**System Requirements Specifications (SRS)**

for

**Campus Event Check-in System with Student ID and Payment Integration**

**Part 1**

**Tutorial Section: TT1L**

**Group No.: Group 6**

|  |  |  |
| --- | --- | --- |
| HAZIQ IZZUDDIN BIN AHMAD TARMIDZI | 1211112293 |  |
| WAN AMIRUL AMIR BIN WAN ROMZI | 1211112289 |  |
| MUHAMMAD ARIF BIN ABDUL JALEEL | 1231303334 |  |
| LIM XIAO QI | 1211108212 |  |

**Date:** 23 May 2025

# **Contents**

[**Contents** 2](#_Toc198769267)

[**Revisions** 4](#_Toc198769268)

[**1** **Introduction** 5](#_Toc198769269)

[**1.1** **Purpose** 5](#_Toc198769270)

[**1.2** **Scope** 6](#_Toc198769271)

[**1.3** **Product overview** 7](#_Toc198769272)

[**1.3.1** **Product perspective** 7](#_Toc198769273)

[**1.3.2** **Product functions** 8](#_Toc198769274)

[**1.3.3** **User characteristics** 10](#_Toc198769275)

[**1.3.4** **Limitations** 11](#_Toc198769276)

[**1.4** **Definitions** 12](#_Toc198769277)

[**2** **References** 13](#_Toc198769278)

[**3** **Requirements** 14](#_Toc198769279)

[**3.1** **Functions** 14](#_Toc198769280)

[**3.1.1** **Student** 14](#_Toc198769281)

[**3.1.1.1** **Student Check-in** 14](#_Toc198769282)

[3.1.1.2 **Payment Processing** 15](#_Toc198769283)

[**3.1.2** **Event Organizer** 15](#_Toc198769284)

[**3.1.2.1** **Attendance Monitoring** 15](#_Toc198769285)

[**3.1.2.2** **Event Monitoring** 17](#_Toc198769286)

[**3.1.2.3** **Event Management** 18](#_Toc198769287)

[**3.1.3** **Administrator** 19](#_Toc198769288)

[**3.1.3.1** **Reporting** 19](#_Toc198769289)

[**3.1.3.2** **User Management** 20](#_Toc198769290)

[**3.1.4** **Summary Table** 20](#_Toc198769291)

[**3.2** **Performance requirements** 21](#_Toc198769292)

[**3.3** **Usability requirements** 21](#_Toc198769293)

[**3.4** **Interface requirements** 21](#_Toc198769294)

[**3.5** **Logical database requirements** 21](#_Toc198769295)

[**3.6** **Design constraints** 21](#_Toc198769296)

[**3.7** **Software system attributes** 21](#_Toc198769297)

[**3.8** **Supporting information** 21](#_Toc198769298)

[**4** **Verification** 22](#_Toc198769299)

[**4.1** **Verification Approach** 22](#_Toc198769300)

[**4.2** **Verification Criteria** 22](#_Toc198769301)

[**5** **Appendices** 23](#_Toc198769302)

[**5.1** **Assumptions and dependencies** 23](#_Toc198769303)

[**5.2** **Acronyms and abbreviations** 23](#_Toc198769304)

[**5.3** **Glossary** 23](#_Toc198769305)

# **Revisions**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| SRS Part 1 | Haziq Izzuddin,  Wan Amirul Amir,  Muhaammad Arif,  Lim Xiao Qi | Part 1 | 23/05/2025 |

# **Introduction**

* 1. **Purpose**

The purpose of the Campus Event Check-in System is to streamline the process of student event registration and attendance by integrating with the university’s Student Information Database and Payment Gateway. The system aims to automate check-ins, support secure payments, and provide real-time attendance tracking, improving convenience for students and efficiency for event organizers.

* 1. **Scope**

The Campus Event Check-in System will be used by students and event organizers within the university to manage event registration, attendance, and payments in a streamlined and integrated manner.

The system provides core functionalities such as student check-in using Student ID, event creation, payment processing, and real-time attendance tracking. It will integrate with the university’s Student Information Database for authentication and the university's Payment Gateway for processing transactions.

**The benefits of using this system include:**

* Simplified event check-in process for students
* Accurate and real-time attendance tracking
* Seamless payment experience for paid events
* Improved management tools for organizers
* Enhanced data reporting and event monitoring

**The objectives of the system are to:**

* Enable students to check in to events using their Student ID
* Facilitate secure, cashless payments for event participation
* Allow event organizers to create and manage events
* Track and report attendance data automatically
* Provide admin tools for user access and system monitoring

**The goals of the system are to:**

* Improve the efficiency and accuracy of event attendance tracking
* Minimize administrative overhead for event management
* Enhance the user experience for both students and organizers
  1. **Product overview**
     1. **Product perspective**

The Campus Event Check-in System is a web-based application that operates through an internet connection. When users access the system, the server authenticates their identity using the University Student Database. Based on their role as a student, event organizer, or admin officer, users are given access to specific features. The system also connects to the University Payment Gateway to process event-related transactions. All data is fetched and managed through the web server and presented to the user in real time.

A diagram of a college event

AI-generated content may be incorrect.

**1.3.1.1 Context Diagram**

* + 1. **Product functions**

The Campus Event Check-in System provides the following primary functions:

* **Student Check-in:** Students can check in to campus events by scanning or entering their Student ID. This verifies their attendance in real time and updates the attendance records automatically.
* **Payment Processing:** For events that require fees, students can make payments through the integrated payment gateway. The system supports both online and on-site payment options.
* **Attendance Monitoring:** Event organizers can view live attendance statistics and generate reports to monitor event participation and trends.
* **Event Management:** Administrative staff and organizers can create, edit, and manage event details, including scheduling, ticketing, and attendance rules.
* **Reporting:** The system generates detailed attendance and payment reports, helping organizers and administrators with financial reconciliation and event analysis.
* **User Management:** The system supports different user roles with appropriate access levels, including students, event organizers, and administrators.
* **Attendance Report (event organizer)**
* **Payment Report (event organizer)**
* **Activity logs (admin)**
* **Give Feedback (student)**
* **View feedback (event organizer)**
* **View ongoing events (student)**

A diagram of a diagram

AI-generated content may be incorrect.

**1.3.2.1 Use Case Diagram**

* + 1. **User characteristics**

The Campus Event Check-in System is designed to serve several distinct user groups, each with specific characteristics and needs:

* **Students:**  
  The primary users of the system, students vary widely in technical expertise but generally possess basic digital literacy. They require a straightforward, intuitive interface to quickly check in to events and make payments using their Student ID. The system must be easy to use on multiple devices, including smartphones and campus kiosks, to accommodate diverse user preferences.
* **Event Organizers:**  
  These users are responsible for creating and managing events, monitoring attendance, and handling ticketing. They typically have moderate technical skills and need access to real-time data and reporting features. Their tasks involve scheduling events, verifying attendance, and overseeing payment statuses, so the system should provide efficient management tools with clear dashboards.
* **Administrative Staff:**  
  Administrative users oversee the overall operation and maintenance of the system. They manage user roles, system configurations, and ensure data integrity. These users are expected to have a higher level of technical proficiency and require access to advanced administrative functions and security controls.

By understanding these user groups and their characteristics, the system can be designed to meet their specific usability needs, ensuring a positive user experience for all stakeholders.

* + 1. **Limitations**

The Campus Event Check-in System has several limitations that define its operational boundaries:

* The system is designed exclusively for campus events that are officially approved and managed by the university. It does not support external or off-campus events.
* It depends on the continuous availability and proper functioning of the university’s student information system for authenticating Student IDs. Any downtime or issues with this external system will impact the check-in process.
* Payment processing relies entirely on the university’s integrated payment gateway. The system does not support third-party or external payment methods outside this gateway.
* The system is limited to supporting users who have basic to moderate technical skills. It assumes students and staff will use standard web-enabled devices for access; specialized hardware or offline functionality is not provided.
* The platform must comply with university policies regarding data privacy and security, which may impose restrictions on data sharing and retention.

These limitations should be considered when deploying and maintaining the system to ensure realistic expectations and proper resource allocation.

* 1. **Definitions**
* **Student ID:** A unique identifier assigned to each student by the university for authentication and identification purposes.
* **Check-in:** The process by which a student verifies their attendance at a campus event through the system.
* **Payment Gateway:** A secure external service used to process financial transactions related to event fees.
* **Event Organizer:** A university staff member or authorized student responsible for creating, managing, and overseeing campus events.
* **Administrative Staff:** Personnel responsible for maintaining the system, managing user roles, and ensuring smooth operation.
* **Attendance Report:** A document or data set generated by the system that details the number of attendees and their check-in status for an event.
* **Real-time:** Data or actions that are processed immediately as they occur, with minimal delay.
* **Middleware:** Software that connects different systems, enabling them to communicate and exchange data seamlessly.

# **References**

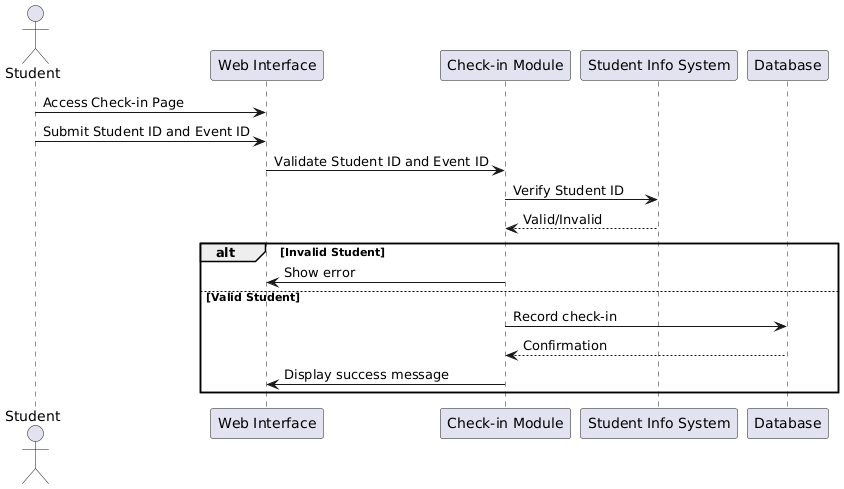
* IEEE. (2018). *ISO/IEC/IEEE 29148:2018 Systems and software engineering—Life cycle processes—Requirements engineering*. <https://www.iso.org/standard/72089.html>
* Pohl, K. (2010). *Requirements Engineering: Fundamentals, Principles, and Techniques*. Springer.

# **Requirements**

* 1. **Functions**
     1. **Student**

Students are the primary users who attend events, check in, and make payments when necessary.

* + - 1. **Student Check-in**
* **Function:** Allows students to check in to an event using their Student ID.
* **Explanation:** The system verifies the student’s identity and records their attendance.
* **Diagram:** Student Check-in



* + - 1. **Payment Processing**
* **Function:** Enables students to make payments for paid events.
* **Explanation:** The system integrates with a secure payment gateway to process transactions and record them in the database.
* **Diagram:** Payment Processing

A diagram of a payment process

AI-generated content may be incorrect.

* + 1. **Event Organizer**

Event Organizers are responsible for creating events, monitoring attendance, and managing event logistics.

* + - 1. **Attendance Monitoring**

**Function**: Allows organizers to view live attendance and check-in statistics.

**Explanation**: The system provides real-time data on who has checked in, helping organizers monitor event participation.

**Diagram**: Attendance Monitoring

A diagram of a data flow

AI-generated content may be incorrect.

* + - 1. **Event Monitoring**

**Function**: Gives organizers access to live event summaries, including attendance and payment updates.

**Explanation**: A dashboard view shows key metrics and allows for timely interventions if needed (example: capacity limits).

**Diagram**: Event Monitoring

A diagram of a event

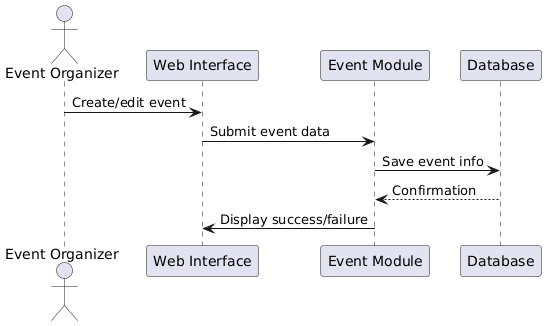
AI-generated content may be incorrect.

* + - 1. **Event Management**

**Function**: Lets organizers create, update, and delete event details such as title, date, location, and ticketing options.

**Explanation**: Ensures events are accurately listed and managed in the system.

**Diagram**: Event Management



* + 1. **Administrator**

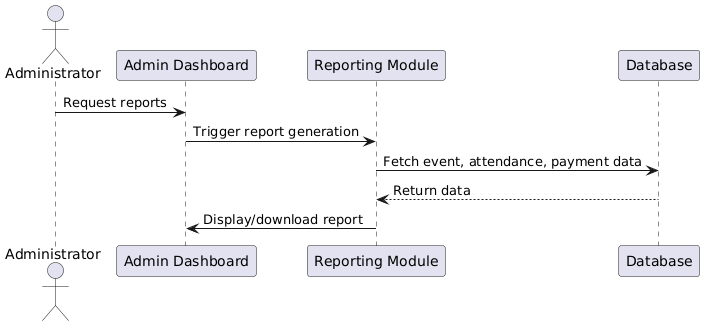
Administrators manage the system backend, reporting, and user roles

* + - 1. **Reporting**

**Function**: Generates reports on attendance, payments, and event success metrics.

**Explanation**: Helps administrators perform audits, analyze trends, and support strategic planning.

**Diagram**: Reporting

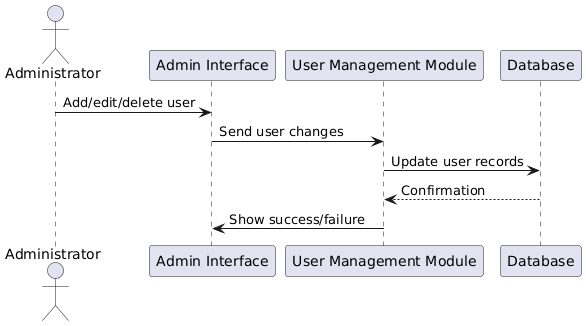


* + - 1. **User Management**

**Function**: Administers system users and their roles (e.g., assigning permissions, creating new organizers).

**Explanation**: Ensures users have appropriate access and that user data is accurate and secure.

**Diagram**: User Management



* + 1. **Summary Table**

|  |  |  |  |
| --- | --- | --- | --- |
| Actor | Function | Description | Diagram |
| Student | Student Check-in | Check into events with Student ID | Student Check-in |
| Payment Processing | Make payments for paid events | Payment Processing |
| Event Organizer | Attendance Monitoring | View real-time check-ins | Attendance Monitoring |
| Event Monitoring | Monitor overall event performance | Event Monitoring |
| Event Management | Create and manage event details | Event Management |
| Administrator | Reporting | Generate detailed reports | Reporting |
| User Management | Manage user roles and access | User Management |

* 1. **Performance requirements**
  2. **Usability requirements**
  3. **Interface requirements**
  4. **Logical database requirements**
  5. **Design constraints**
  6. **Software system attributes**
  7. **Supporting information**

# **Verification**

* 1. **Verification Approach**
* **How:** The system will be verified using functional testing, unit testing, and system integration testing to ensure it performs as required.
* **Who:** The product development team and the quality assurance (QA) department will be responsible for carrying out the verification activities.
* **When:** Verification will take place at key points during the development cycle, such as after the completion of each sprint.
* **Where:** All verification activities will be conducted within the QA testing environment.
  1. **Verification Criteria**

The system will be considered successfully verified when the following criteria are met:

* User authentication correctly allows access for valid Student IDs and denies invalid attempts.
* Event check-ins are accurately recorded and reflected in real-time without data loss or errors.
* Payment transactions through the integrated payment gateway are processed securely and successfully.
* Reports generated by the system accurately represent attendance and payment data.
* System response times meet the required performance standards, such as responding within 2 seconds under normal load.
* The user interface allows users to complete primary tasks efficiently and intuitively, meeting usability requirements.

# **Appendices**

* 1. **Assumptions and dependencies**
* The system assumes that all students and event organizers have valid accounts in the university's Student Information Database.
* The system depends on the continuous availability of the university’s authentication and payment gateway services.
* Internet access is required for all system functionalities to be accessible.
* Event organizers are assumed to have proper authorization to create and manage events.
* The system assumes that existing campus infrastructure (servers, networks) can support additional load introduced by this platform.
  1. **Acronyms and abbreviations**
* **SRS**: Software Requirements Specification
* **FCI**: Faculty of Computing & Informatics
* **ID**: Identification
* **API**: Application Programming Interface
* **UI**: User Interface
* **DB**: Database
* **HTTPS**: Hypertext Transfer Protocol Secure
  1. **Glossary**
* **Check-in**: The process of registering presence at an event using the system.
* **Student ID**: A unique identifier assigned to each student by the university.
* **Payment Gateway**: A service that authorizes and processes payments securely.
* **Event Organizer**: A staff or student responsible for managing campus events.
* **Authentication**: The process of verifying user identity before granting access.