A black background with a black square

Description automatically generated with medium confidence

CSE6224 Software Requirements Engineering

Campus Wellness Portal with

Medical System and Fitness Center

Integration Project – Part 1

SRS Document

Tutorial Section: TT6L

Group 4

|  |  |
| --- | --- |
| Group member | Student ID |
| Nicholas Thong Meng Shui | 241UC2415Y |
| Muhammad Anas bin Khairul Azman | 241UC2401Z |
| Fikrul Amsyar |  |

**Table of Contents**

[Changelog Table: 4](#_Toc198656960)

[1. Introduction 6](#_Toc198656961)

[1.1 Purpose 6](#_Toc198656962)

[1.2 Scope 6](#_Toc198656963)

[1.3 Product Overview 7](#_Toc198656964)

[1.3.1 Product Perspective 7](#_Toc198656965)

[1.3.2 Product functions 10](#_Toc198656966)

[1.3.3 User Characteristics 12](#_Toc198656967)

[1.3.4 Product Limitations 12](#_Toc198656968)

[2.0 References 13](#_Toc198656969)

[3.0 Requirements 13](#_Toc198656970)

[3.1 Functional Requirement 13](#_Toc198656971)

[3.1.5 Cancel Health Centre Appointments 15](#_Toc198656972)

[3.1.6 Track Wellness Goals 17](#_Toc198656973)

[3.1.7 View Fitness Class Notifications 19](#_Toc198656974)

[3.1.8 View Personalized Health Tips 21](#_Toc198656975)

[3.1.9 Manage Health Centre Appointments 23](#_Toc198656976)

[3.1.10 Manage Fitness Class Booking 25](#_Toc198656977)

[3.1.11 Integrate Wellness Data Across Systems 27](#_Toc198656978)

[3.2 Performance Requirements 29](#_Toc198656979)

[3.3 Usability Requirements 29](#_Toc198656980)

[3.4 Interface Requirements 29](#_Toc198656981)

[3.4.1 Login Interface 29](#_Toc198656982)

[3.4.2 Health Centre Appointment Interface 29](#_Toc198656983)

[3.5 Logical Database Requirements 30](#_Toc198656984)

[3.6 Design Constraints 30](#_Toc198656985)

[3.7 Software System Attributes 31](#_Toc198656986)

[3.7.1 Security 31](#_Toc198656987)

[3.8 Supporting Information 32](#_Toc198656988)

[4.0 Verification 32](#_Toc198656989)

[5.0 Appendices 32](#_Toc198656990)

Pro



# Changelog Table:

|  |  |  |
| --- | --- | --- |
| Version | Author | Description |
| v1.0 | Nicholas Thong Meng Shui | * Created this document |
| V1.1 | Nicholas Thong Meng Shui | * Added information and content to 1.3.1 Product Perspective |
| V1.2 | Nicholas Thong Meng Shui | * Added information and content to 1.3.3 User Characteristics |
| V1.3 | Muhammad Anas bin Khairul Azman | * Added content to 1.3.2 Product Functions |
| V1.4 | Nicholas Thong Meng Shui | * Added Use Case Diagram to 3.1 Functions |
| V1.5 | Nicholas Thong Meng Shui | * Removed EHR System from 1.3.1 * Updated 1.3.2 Campus Recreation Facility Integration * Created and added content to table 3-1 in 3.1 functions * Added table labels to tables 1-1 and 3-1 |
| V1.5.1 | Muhammad Anas bin Khairul Azman | * Added Product Limitations in 1.3.4 * Added pretext for 3.1 Performance Requirements |
| V1.5.2 | Muhammad Anas bin Khairul Azman | * Added performance requirement and use case table for 3.1.5 |
| V1.5.3 | Muhammad Anas bin Khairul Azman | * Added functional requirement and use case table for 3.1.6, 3.1.7, 3.1.9 |
| V1.5.4 | Muhammad Anas bin Khairul Azman | * Added sequence diagrams for 3.1.5, 3.1.6, 3.1.7, 3.1.9 |
| V1.5.5 | Muhammad Anas bin Khairul Azman | * Fixed numbering for functional requirements |
| V.1.5.6 | Nicholas Thong Meng Shui | * Added Functional Requirements and use case table for 3.1.8, 3.1.10, 3.1.11 * Added Sequence Diagram for 3.1.8 |
| V1.5.7 | Muhammad Anas bin Khairul Azman | * Added cover page. * Added proper Table of Contents. * Added pre-text for upcoming chapters. * Added 3.2 Performance Requirements * Added 3.7 Security Requirements |
| V1.6 | Muhammad Anas bin Khairul Azman | * Added 3.4 Interface Requirements * Login interface and appointment interface |
| V1.6.1 | Nicholas Thong Meng Shui | * Updated figure 1.0 Use case diagram in 3.1 Functional Requirements * Updated table 3-1 Use case ID in 3.1 Functional Requirements * Updated use case table for 3.1.10, 3.1.11 * Added Sequence Diagrams for 3.1.10, 3.1.11 |

# Introduction

## Purpose

## Scope

## 1.3 Product Overview

### 1.3.1 Product Perspective

The Wellness Platform is a system that integrates the university health centre’s appointment system and campus recreation facility management software. Its primary role is to support student wellness through personalized health resources and wellness goals, integration with university medical appointment records, and interaction with recreational facilities.

As such, this system communicates and interoperates with several other components of the university's infrastructure. These include the university authentication system, recreation portal, and health portal. The system utilizes these interfaces to authenticate users, retrieve health data, enable health appointment scheduling, enable fitness class scheduling, and provide wellness recommendations.

1. System Interfaces

* **Authentication System**: The system relies on the university's login infrastructure to authenticate student users securely.
* **Recreation Portal**: Interfaces with the university’s recreation facility management software to enable fitness class booking and data sharing.
* **Health Portal**: Interfaces with the university’s health centre’s appointment system to allow students to schedule health appointments.

1. User Interfaces
   1. Wellness Platform website shall make use of white (RGB Hex code: #FFFFFF) for its overall background and navigation bar font colour.
   2. Wellness Platform website shall always have a black (RGB Hex code: #000000) for non-navigation bar font colour.
   3. Wellness Platform website shall always have a blue (RGB Hex code: #000EBF) navigation bar present at the top of the page.
2. Hardware Interfaces

The system supports standard student devices such as personal computers and mobile phones. No specialized hardware is required.

1. Software Interfaces

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Mnemonic** | **Specification Number** | **Version** | Reference |
| University Health Portal | UNIV\_HEALTH\_PORT | N/A | Current | [University Health Patient Portal](https://www.universityhealth.com/patient-visitor-resources/patients/patient-portal) |
| University Recreation Portal | UNIV\_REC\_PORTAL | N/A | Current | [Columbia University Recreation Portal](https://perec.columbia.edu/content/member-portal-resources) |
| OAuth 2.0 Authentication Framework | OAUTH2 | RFC 6749 | Latest | [OAuth 2.0 Specification](https://tools.ietf.org/html/rfc6749) |
| JSON Web Token Standard | JWT | RFC 7519 | Latest | [JWT Specification](https://tools.ietf.org/html/rfc7519) |
| RESTful API Interface | REST\_API | OpenAPI 3.0 | Latest | [OpenAPI Specification](https://swagger.io/specification/) |

Interfaces:

* University Health Portal
  + **Purpose**: To allow students to manage medical appointments and view personal health records.
  + **Interface Definition**: Web-based interface; provides scheduling, messaging, and medical record access. Integration through RESTful API.
  + **Reference**: University Health Patient Portal
* University Recreation Portal
  + **Purpose**: To enable scheduling and tracking of fitness classes and recreational activities.
  + **Interface Definition**: Booking and facility access information exchanged over REST API. Includes user authentication via SSO.
  + **Reference**: Columbia University Recreation Portal
* OAuth 2.0 Authentication Framework
  + **Purpose**: To authenticate students securely across university systems.
  + **Interface Definition**: Standard OAuth 2.0 protocol to handle login tokens and authorization. Interfaces with the university’s identity provider.
  + **Reference**: OAuth 2.0 Specification
* JSON Web Token (JWT)
  + **Purpose**: To transmit session data securely between frontend and backend systems.
  + **Interface Definition**: Encodes claim-based tokens using standard JWT format (RFC 7519), embedded in HTTP headers.
  + **Reference**: JWT Specification
* RESTful API Interface
  + **Purpose**: To facilitate integration between the wellness platform and external systems such as EHR, health portal, and recreation systems.
  + **Interface Definition**: Uses HTTP(S) with standard methods (GET, POST, PUT, DELETE). Data formatted as JSON. Described using OpenAPI Specification.
  + **Reference**: OpenAPI Specification

1. Communication Interfaces

* HTTPS protocol is used for secure communication with internal university systems.
* Authentication via OAuth 2.0 and JWT for cross-system data sharing.

1. Memory Constraints

The system is lightweight and web-based, requiring minimal local memory. All user data is stored and processed in cloud/university-hosted servers.

1. Operations

The Wellness Platform is intended to function under both user-driven and automated operations. The following operational requirements are identified:

1) Modes of Operation

* **User-Initiated Mode**: Students interact with the web application to:
  + Set and view wellness goals.
  + Book or cancel health centre or fitness appointments.
  + View personalized health resources.
* **Admin Mode**: Authorized staff access logs or analytics reports through secure dashboards.

2) Periods of Interactive and Unattended Operations

* Interactive Operations:
  + Occur during active user sessions.
  + Includes booking appointments, viewing data, or modifying goals.
* Unattended Operations:
  + Background data syncing with Health Portal and Recreation Facility systems.
  + Periodic refresh of user statistics (e.g., wellness goals, appointment reminders).
  + Runs on a scheduled basis (e.g., nightly or hourly intervals).

3) Data Processing Support Functions

* **Data Aggregation**: Collects fitness and medical data from multiple systems and consolidates it in the student dashboard.
* **Data Validation**: Ensures data accuracy and compliance with privacy regulations (e.g., HIPAA).
* **Error Logging**: Logs failures during API communication or user actions for diagnostics.
* **Monitoring**: System health checks and logging for server uptime and performance metrics.

1. Site Adaptation Requirements

The system requires configuration of university-specific endpoints (e.g., Recreation API URLs, fitness class schedule formats) during deployment. It also requires institution-specific policy parameters (e.g., data retention limits, HIPAA compliance flags).

1. Interfaces with Services

Integrates with:

* University SaaS-based portals (e.g., health appointment scheduling)
* Cloud storage services (if enabled)

### 1.3.2 Product functions

Campus Recreation Facility Integration

* Search and filter classes by their type, date and location.
* Book fitness classes.
* Notification for open slots in fitness classes.
* Google/Outlook calendar integration which syncs any booked fitness classes.
* Email reminders of booked fitness classes

University Health Portal Integration

* Set personal wellness goals (steps, workout, sleep).
* Track personal wellness goals.
* Dashboard of wellness goal progress.
* Weekly summary of wellness goals.
* Receive tailored health resources based on user preferences.
* Schedule appointments with the health centre.
* Cancel scheduled appointments.

### 1.3.3 User Characteristics

Table 1-1 User Characteristics:

|  |  |  |
| --- | --- | --- |
| User group | Description | Minimum Ability/Knowledge |
| Students | Students currently enrolled in the university | Possess basic computer skills |

### 1.3.4 Product Limitations

#### 1.3.4.1 General Limitations

The main limitations of our product comes from the fact that it is only accessible via web browser for ease of access for students. Furthermore, the use of this wellness portal is only accessible for students within the university and is not available for staff members of the university or students outside of the university.

# 2.0 References

# 3.0 Requirements

## 3.1 Functional Requirement

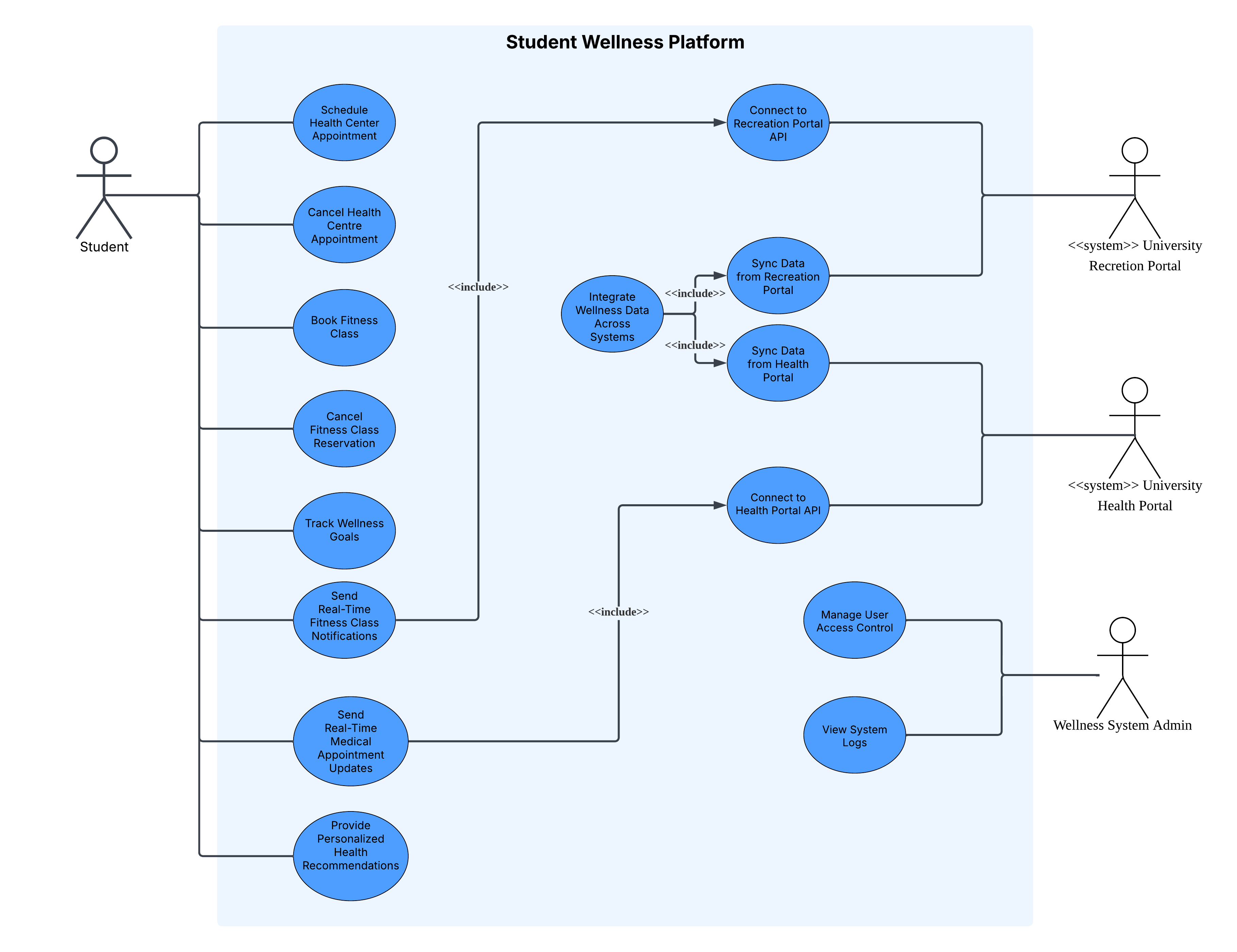


Figure 1.0 : Use case diagram

Table 3-1 Use Case ID:

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID** | **Actor** | **Use Case (Function)** | **Brief Description** |
| **UC\_001** | Student | Login | Allows the student to log into the system using their credentials. |
| **UC\_002** | Student | Book Fitness Class | Enables the student to select and book a fitness class from the available options. |
| **UC\_003** | Student | Cancel Fitness Class | Allows the student to cancel a previously booked fitness class. |
| **UC\_004** | Student | Schedule Health Centre Appointment | Allows the student to choose a time slot and book an appointment with the health centre. |
| **UC\_005** | Student | Cancel Health Centre Appointment | Enables the student to cancel a scheduled health centre appointment. |
| **UC\_006** | Student | Track Wellness Goals | Allows the student to set, monitor, and update personal wellness goals. |
| **UC\_007** | Student | View Notifications | Displays system-generated alerts or updates to the student. |
| **UC\_008** | Student | View Personalised Health Tips | Provides the student with curated health and wellness tips. |
| **UC\_009** | University Health Portal | Manage Health Centre Appointments | Receives and updates student health appointment bookings and cancellations. |
| **UC\_010** | University Recreation Portal | Manage Fitness Class Booking | Updates the fitness class schedule and availability based on bookings or cancellations. |
| **UC\_011** | System (Automated) | Integrate Wellness Data Across Systems | This system-level function operates as an internal module, synchronizing student wellness data across connected systems (e.g., Health Centre, Fitness). It may interact with internal databases or APIs to perform this integration automatically. |
| **UC\_012** | Wellness System Admin | Manage User Access Control | Allows the admin to create, modify, and delete user accounts and assign or revoke roles/permissions. |
| **UC\_013** | Wellness System Admin | View System Logs | Enables the admin to retrieve and review application and integration logs for monitoring and auditing. |

### 3.1.5 Cancel Health Centre Appointments

Functional Requirement for Cancel Health Centre Appointment

|  |  |
| --- | --- |
| **Requirement ID** | REQ\_501 |
| **Description** | System shall allow the user to cancel a health centre appointment. |
| **Author** | Muhammad Anas |

|  |  |
| --- | --- |
| **Requirement ID** | REQ\_502 |
| **Description** | System shall prompt the user to confirm if they want to cancel the selected appointment. |
| **Author** | Muhammad Anas |

|  |  |
| --- | --- |
| **Requirement ID** | REQ\_503 |
| **Description** | System shall display a confirmation message that the selected appointment has been cancelled. |
| **Author** | Muhammad Anas |

|  |  |
| --- | --- |
| **Requirement ID** | REQ\_504 |
| **Description** | System shall update user’s appointments to reflect cancellation of appointment. |
| **Author** | Muhammad Anas |

Use Case Table

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC005 | |
| **Feature** | F005 Cancel Health Centre Appointments | |
| **Purpose** | To allow users to cancel their scheduled health centre appointments. | |
| **Actor** | Student | |
| **Trigger** | User clicks on “Cancel Appointment” button on a given appointment. | |
| **Precondition** | User is in the manage health centre appointment screen | |
| **Scenario Name** | **Step** | **Action** |
| **Main Flow** | 1 | User clicks on manage health centre appointment screen. |
|  | 2 | User clicks on “Cancel Appointment” for a given appointment. |
|  | 3 | System displays dialog box to confirm if the user wants to cancel said appointment. |
|  | 4 | User clicks on “Yes” in the dialog box. |
|  | 5 | System updates user’s appointment table to reflect the cancelled appointment. |
|  | 6 | System displays confirmation message that the appointment has been cancelled. |
|  | 8 | System displays the health centre appointment screen with the cancelled appointment now updated. |
| **Alternate Flow – User clicks “No” in the dialog box** | 1 | User clicks on “No” in the dialog box. |
|  | 2 | System displays the health centre appointment screen. |
| **Rules** | * User must already have an existing scheduled appointment. | |
| **Author** | Muhammad Anas | |

Sequence Diagram for Cancel Health Centre Appointment

A diagram of a flowchart

Description automatically generated

### 3.1.6 Track Wellness Goals

Functional Requirement for Tracking Wellness Goals

|  |  |
| --- | --- |
| **Requirement ID** | REQ\_601 |
| **Description** | System shall display the user’s wellness goals over the past month in the dashboard. |
| **Author** | Muhammad Anas |

|  |  |
| --- | --- |
| **Requirement ID** | REQ\_602 |
| **Description** | System shall allow the user to set new wellness goals. |
| **Author** | Muhammad Anas |

|  |  |
| --- | --- |
| **Requirement ID** | REQ\_603 |
| **Description** | System shall allow the user to update their wellness goals. |
| **Author** | Muhammad Anas |

Use Case Table

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC006 | |
| **Feature** | F006 Track Personal Wellness Goals | |
| **Purpose** | To allow users to monitor, set and update their personal wellness goals. | |
| **Actor** | Student | |
| **Trigger** | User clicks on “Wellness Goals” button on the platform portal. | |
| **Precondition** | User is in the wellness platform portal. | |
| **Scenario Name** | **Step** | **Action** |
| **Main Flow – Monitoring wellness goals** | 1 | User clicks on “Wellness Goals” button on the platform portal. |
|  | 2 | System displays wellness goals dashboard to the user. |
| **Alternate Flow – Setting new wellness goals** | 3 | User clicks on “Set new wellness goal” button. |
|  | 4 | System prompts the user to input their wellness goal. |
|  | 5 | System updates the user’s wellness goal database. |
|  | 6 | System displays confirmation message that the wellness goal has been added. |
|  | 7 | System displays the updated wellness goal dashboard. |
| **Alternate Flow – Updating existing wellness goals** | 1 | User clicks on “Update” button on a given wellness goal. |
|  | 2 | System prompts the user to update the values for the wellness goal. |
|  | 3 | System updates the user’s wellness goal database. |
|  | 4 | System displays confirmation message that the wellness goal has been updated. |
|  | 6 | System displays the updated wellness goal dashboard |
| **Rules** | - | |
| **Author** | Muhammad Anas | |

A diagram of a wellness goal screen

Description automatically generated

### 3.1.7 View Fitness Class Notifications

Functional Requirement for Viewing Notifications

|  |  |
| --- | --- |
| Requirement ID | REQ\_701 |
| Description | System shall allow the user to view any fitness class notifications. |
| Author | Muhammad Anas |

|  |  |
| --- | --- |
| Requirement ID | REQ\_702 |
| Description | System shall display the fitness class notifications to the user. |
| Author | Muhammad Anas |

Use Case Table

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC007 | |
| **Feature** | F007 View Fitness Class Notifications | |
| **Purpose** | To allow users to view their fitness class notifications. | |
| **Actor** | Student | |
| **Trigger** | User clicks on “Notifications” button on the platform portal. | |
| **Precondition** | User is in the wellness platform portal. | |
| **Scenario Name** | **Step** | **Action** |
| **Main Flow** | 1 | User clicks on “Notifications” button on the platform portal. |
|  | 2 | System displays the fitness class notifications to the user. |
| **Rules** | * Users must be signed up/have booked a fitness class. * Without it, notifications will be empty. | |
| **Author** | Muhammad Anas | |

Sequence Diagram for viewing fitness class notifications

A diagram of a computer flow

Description automatically generated

### 3.1.8 View Personalized Health Tips

Functional Requirements for View Personalized Health Tips

|  |  |
| --- | --- |
| Requirement ID | REQ\_801 |
| Description | |  | | --- | | System shall retrieve health tips tailored to the student’s profile and goals. |  |  | | --- | |  | |
| Author | Nicholas Thong Meng Shui |

|  |  |
| --- | --- |
| Requirement ID | REQ\_802 |
| Description | System shall display retrieved health tips in the student’s dashboard. |
| Author | Nicholas Thong Meng Shui |

|  |  |
| --- | --- |
| Requirement ID | REQ\_803 |
| Description | System shall allow the student to request more details on any displayed tip. |
| Author | Nicholas Thong Meng Shui |

Use Case Table

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID** | UC\_008 | | |
| **Feature** | |  | | --- | |  |  |  | | --- | | View Personalized Health Tips | | | |
| **Purpose** | Provide curated health and wellness tips based on student profile. | | |
| **Actor** | |  | | --- | |  |  |  | | --- | | Student | | | |
| **Trigger** | Student selects “View Health Tips.” | | |
| **Precondition** | Student is logged in; tips database is accessible. | | |
| **Scenario Name** | **Step** | **Action** |
| **Main Flow – User viewing health tips** | 1 | |  |  | | --- | --- | |  | Student clicks “Health Tips” menu. | |
| 2 | |  | | --- | | System retrieves tailored tips (REQ\_801). | |
| 3 | |  |  | | --- | --- | |  | System displays tips list (REQ\_802). | |
| 4 | |  | | --- | | Student selects a tip for more details. | |
| 5 | System shows full tip details (REQ\_803). |
| **Alternate Flow – No Tips Available** | 2a | If no tips match profile, System displays “No personalized tips available.” |
| **Rules** | Tips must be relevant to current wellness goals. | | |
| **Author** | Nicholas Thong Meng Shui | |

A diagram of a flowchart

AI-generated content may be incorrect.

### 3.1.9 Manage Health Centre Appointments

Functional Requirement for Manage Health Centre Appointments

|  |  |
| --- | --- |
| Requirement ID | REQ\_901 |
| Description | System shall allow the user the view their health centre appointments (scheduled and cancelled). |
| Author | Muhammad Anas |

Use Case Table

|  |  |  |
| --- | --- | --- |
| **Use Case ID** | UC009 | |
| **Feature** | F009 Manage Health Centre Appointments | |
| **Purpose** | To receive and update a user’s scheduled and cancelled health centre appointments. | |
| **Actor** | University Health Portal | |
| **Trigger** | User clicks on “Schedule Appointment” or “Cancel Appointment” button on the “Health Centre Appointments” screen. | |
| **Precondition** | User is in the Health Centre Appointments screen. | |
| **Scenario Name** | **Step** | **Action** |
| **Main Flow – User scheduled an appointment** | 1 | User clicks on “Schedule Appointment” button on the Appointments screen. |
|  | 2 | System prompts the user to set appointment date. |
|  | 3 | System updates the appointments database for the user. |
|  | 4 | System displays updated appointments table to the user. |
| **Alternate Flow – User cancelled an appointment** | 1 | User clicks on “Cancel Appointment” on the Appointments screen |
|  | 2 | System displays dialog box to confirm if the user wants to cancel their appointment. |
|  | 3 | User clicks on the “Yes” button in the dialog box. |
|  | 4 | System updates the appointments database for the user. |
|  | 5 | System displays updated appointments table to the user. |
| **Rules** | - | |
| **Author** | Muhammad Anas | |

A diagram of a flowchart

Description automatically generated

### 3.1.10 Manage Fitness Class Booking

Functional Requirements for Manage Fitness Class Booking

|  |  |
| --- | --- |
| Requirement ID | REQ\_1001 |
| Description | System shall accept booking requests from the student platform. |
| Author | Nicholas Thong Meng Shui |

|  |  |
| --- | --- |
| Requirement ID | REQ\_1002 |
| Description | System shall update class availability upon each booking or cancellation. |
| Author | Nicholas Thong Meng Shui |

|  |  |
| --- | --- |
| Requirement ID | REQ\_1003 |
| Description | System shall confirm bookings or cancellations back to the student platform. |
| Author | Nicholas Thong Meng Shui |

|  |  |
| --- | --- |
| Requirement ID | REQ\_1004 |
| Description | System shall reject booking requests when no slots remain and notify the student. |
| Author | Nicholas Thong Meng Shui |

Use Case Table

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID** | UC\_010 | | |
| **Feature** | |  | | --- | |  |  |  | | --- | | Manage Fitness Class Booking | | | |
| **Purpose** | Keep class schedules and seat counts in sync with student bookings. | | |
| **Actor** | |  | | --- | |  |  |  | | --- | | University Recreation Portal | | | |
| **Trigger** | |  | | --- | |  |  |  | | --- | | Platform sends a booking or cancellation API call. | | | |
| **Precondition** | Recreation Portal API credentials valid; class exists in schedule. | | |
| **Scenario Name** | **Step** | **Action** |
| **Main Flow – System receiving new booking information** | 1 | |  |  | | --- | --- | |  | Recreation Portal receives API request to book/cancel (including student ID and class ID). | |
| 2 | |  | | --- | | Portal validates request payload. | |
| 3 | |  |  |  |  | | --- | --- | --- | --- | |  | |  | | --- | |  |  |  | | --- | | Portal checks current availability (REQ\_1002). | | |
| 4 | |  |  |  | | --- | --- | --- | | |  | | --- | |  |  |  | | --- | | If booking: decrement slot count; if cancellation: increment slot count. | | |
| 5 | Portal returns confirmation or error (REQ\_1003 / REQ\_1004). |
| **Alternate Flow – Full Class** | 3a | If slots = 0, Portal returns “Class Full” error (REQ\_FB004). |
| **Alternate Flow – Invalid Class or Student ID** | 1a | Recreation Portal sends booking request with an unknown class Id or student Id. |
| 1b | Booking Service validates IDs and finds one or both to be invalid. |
| 1c | Booking Service returns error “Invalid student or class identifier.” |
| **Rules** | Booking requests must include valid student and class identifiers. | | |
| **Author** | Nicholas Thong Meng Shui | |

A diagram of a diagram

AI-generated content may be incorrect.

### 3.1.11 Integrate Wellness Data Across Systems

Functional Requirements for Integrate Wellness Data Across Systems

|  |  |
| --- | --- |
| Requirement ID | REQ\_1101 |
| Description | System shall periodically pull updated data from the Recreation and Health Portal APIs. |
| Author | Nicholas Thong Meng Shui |

|  |  |
| --- | --- |
| Requirement ID | REQ\_1101 |
| Description | System shall merge and normalize pulled data into the central wellness store. |
| Author | Nicholas Thong Meng Shui |

|  |  |
| --- | --- |
| Requirement ID | REQ\_1101 |
| Description | System shall log each synchronization event and any errors encountered. |
| Author | Nicholas Thong Meng Shui |

|  |  |
| --- | --- |
| Requirement ID | REQ\_1101 |
| Description | System shall notify administrators if a sync fails repeatedly. |
| Author | Nicholas Thong Meng Shui |

Use Case Table

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID** | UC\_011 | | |
| **Feature** | |  | | --- | |  |  |  | | --- | | Integrate Wellness Data Across Systems | | | |
| **Purpose** | Automatically synchronize wellness data between external portals and the platform. | | |
| **Actor** | |  | | --- | |  |  |  | | --- | | System (Automated) | | | |
| **Trigger** | |  | | --- | |  |  |  | | --- | | Scheduled job or data-change event. | | | |
| **Precondition** | |  | | --- | |  |  |  | | --- | | API credentials valid; central data store available. | | | |
| **Scenario Name** | **Step** | **Action** |
| **Main Flow – User scheduled an appointment** | 1 | |  |  | | --- | --- | |  | Scheduler triggers sync job (daily or event-driven). | |
| 2 | |  | | --- | | System calls Recreation Portal API to fetch data (REQ\_1101).. | |
| 3 | |  |  |  |  | | --- | --- | --- | --- | |  | |  | | --- | |  |  |  | | --- | | System calls Health Portal API to fetch data (REQ\_1101). | | |
| 4 | |  |  |  | | --- | --- | --- | | |  | | --- | |  |  |  | | --- | | System normalizes and merges datasets (REQ\_1102). | | |
| 5 | System writes merged data to central store and logs event (REQ\_1103). |
| 6 | If errors > threshold, system sends alert to admin (REQ\_1104). |
| **Alternate Flow – Partial Failure** | 3a | If one API fails, System continues with other, logs error, and sends partial success. |
| **Alternate Flow – Partial Data Schema Mismatch** | 4a | Integration Service receives data with unexpected format from one portal. |
| 4b | Service skips malformed records and logs a warning for each. |
| 4c | Service continues merging the rest of the data and records the partial‐success event. |
| **Rules** | Sync jobs shall not overlap; only one sync at a time. | | |
| **Author** | Nicholas Thong Meng Shui | |

A diagram of a software project

AI-generated content may be incorrect.

## 3.2 Performance Requirements

The performance for the Campus Wellness Portal are as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | **Description** | **Priority** | **Author** |
| REQ\_P001 | The average response time of Wellness Portal functions shall be less than 2 seconds. | High | Muhammad Anas |

## 3.3 Usability Requirements

## 3.4 Interface Requirements

This section covers the external interface requirements of the Campus Wellness Portal, separated by the features of the system.

### 3.4.1 Login Interface

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | REQ\_IO101 | **Version** | 1.0 |
| **Item** | Student ID (input) | | |
| **Description** | Input student ID on the portal login | | |
| **Purpose** | To allow the student to log into the portal | | |
| **Format** | String/Text | **Valid Range** | Not applicable |
| **Related I/O** | None | | |
| **Author** | Muhammad Anas | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | REQ\_IO102 | **Version** | 1.0 |
| **Item** | Student password (input) | | |
| **Description** | Input student password on the portal login | | |
| **Purpose** | To allow the student to log into the portal | | |
| **Format** | String/Text | **Valid Range** | Not applicable |
| **Related I/O** | None | | |
| **Author** | Muhammad Anas | | |

### 3.4.2 Health Centre Appointment Interface

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | REQ\_IO103 | **Version** | 1.0 |
| **Item** | ‘Schedule Appointment’ button | | |
| **Description** | Button to schedule an appointment | | |
| **Purpose** | To allow the student to schedule an appointment | | |
| **Format** | Button | **Valid Range** | Not applicable |
| **Related I/O** | None | | |
| **Author** | Muhammad Anas | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | REQ\_IO104 | **Version** | 1.0 |
| **Item** | ‘Cancel Appointment’ button | | |
| **Description** | Button to cancel a scheduled appointment | | |
| **Purpose** | To allow the student to cancel a scheduled appointment | | |
| **Format** | Button | **Valid Range** | Not applicable |
| **Related I/O** | None | | |
| **Author** | Muhammad Anas | | |

## 3.5 Logical Database Requirements

## Design Constraints

## 3.7 Software System Attributes

This section specifies the attributes that are expected from the Wellness Portal and its implementations.

### 3.7.1 Security

The security requirements for the Wellness Portal are as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | **Description** | **Priority** | **Author** |
| REQ\_Q001 | The Wellness Portal may only be accessed by students of the university. | High | Muhammad Anas |

## 3.8 Supporting Information

# 4.0 Verification

# 5.0 Appendices