Boost Physio Clinic (BPC) Booking System Report

1. Introduction

The **Boost Physio Clinic (BPC) Booking System** is a Java-based application designed to manage appointments for physiotherapists and their patients. The system allows users (patients) to book, cancel, change, and attend appointments, while tracking physiotherapists' appointment attendance, cancellations, and history. The system provides comprehensive functionalities like generating final reports on attended treatments, canceled treatments, and ranking physiotherapists based on the number of attended treatments. Booking and canceling/ Rebooking appointments.

- Add/ remove patients
- Tracking treatment attendance
- Attend a treatment appointment
- Maintaining patient and physiotherapist information
- Generating detailed final reports

This system is console-based and runs as a standalone application without the use of a database or external services.

2. Key Features and Functionality

- **Booking Functionality**:
    - Patients can book an appointment with available physiotherapists. They can select from a list of available treatment times.
    - The system generates unique booking IDs for each appointment.
- **Cancellation Functionality**:
    - Patients can cancel their booked appointments, with their details tracked for historical reference.
    - Canceled treatments are marked as AVAILABLE for rebooking by other patients.
    - A history of canceled treatments is maintained, storing patient and physiotherapist names, along with the reason for cancellation.

- **Rebooking Functionality**:
    - If a patient chooses to cancel an appointment, they can rebook a new appointment with an available physiotherapist and treatment time.
    - The system checks for conflicts in scheduling (e.g., double bookings for the same time).
- **Attendance Tracking**:
    - Patients can mark appointments as attended, and the system tracks each physiotherapist's attendance.
    - The system generates a ranking of physiotherapists based on the number of treatments attended.
- **Final Report Generation**:
    - The system generates a final report that includes:
        - ❖ A list of all appointments categorized by status.
        - ❖ A ranking of physiotherapists by the number of attended appointments.
        - ❖ A section displaying canceled treatments.

3. Input Validation

The system implements several input validations checks to ensure smooth interaction with the user. These include:

- Integer Validation: When selecting a booking or cancellation option, the system ensures the input is an integer to avoid errors caused by invalid characters.
- Range Checks: User selections are checked to ensure that they are within the valid range of options (e.g., selecting an available treatment from the list).
- Non-null Validation: The system checks if an object is null (e.g., patient or physiotherapist) before performing any operations that depend on them.

4 Assumptions Made:

4.1 System Assumptions

- The application is for in-clinic use only (no remote access or database).
- Security (e.g., login, roles) is out of scope.

- All data exists in memory during runtime.
- Data such as physiotherapists and patients are initialized at startup.
- No integration with external calendar or notification systems.

4.2 Booking Assumptions

- Bookings are made on a first-come, first-served basis.
- A treatment can only be booked if it is in the **AVAILABLE** status.
- Canceling a treatment sets it back to **AVAILABLE** and logs it for historical reference.
- Attending a treatment changes its status to **ATTENDED**.
- Each patient can book only one treatment per time slot.

4.3 Reporting Assumptions

- **ATTENDED** and **BOOKED** treatments are included in the main report.
- **CANCELLED** treatments are shown in a separate section.
- Physiotherapists are ranked by the number of attended treatments.

5. System Design & Implementation Overview

Main Class

- The Main class controls the core user interaction flow, managing all user inputs and triggering the necessary methods for booking, cancellation, attendance, and report generation.
- It serves as the entry point for executing the program and ensures the interaction sequence is user-friendly.

Treatment Management

- Treatment objects represent the scheduled treatment, associated with both a patient and a physiotherapist.

- Appointment Status: Each treatment has a status (AVAILABLE, BOOKED, ATTENDED, CANCELLED), allowing the system to determine if a treatment can be booked or attended.

Cancelled Treatment Record

- This class is used to store information about canceled treatments, keeping track of when and by whom a treatment was canceled, along with the patient and physiotherapist involved.
- The canceled treatment record is added to a list for future reference.

5.1 Class Summary

Class	Responsibility
BookingManager	Main controller for managing physiotherapists, patients, treatments, and reports.
Physiotherapist	Stores details and expertise of each physiotherapist.
Treatment	Represents a single treatment appointment (time, physiotherapist, patient, status).
Patient	Stores patient ID, name, address, and phone number.
AppointmentStatus	Enum: AVAILABLE, BOOKED, ATTENDED, CANCELED.
CanceledTreatmentRecord	Stores snapshot details of a canceled treatment for audit and reporting.

5.2 System Workflow

1. **Initialization (via Main)**:
    - Adds 3 physiotherapists.
    - Each has preloaded treatments for 4 weeks.
    - Several patients are registered.

2. **Booking Process**:
    - Patient selects a treatment ID.
    - If status == AVAILABLE, booking proceeds.

- Treatment status becomes BOOKED.

3. **Canceling a Booking**:
    - Treatment status becomes AVAILABLE.
    - Treatment is logged in List<CanceledTreatmentRecord>.

4. **Attending Treatment**:
    - status set to ATTENDED.

5. **Final Report Generation**:
    - Lists all ATTENDED and BOOKED treatments grouped by physiotherapist.
    - Physiotherapists are ranked.
    - Canceled treatments shown in separate section.

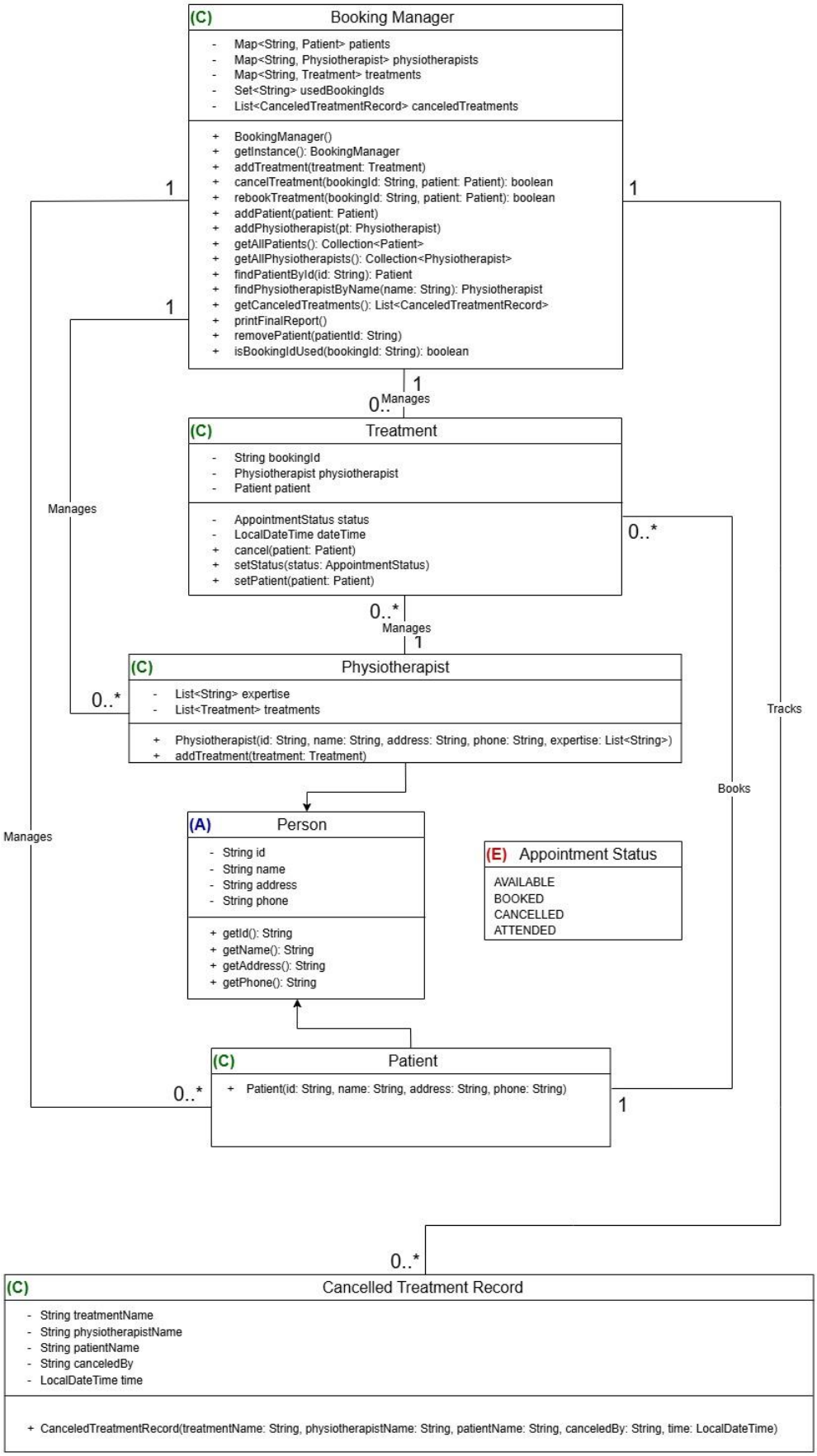
5.3 Code Implementation Details

- **Treatment Booking and Cancellation Logic:**
    - Booking: When a patient books an appointment, the system checks for availability and assigns a unique booking ID.
    - Cancellation: The system validates the patient's selection, cancels the treatment, and then updates its status to AVAILABLE for rebooking. The canceled treatment is recorded in the CanceledTreatmentRecord.
    - Attendance: A patient marks an appointment as attended, which updates the treatment status and contributes to physiotherapist attendance counts.
- **Data Handling:**
    - The system uses a List to store treatments, and a Map to track the number of attended appointments for each physiotherapist. The map is used for generating rankings based on attendance.

6. UML Class Diagram

Attached below is the UML diagram showing class relationships and structure:

Boost Physio Clinic UML Diagram



7. JUnit Testing

JUnit tests were implemented to ensure the correctness of the system's core components, including:

- BookingManager: Tests verify that treatments can be booked, canceled, and tracked correctly. Edge cases like invalid patient IDs and conflicting appointments are tested.
- Treatment: Tests ensure that treatments can change statuses appropriately (e.g., from AVAILABLE to BOOKED), and the cancellation process works as expected.
- Physiotherapist: Ensures physiotherapists' treatment records are correctly updated when treatments are booked, canceled, or attended.

7.1 Tools and Libraries

- JUnit 5
- Assertions: assertEquals, assertTrue, assertFalse, assertNotNull

7.2 Coverage Areas

Test Class	**Purpose**
BookingManagerTest	Tests booking, canceling, report generation, patient search, and treatment logic.
TreatmentTest	Validates status transitions and canceling behavior.
PatientTest	Confirms patient creation and field access.
PhysiotherapistTest	Ensures treatments are properly associated and accessed.
CanceledTreatment Test	CanceledTreatment test verifies correct tracking, status update, and reporting.
PersonTest	Tests Person class functionality, validating attributes like name, ID, and contact.

7.3 Sample JUnit Code

```
@Test
    public void testBookingAndCanceling()
{
    BookingManager manager = new BookingManager();
    Patient patient = new Patient("P001", "Alice", "10 Lane", "1234567890");
    manager.addPatient(patient);
    Physiotherapist physio = new Physiotherapist("PT1", "Dr. A", "1 Clinic Road",
    "987654321", List.of("Neck Pain"));
    manager.addPhysiotherapist(physio);
    Treatment t = new Treatment("T100", physio, LocalDateTime.now().plusDays(1));
    manager.addTreatment(t);
    assertTrue(manager.bookTreatment("T100", patient));
    assertTrue(manager.cancelTreatment("T100", patient));
    assertEquals(AppointmentStatus.AVAILABLE, t.getStatus());
}
```

8. Refactoring Activities

Throughout development, the following improvements were made:

- **Code Decomposition**: Complex methods in BookingManager were broken into private helpers.
- **Enum Integration**: All status logic centralized through AppointmentStatus.
- **Object Responsibility**: Shifted status transitions into Treatment class.
- **Historical Logging**: Introduced CanceledTreatmentRecord for canceled treatments.
- **Test Coverage Increase**: Added edge case tests for invalid booking, double-booking, etc.

9. Design Patterns and Principles

9.1 Principles

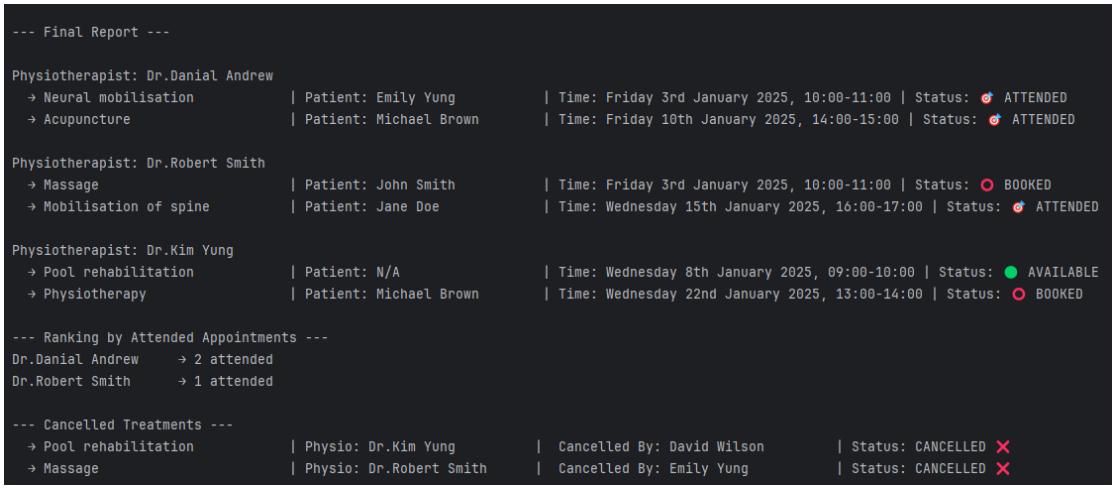
- **Single Responsibility**: Every class has one focused role.
- **Open/Closed Principle**: System can be extended (e.g., new statuses or reports) without modifying existing logic.
- **Encapsulation**: All fields are private with public accessors.
- **DRY (Don't Repeat Yourself)**: Common logic abstracted to helper methods.

9.2 Patterns

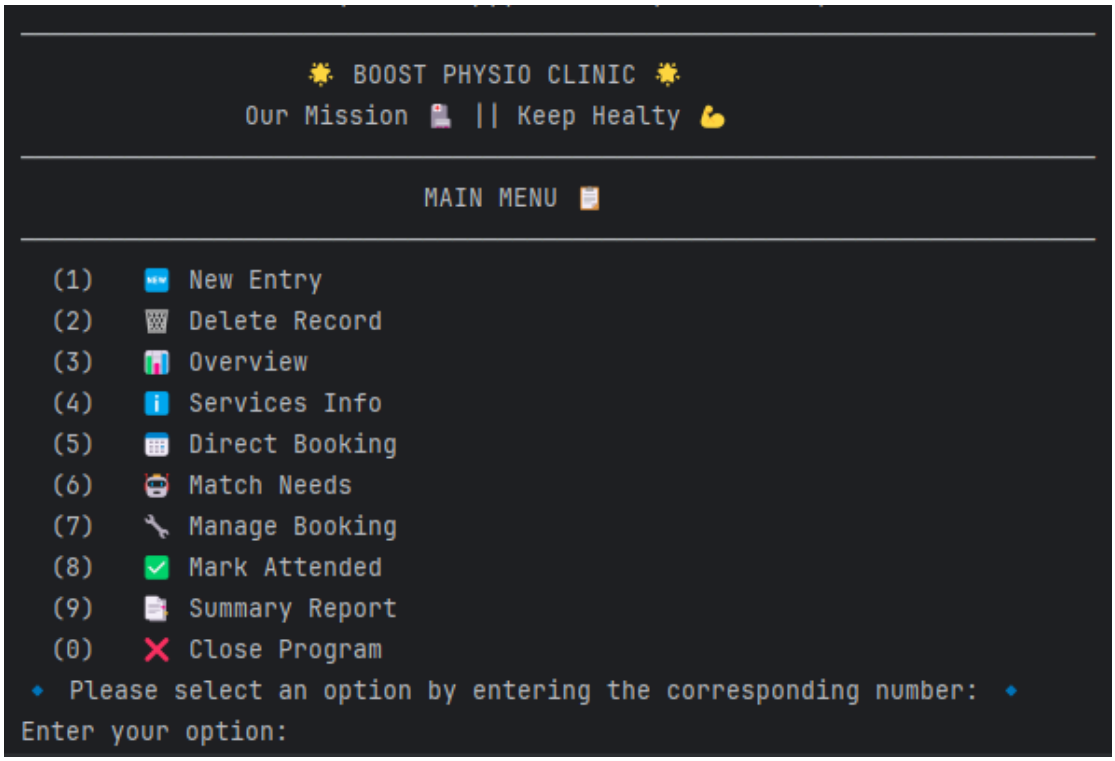
- **Factory Pattern**: Used in Main to set up predefined system objects.
- **Strategy Pattern** *(planned enhancement)*: To enable customizable reporting.
- **Enum State Pattern**: Clean use of AppointmentStatus to define treatment life cycle.

10. Samples Output

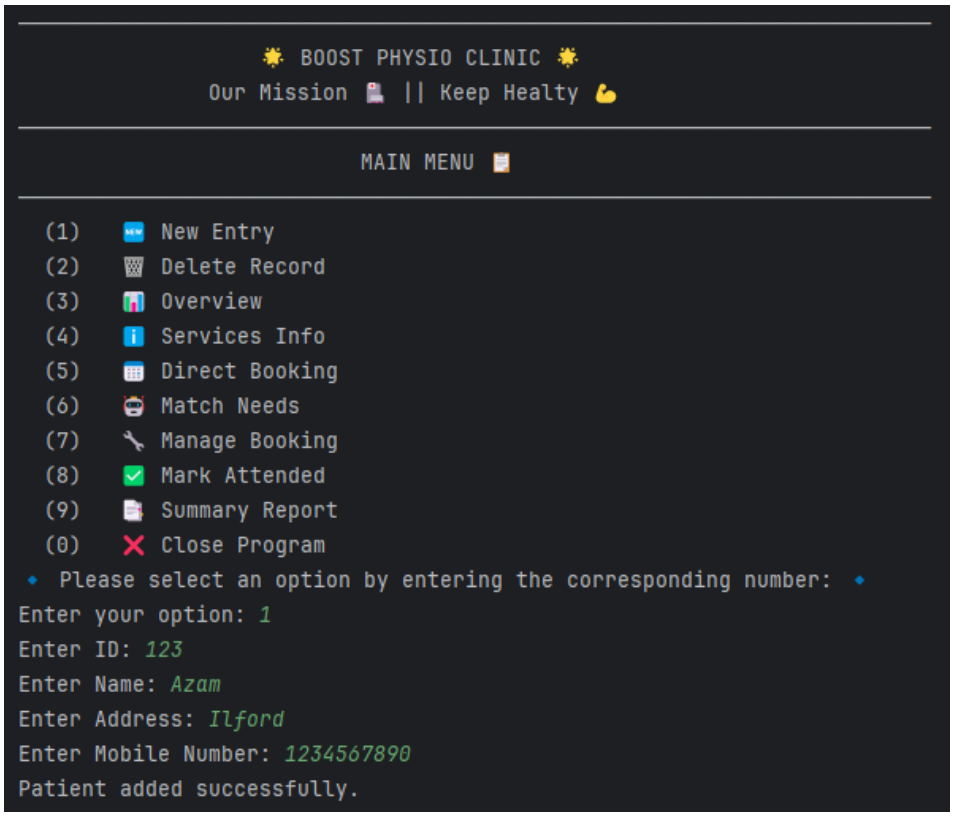
=== Final Report or Summary Report ===



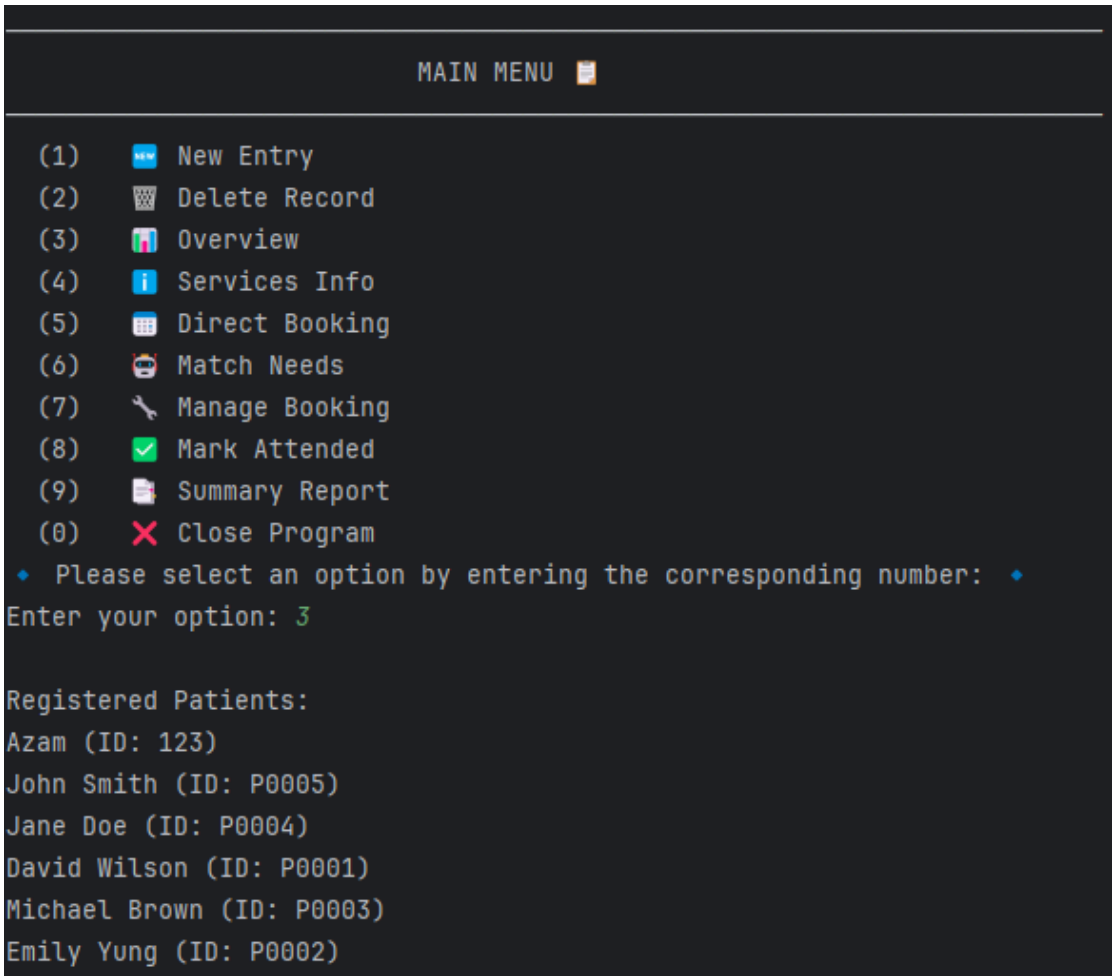
=== User Interface ===



=== Add Patient ===



=== Add Patient View Record ===



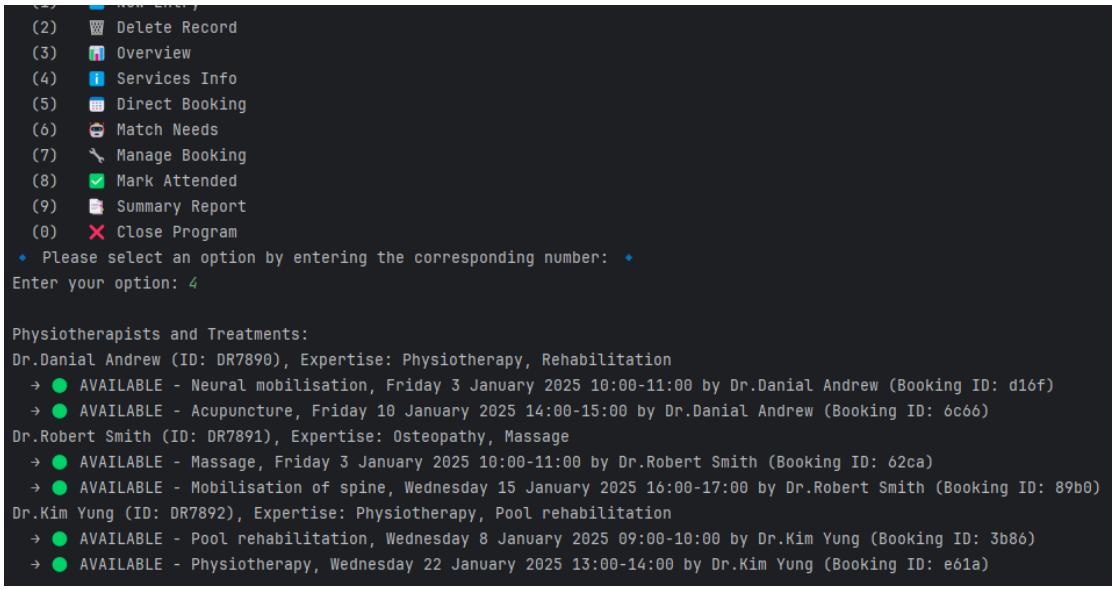
=== Remove Patient ===

```
                    BOOST PHYSIO CLINIC
                Our Mission || Keep Healty

                          MAIN MENU

  (1)   New Entry
  (2)   Delete Record
  (3)   Overview
  (4)   Services Info
  (5)   Direct Booking
  (6)   Match Needs
  (7)   Manage Booking
  (8)   Mark Attended
  (9)   Summary Report
  (0)   Close Program
 Please select an option by entering the corresponding number:
Enter your option: 2
Enter Patient ID to remove: 123
Patient removed successfully.
```

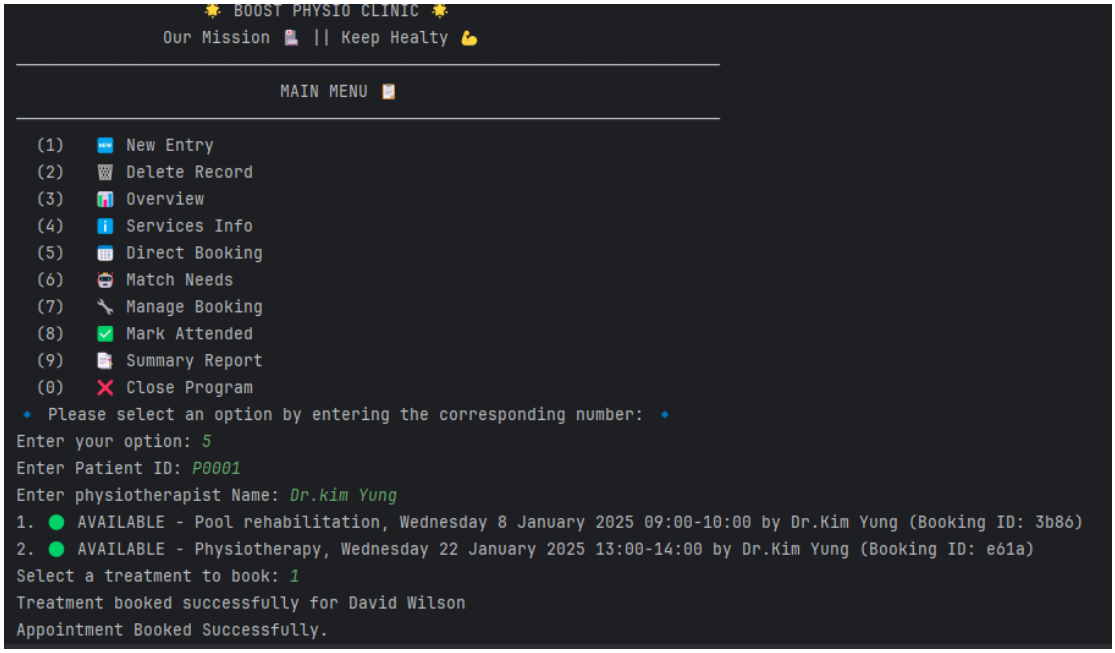
=== Delete Patient View record ===

```
                Our Mission || Keep Healty

                          MAIN MENU

  (1)   New Entry
  (2)   Delete Record
  (3)   Overview
  (4)   Services Info
  (5)   Direct Booking
  (6)   Match Needs
  (7)   Manage Booking
  (8)   Mark Attended
  (9)   Summary Report
  (0)   Close Program
 Please select an option by entering the corresponding number:
Enter your option: 3

Registered Patients:
John Smith (ID: P0005)
Jane Doe (ID: P0004)
David Wilson (ID: P0001)
Michael Brown (ID: P0003)
Emily Yung (ID: P0002)
```

=== View All Treatments & Physiotherapists ===

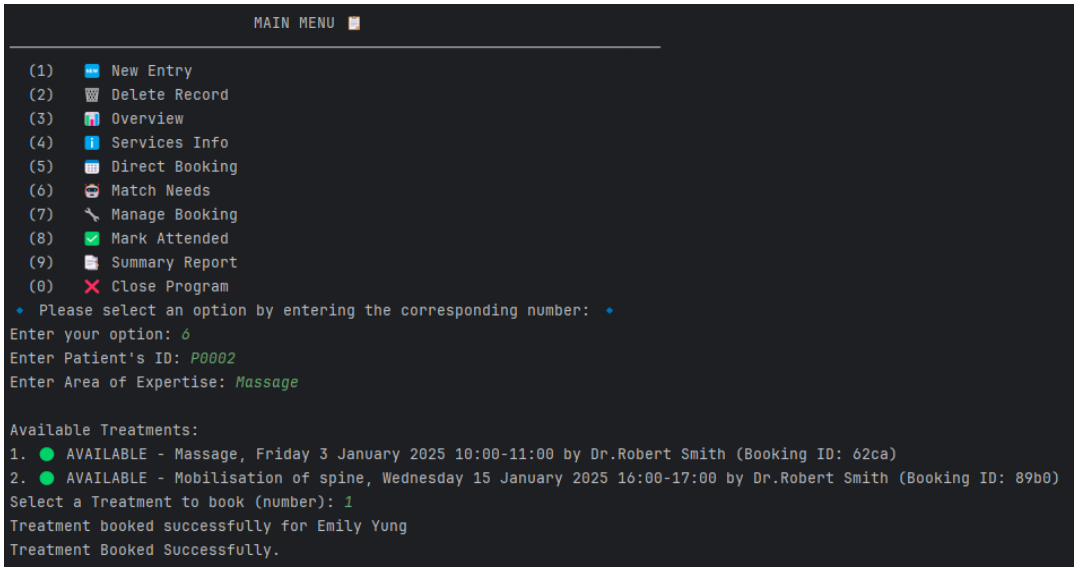


=== Booked Appointment by Physiotherapist ===



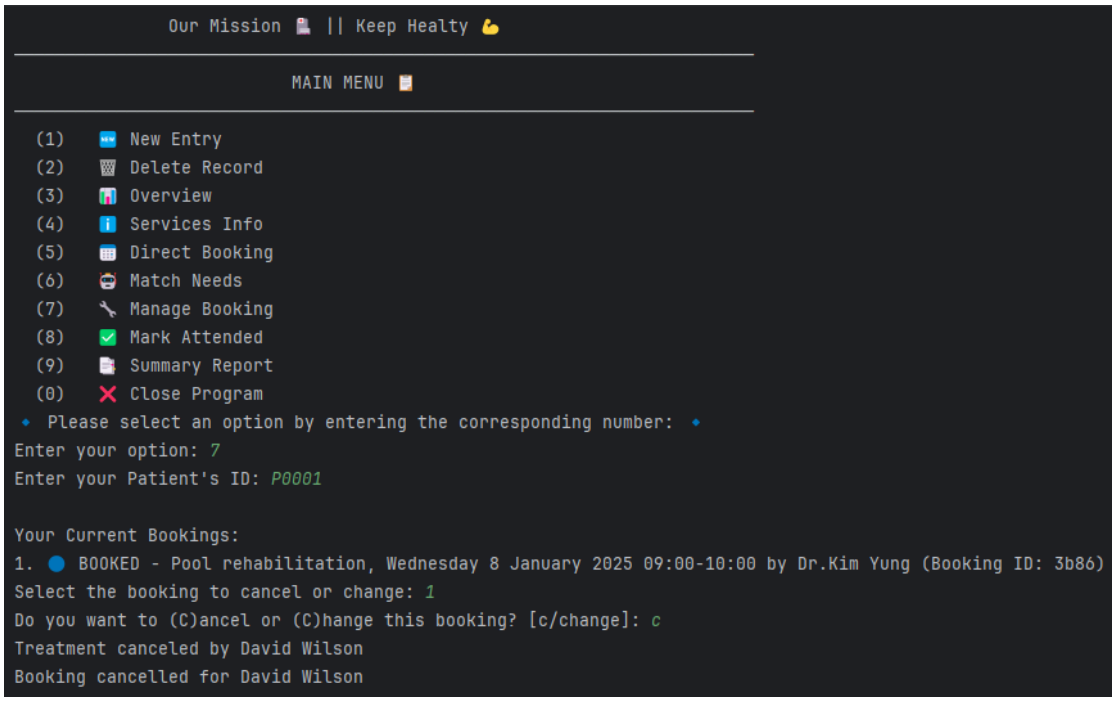
=== Booked Appointment by Physiotherapist Final Report or Summary Report & If an appointment has been booked, another patient cannot book it again ===

=== Booked Appointment by Expertise & If an appointment has been booked, another patient cannot book it again ===

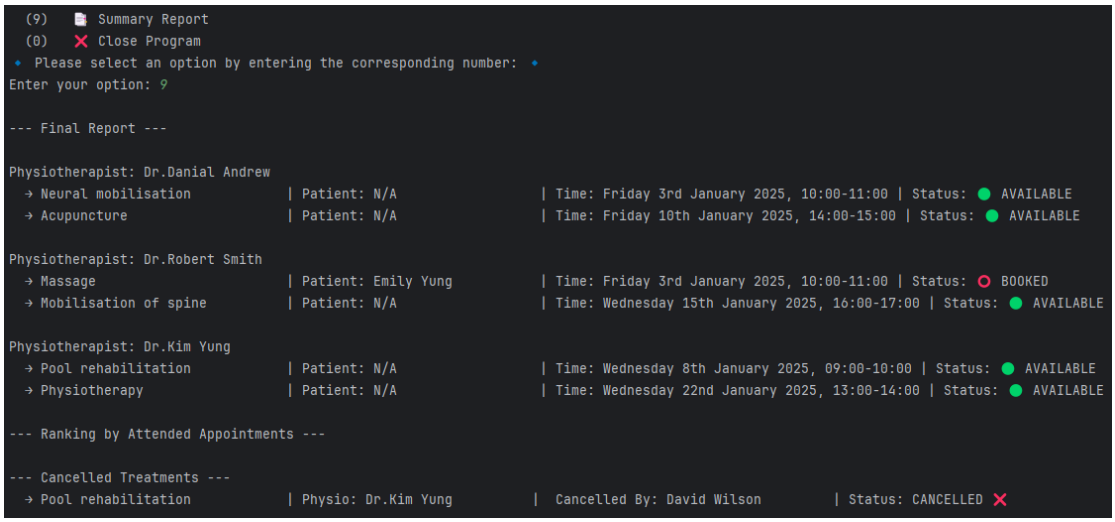


=== Booked Appointment by Expertise Final Report or Summary Report ===

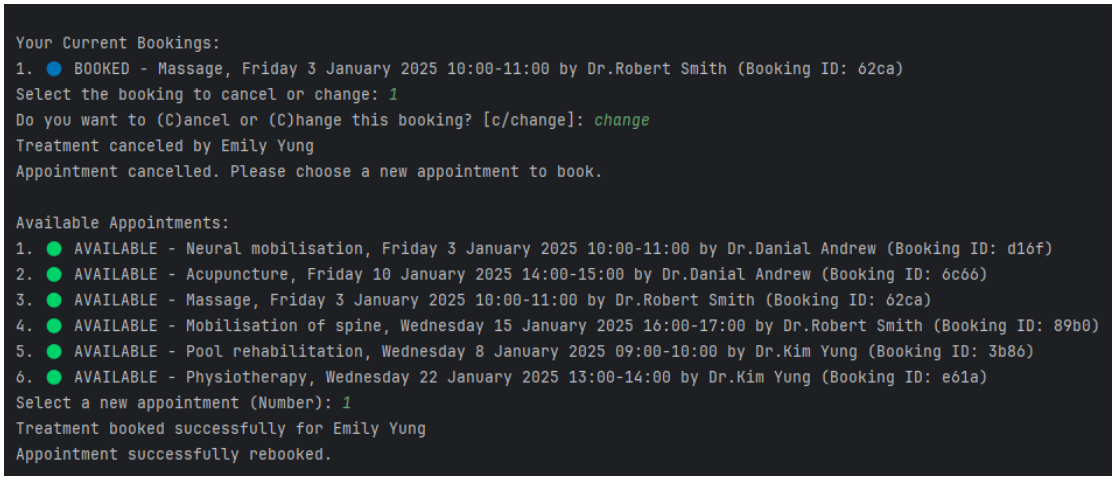
=== Cancelled Appointment ===



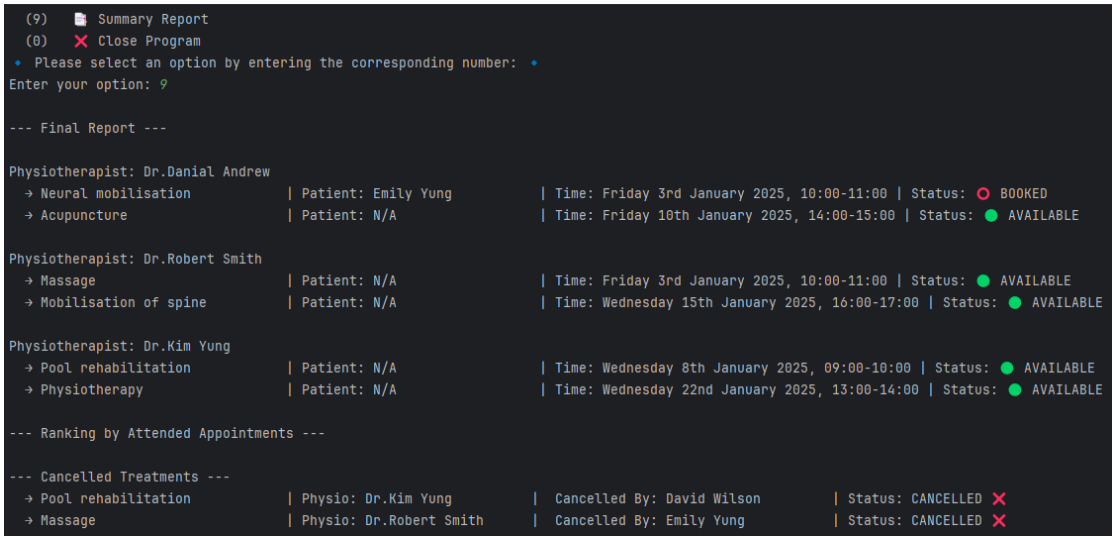
=== Cancelled Appointment showed in Final Report or Summary Report & Cancelled Appointment Available for Other Patients ===



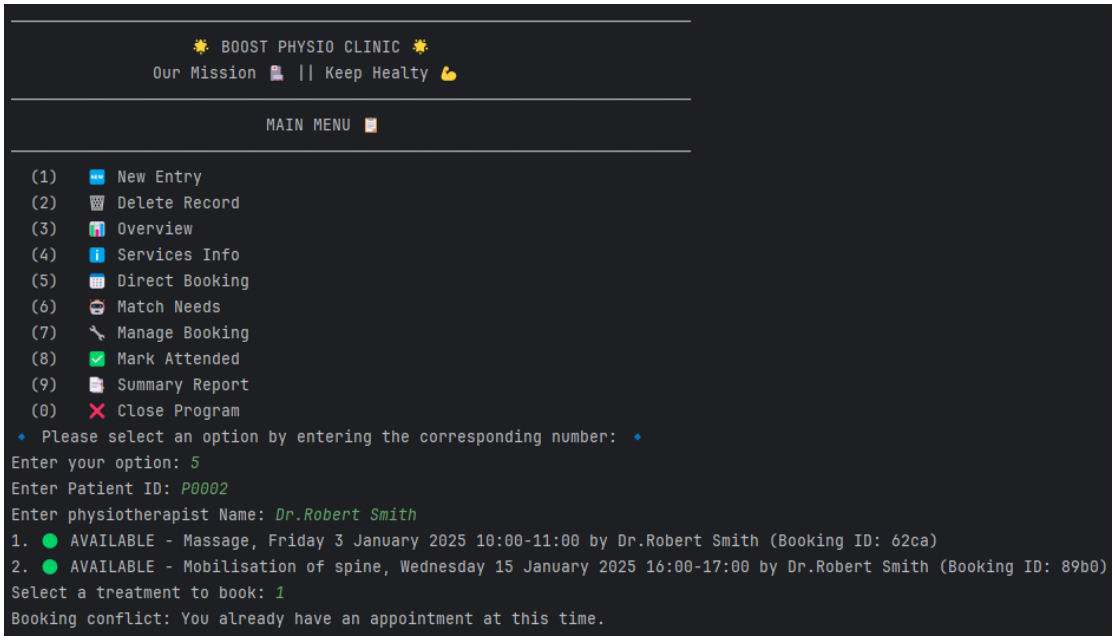
=== Changed Appointment ===



=== Changed Appointment in Final Report or Summary Report ===



=== No Repeated Booking: A patient cannot book two appointments running at the same time (for testing, in my timetable, set up at least two appointments running at the same time by different physiotherapists and Same time===



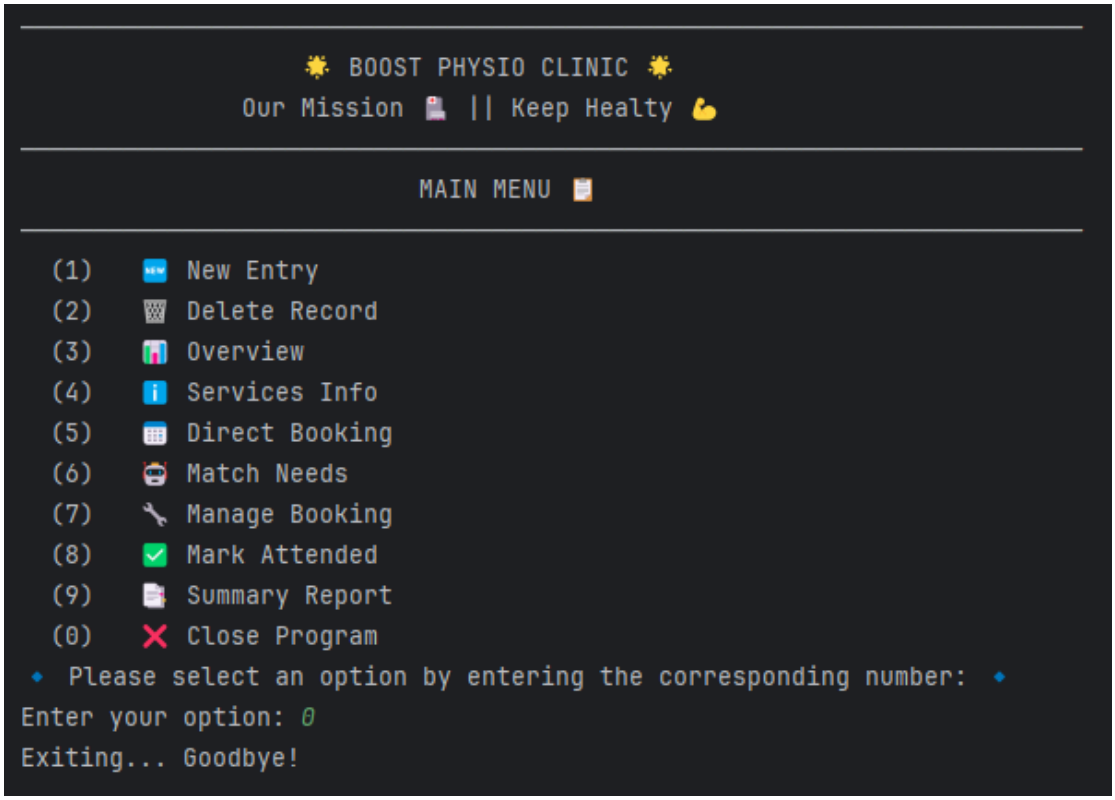
=== Attend an Appointment===

```
──────────────────────────────────────────────────
                🌟 BOOST PHYSIO CLINIC 🌟
              Our Mission 🚀 || Keep Healty 💪
──────────────────────────────────────────────────
                         MAIN MENU 📋
──────────────────────────────────────────────────
  (1)   🆕 New Entry
  (2)   🗑️ Delete Record
  (3)   📊 Overview
  (4)   ℹ️ Services Info
  (5)   📅 Direct Booking
  (6)   🤖 Match Needs
  (7)   🔧 Manage Booking
  (8)   ✅ Mark Attended
  (9)   📑 Summary Report
  (0)   ❌ Close Program
🔹 Please select an option by entering the corresponding number: 🔹
Enter your option: 8
Enter your Patient's ID: P0002

Your Booked Appointments:
1. 🔵 BOOKED - Neural mobilisation, Friday 3 January 2025 10:00-11:00 by Dr.Danial Andrew (Booking ID: d16f)
Select the Appointment you are attending (number): 1
Appointment Roberted as Attended.
```

=== Attended an Appointment in Final Report or Summery Report and also showing by Attended Appointments===

```
Enter your option: 9

--- Final Report ---

Physiotherapist: Dr.Danial Andrew
  → Neural mobilisation           | Patient: Emily Yung          | Time: Friday 3rd January 2025, 10:00-11:00 | Status: 🎯 ATTENDED
  → Acupuncture                   | Patient: N/A                 | Time: Friday 10th January 2025, 14:00-15:00 | Status: 🟢 AVAILABLE

Physiotherapist: Dr.Robert Smith
  → Massage                       | Patient: N/A                 | Time: Friday 3rd January 2025, 10:00-11:00 | Status: 🟢 AVAILABLE
  → Mobilisation of spine         | Patient: N/A                 | Time: Wednesday 15th January 2025, 16:00-17:00 | Status: 🟢 AVAILABLE

Physiotherapist: Dr.Kim Yung
  → Pool rehabilitation           | Patient: N/A                 | Time: Wednesday 8th January 2025, 09:00-10:00 | Status: 🟢 AVAILABLE
  → Physiotherapy                 | Patient: Michael Brown       | Time: Wednesday 22nd January 2025, 13:00-14:00 | Status: 🔴 BOOKED

--- Ranking by Attended Appointments ---
Dr.Danial Andrew     → 1 attended

--- Cancelled Treatments ---
  → Pool rehabilitation           | Physio: Dr.Kim Yung          |  Cancelled By: David Wilson         | Status: CANCELLED ❌
  → Massage                       | Physio: Dr.Robert Smith      |  Cancelled By: Emily Yung           | Status: CANCELLED ❌
```

=== Good Bye===


BOOST PHYSIO CLINIC
Our Mission || Keep Healty
MAIN MENU
(1) New Entry
(2) Delete Record
(3) Overview
(4) Services Info
(5) Direct Booking
(6) Match Needs
(7) Manage Booking
(8) Mark Attended
(9) Summary Report
(0) Close Program
Please select an option by entering the corresponding number:
Enter your option: 0
Exiting... Goodbye!

11. Future Enhancements

1. **GUI Development**: Using JavaFX or Swing.
2. **Persistent Storage**: Integrate with SQLite or PostgreSQL.
3. **REST API Layer**: Spring Boot service for web-based interface.
4. **Authentication**: Admin and Reception roles.
5. **Calendar Integration**: Real-time treatment schedules.
6. **Email Notifications**: Reminders for appointments.

12. Conclusion

The BPC Booking System successfully fulfills the core requirements with:

- Strong adherence to object-oriented principles
- Clean modular design
- Full JUnit coverage
- Expandability for future features

The Boost Physio Clinic Booking System successfully manages treatment appointments, cancellations, and rebooking, with built-in validation to prevent invalid input. It tracks patient treatment histories and provides detailed reports to assist clinic staff in evaluating physiotherapists' performance. While functional, the system could be further enhanced with a GUI, notifications, and integration with other systems for better usability and automation. It is robust, maintainable, and ready for further development as a complete clinical appointment solution.

13. UML Diagram of Boost Physio Clinic (BPC)

.

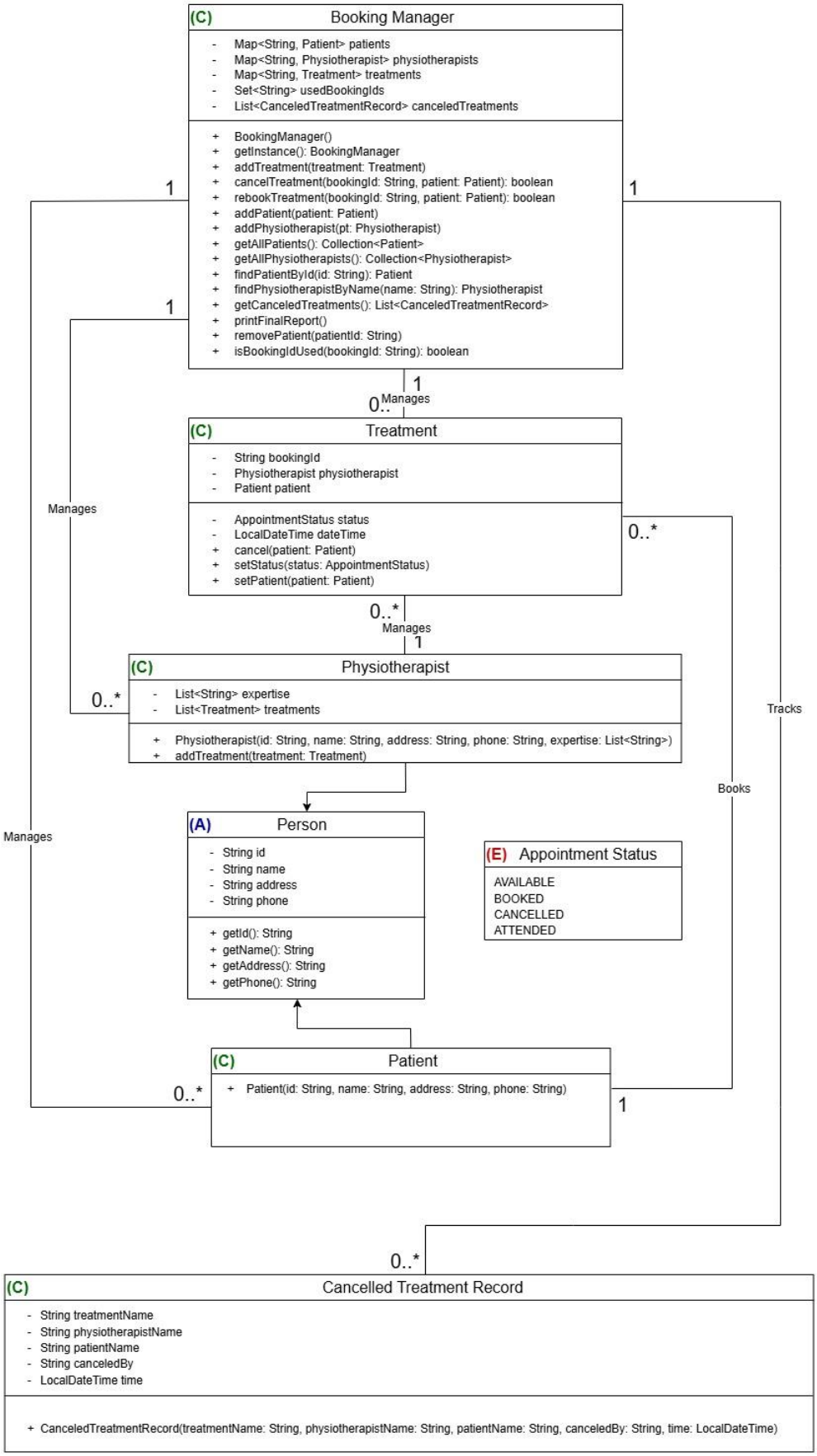
.

.

.

.

Boost Physio Clinic UML Diagram



14. GitHub Commits of Boost Physio Clinic (BPC)