use case description	This use case explains how a student creates a booking for a venue.	
Actors	Student, System	
Preconditions	<ul style="list-style-type: none"> The student must have an account and be logged in. The system must be accessible and functional. 	
Trigger	Student selects the "Book venue" option.	
Main flow	Student	System
	1. Student selects their required capacity from options provided.	1. System displays venues with the selected capacity.
	2. Students selects a suitable venue from options provided.	2. System displays available dates and times for the selected venue.
	3. Student selects a suitable date and time.	3. System validates the availability with the database and confirms the booking details. System sends notification to the student. System updates the venue's availability.
	Use case ends	
Alternate flow	Student	System
	a) Selected date/time slot is unavailable.	a) System prompts the user to select an alternative date/time.
	b) Student selects a new time slot from the suggestions.	b) System continues from (system) step 3. Use case ends
Post conditions	The student successfully schedules a booking for a venue. The student and all relevant parties receive confirmation notifications. The booking details are stored in the system's database.	

ENTITY RELATIONSHIP DIAGRAM

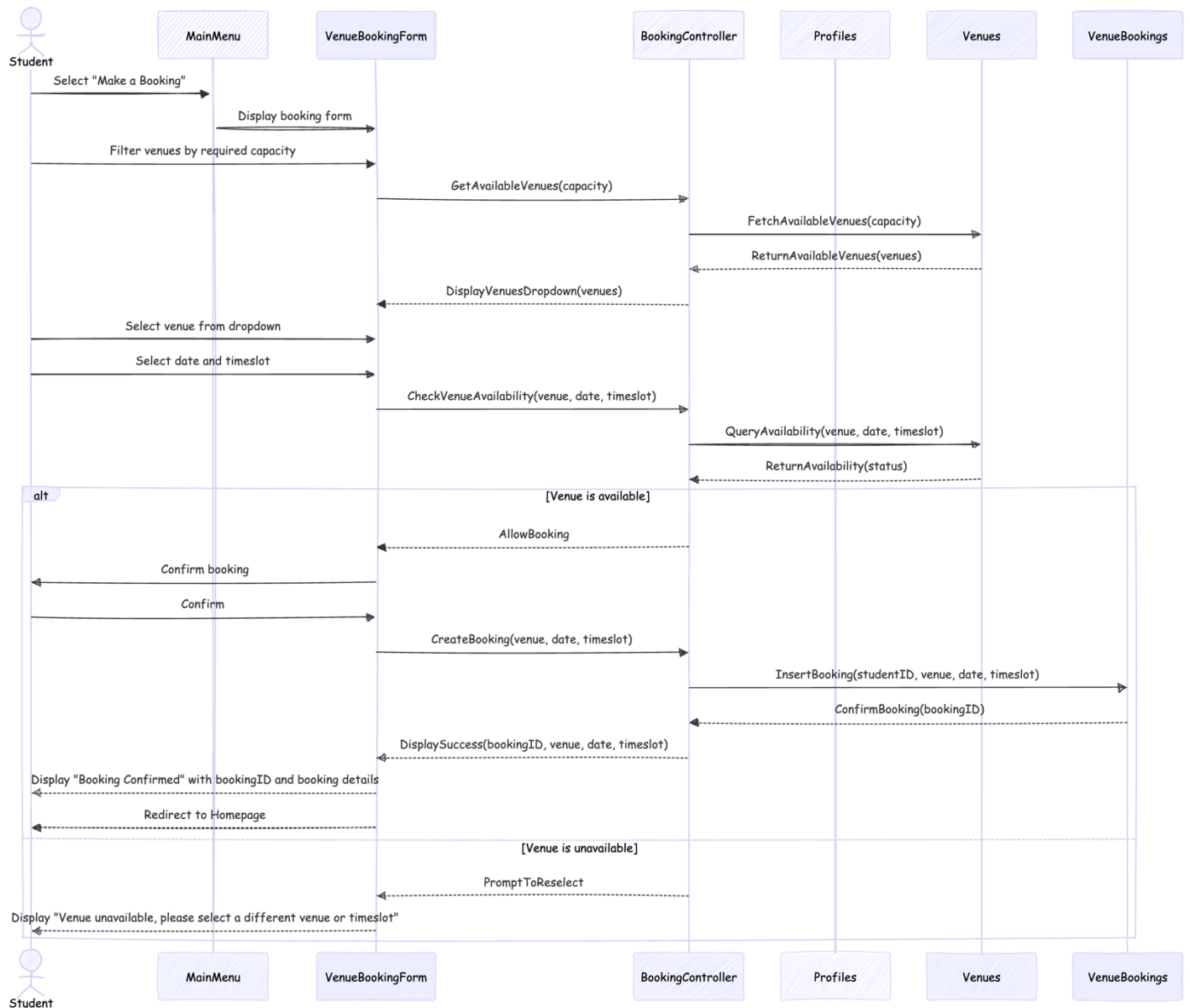


DOMAIN MODEL CLASS DIAGRAM

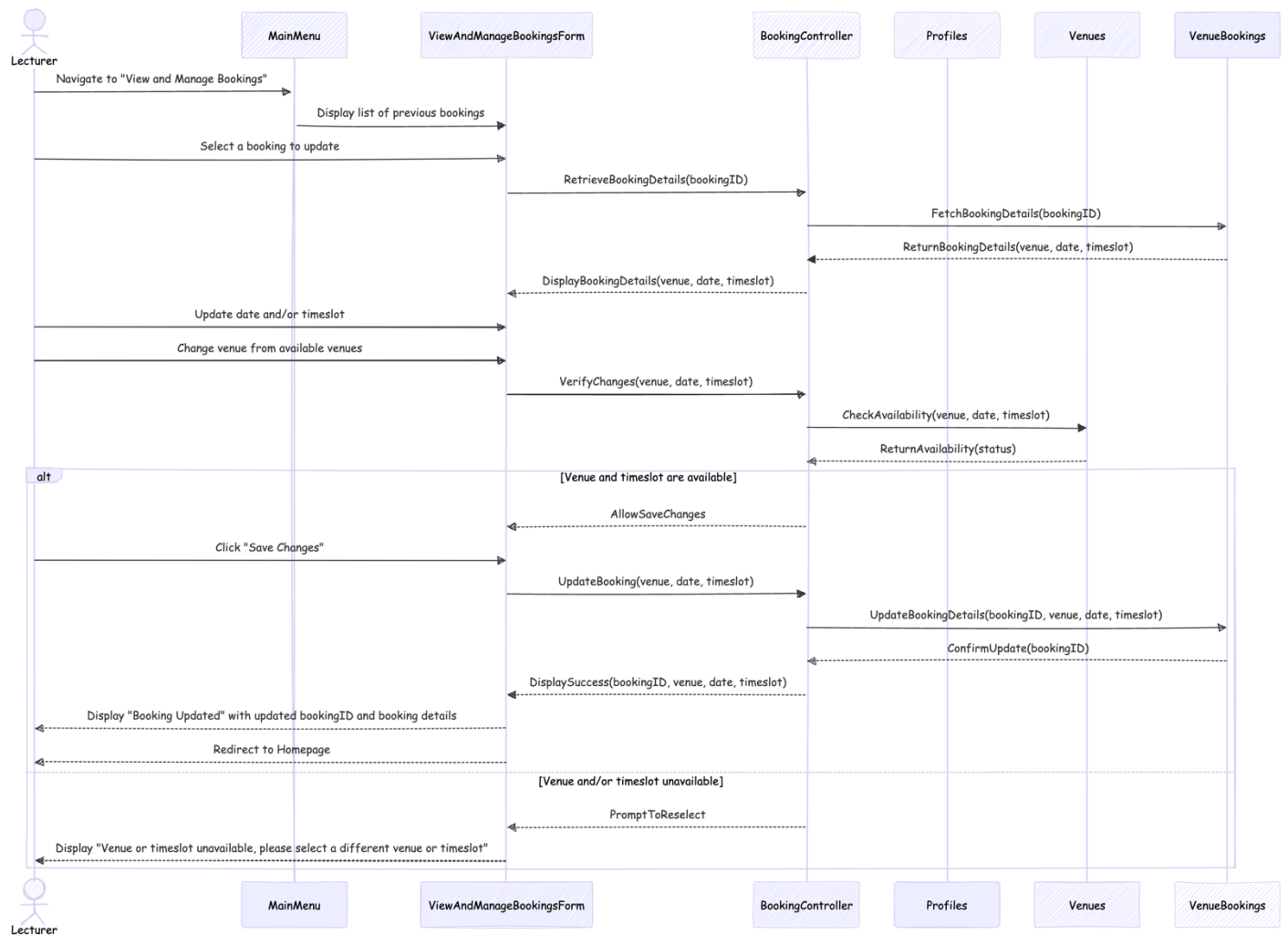


USE CASE REALIZATION (SEQUENCE DIAGRAM)

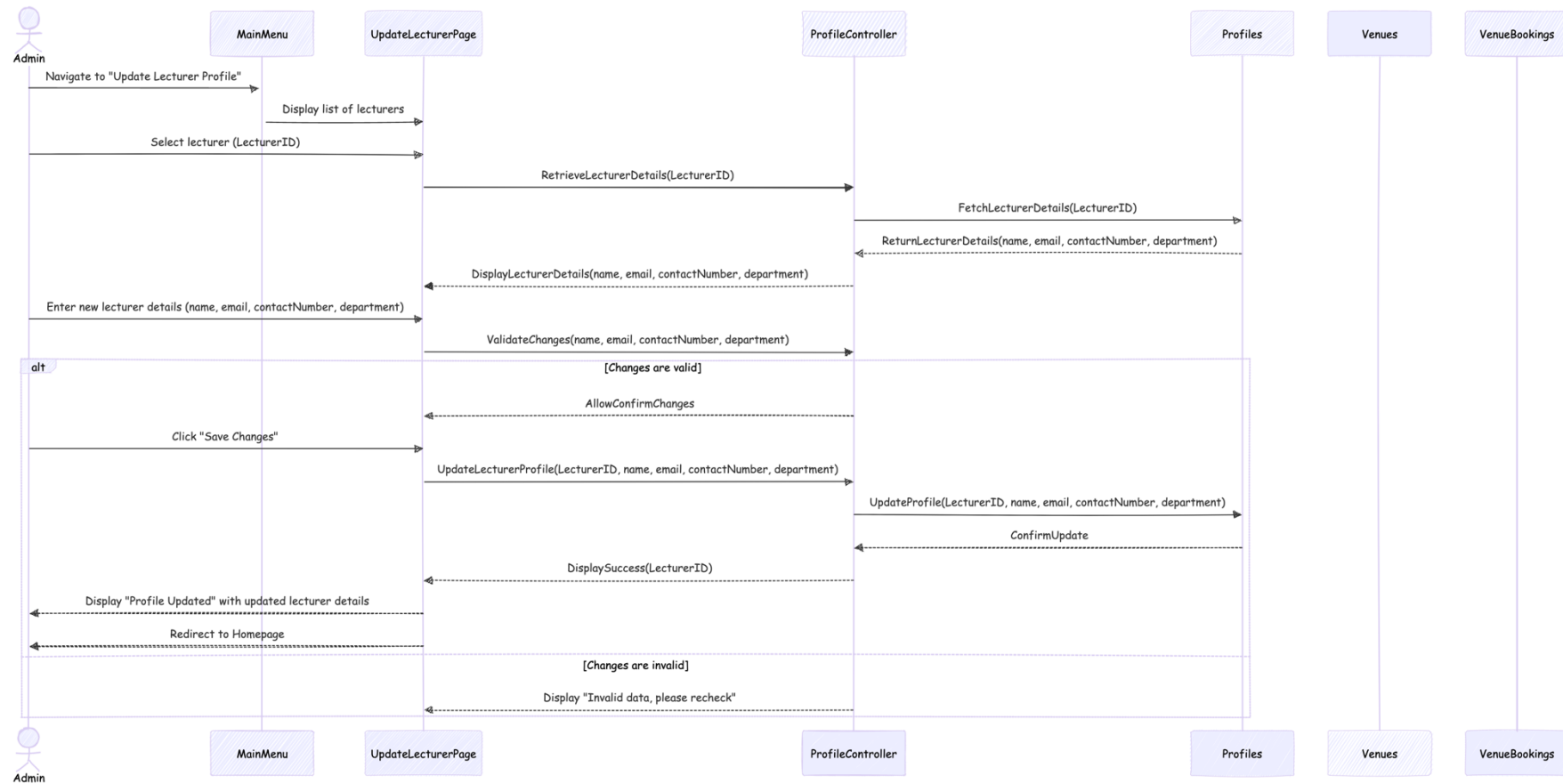
Student creates booking:



Lecturer updates booking:



Admin updates lecturer profile:



REPORT STRUCTURE

1. Introduction:

The FNB 33 Venue Booking Report is a vital report within the venue booking system, specifically designed to offer a detailed overview of the usage and scheduling for the FNB 33 venue. This report provides valuable insights into how frequently the venue is being booked, the type of users making reservations, and any conflicts or issues arising from double bookings. The information in this report assists administrators in understanding venue demand, managing availability, and ensuring smooth operations at FNB 33.

2. Report Design:

The design of the FNB 33 Venue Booking Report was carefully mapped out using a wireframe approach. The report layout was structured to include key data points such as booking times, user details, and conflict detection, allowing administrators to assess the venue's utilization quickly. The report is generated using the DataGridView component, chosen for its ability to present data in a clean and interactive format. This allows users to filter, sort, and search through bookings, making it an ideal tool for tracking the venue's usage.

3. Data Elements:

- Venue Details: Information specific to FNB 33, including location, capacity, and facilities, which helps contextualize the venue's usage.
- Booking Details: A breakdown of all bookings for FNB 33, including the name and type of user (student, lecturer, admin), event type, and duration. This provides a clear view of how the venue is being utilized.
- Booking Trends: Aggregated data on the number of bookings over time (e.g., daily, weekly, or monthly), allowing administrators to identify peak times of usage for FNB 33.
- Booking Conflicts: Any double-booking attempts or conflicts are highlighted, ensuring that no two events overlap at the same time in FNB 33.
- User Type Breakdown: Data on the types of users booking the venue (students, lecturers, or admins) provides insight into how the venue serves different groups within the institution.
- Total Booking Hours: This element provides a sum of all hours booked at FNB 33, offering a quantitative measure of how much the venue is being used over a specific period.



- Booking Trends:**
 This chart displays an overview of the venue's booking patterns based on the number of bookings made for specific time slots and days of the week. It highlights peak usage times, giving insight into when FNB 33 is most in demand. Administrators can use this data to better manage the venue's availability and optimize scheduling.
- Capacity Utilization:**
 This chart provides a clear comparison between the venue's capacity and the actual number of attendees over a given period. It helps assess whether FNB 33 is being fully utilized or underused, allowing administrators to make informed decisions on whether the venue is meeting its potential or if adjustments are needed.
- User Comments:**
 This section, located at the bottom of the report, offers feedback and ratings provided by students, lecturers, and administrators based on their experiences with FNB 33. These comments are invaluable for future users who may want to book the venue and for administrators to track satisfaction and identify areas for improvement.

4. Reason for choosing DataGridView:

- User-Friendly Interface: DataGridView's intuitive, spreadsheet-like interface simplifies navigation, ensuring ease of comprehension for end-users ranging from administrators to faculty members.
- Dynamic Data Binding: The tool facilitates seamless integration with underlying databases, enabling real-time updates and ensuring the accuracy and relevancy of data presented.
- Customization Capabilities: DataGridView's inherent flexibility allows for effortless customization, aligning the report layout precisely with the initially envisioned wireframe design.
- Integration Efficiency: Its seamless integration with diverse data sources ensures swift data retrieval, processing, and display, enhancing the overall reporting efficiency.

5. Conclusion:

In conclusion, the FNB 33 (Venue) Booking Report, meticulously structured with the aid of DataGridView offers a comprehensive overview of venue usage and performance. By providing real-time data on booking trends, peak usage times, and capacity utilization, this report serves as a powerful tool for informed decision-making regarding venue management and scheduling. The addition of user feedback through comments enhances the report by offering qualitative insights into user satisfaction.

Ultimately, the report plays a critical role in optimizing the use of FNB 33, promoting efficient venue management, and ensuring that resources are allocated effectively to meet the demands of students, lecturers, and administrators.

TEST CASES:

Student Creates a Booking:

Test Case Number	1
Test Scenario	To successfully create a venue booking by a student
Test Steps	<ul style="list-style-type: none">- Student logs into the system- Student navigates from the main menu to the make venue booking form.- Student filters venues by the required capacity- Student chooses a venue from the available options.- Student selects a suitable time and date for the booking.- Student confirms the booking.
Prerequisites	<ul style="list-style-type: none">- Student user account is already in existence.- At least one venue and available time slots exist in the system.
Test Data	Student, Venue, time and date slot.
Values for test data	Student: ismaaeel@gmail.com Venue: FNB 33 Time Slot: Date: 2024-11-11, Time: 12:30 AM Additional Information: None
Expected/Intended Results	Upon successful booking, the system should display a confirmation message to the student, a booking ID and the booking details should be stored in the system and the user is directed to the homepage.
Actual Results	As expected.
Test Status [Pass/Fail]	Pass

Lecturer Updates Booking:

Test Case Number	2
Test Scenario	To successfully update a booking as a lecturer
Test Steps	<ul style="list-style-type: none"> - Lecturer logs into the system - Lecturer navigates from the main menu to the make view and manage bookings form - Lecturer is then able to view all their previous bookings. - Lecturer can then change the date and/or timeslot for the booking or change the venue from a list of available venues. - Lecturer selects the save changes button and a message appears to confirm the booking. - The system updates the venue and/or date to the changes made by the Lecturer
Prerequisites	<ul style="list-style-type: none"> - Lecturer user account is already in existence. - At least one venue and available time slots exist in the system.
Test Data	User, lecturer, time and date slot.
Values for test data	Initial Booking Data: Lecturer: dave@wits.ac.za Venue: FNB35 Time Slot: Date: 2024-11-30, Time: 12:30 AM Additional Information: None Updated Booking Data: Venue: Wits Science Stadium Time Slot: Date: 2024-11-30, Time: 12:30
Expected/Intended Results	Upon successful update booking, the system should display a confirmation message to the student, a booking ID and the updated booking details should be stored in the system and the user is directed to the homepage.
Actual Results	As expected.
Test Status [Pass/Fail]	Pass

Update Lecturer Profile:

Test Case Number	3
Test Scenario	To update a lecturer's profile by an admin
Test Steps	<ul style="list-style-type: none"> - Admin logs into the system - Admin navigates from the main menu to the update lecturer page. - Admin chooses the lecturer (lecturer ID) whose profile needs to be updated. - Admin gets new lecturer details. - Admin makes the necessary changes to the profile. - Admin confirms the update.
Prerequisites	<ul style="list-style-type: none"> - Admin user account is created. - At least one lecturer is registered in the system.
Test Data	Admin, lecturer, profile changes (phone number, email etc).
Values for test data	Administrator: MainAdmin@wits.ac.za Lecturer: Dave Schofield (Old Email): dave@wits.ac.za Profile Changes(Email): daveschofield@gmail.com
Expected/Intended Results	After the update, the system should display a confirmation message, and the lecturer's profile information should reflect the changes made by the administrator. The user is directed to the homepage.
Actual Results	As expected
Test Status [Pass/Fail]	Pass

INFO2000A - PEER EVALUATION

Team No: 1

Team Name: **EZPeazy Solutions**

Date: **18/10/2024**

Each team is required to complete a peer evaluation. The following aspects of the team's work should be taken into account when deciding on the percentage contribution to allocate to each member:

- Quality of individual contribution
- Timely submission of work
- Online Availability and communication

The following principles apply:

1. Every team member must agree on percentage contributions and sign the form.
2. The evaluation is not anonymous, but an adult assessment of contribution.
3. Think carefully about your assessment of your peers. Try to avoid extreme assessments unless they are fully justified.
4. In a team of five (5), if each member contributed equally, then each would be allocated 20%, in a team of six (6) each would be allocated 16.67%.
5. A team member who scores higher than the average will be awarded a project mark that is proportionally above the mark obtained by the team.
6. A team member who scores lower than the average will be awarded a project mark that is proportionally below the mark obtained by the team.
7. Extreme cases, e.g. where a student has made no contribution at all, will be handled on a case by case basis by the project coordinator (lecturer)

Member	Member Name	Contribution (%)
1	Abdullah Ali	16.67%
2	Mikaeel Suffla	16.67%
3	Mohammed Bangie	16.67%
4	Raeesa Lorgat	16.67%
5	Ismaaeel Shaikh Ebrahim	16.67%
6	Mohammed Essof	16.67%
Total		100%

Team Signatures

Member	Member Signatures
1	<i>A. Ali</i>
2	<i>M. Suffla</i>
3	<i>M. Bangie</i>
4	<i>R. Lorgat</i>
5	<i>I.S. Ebrahim</i>
6	<i>M. Essof</i>

Motivation (where there are issues that need explanation):

--

--