**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](#_Toc198689838)

[**Revisions** 3](#_Toc198689839)

[**1** **Introduction** 4](#_Toc198689840)

[**1.1** **Purpose** 4](#_Toc198689841)

[**1.2** **Scope** 5](#_Toc198689842)

[**1.3** **Product overview** 6](#_Toc198689843)

[**1.3.1** **Product perspective** 6](#_Toc198689844)

[**1.3.2** **Product functions** 6](#_Toc198689845)

[**1.3.3** **User characteristics** 6](#_Toc198689846)

[**1.3.4** **Limitations** 6](#_Toc198689847)

[**1.4** **Definitions** 6](#_Toc198689848)

[**2** **References** 7](#_Toc198689849)

[**3** **Requirements** 8](#_Toc198689850)

[**3.1** **Functions** 8](#_Toc198689851)

[**3.2** **Performance requirements** 8](#_Toc198689852)

[**3.3** **Usability requirements** 8](#_Toc198689853)

[**3.4** **Interface requirements** 8](#_Toc198689854)

[**3.5** **Logical database requirements** 8](#_Toc198689855)

[**3.6** **Design constraints** 8](#_Toc198689856)

[**3.7** **Software system attributes** 8](#_Toc198689857)

[**3.8** **Supporting information** 8](#_Toc198689858)

[**4** **Verification** 9](#_Toc198689859)

[**5** **Appendices** 10](#_Toc198689860)

[**5.1** **Assumptions and dependencies** 10](#_Toc198689861)

[**5.2** **Acronyms and abbreviations** 10](#_Toc198689862)

# **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 develop a comprehensive digital platform that enables students to conveniently and securely check in to campus events using their unique Student ID. This system is designed to integrate seamlessly with the university’s existing student identification database and payment gateway, ensuring that attendance verification and payment processing are handled efficiently and accurately. By automating these processes, the system aims to reduce administrative overhead, minimize errors in attendance tracking, and improve the overall management of campus events.

Furthermore, the system will provide real-time data and reporting capabilities to event organizers and university administrators, allowing them to monitor attendance and financial transactions easily. This will enhance transparency and support better decision-making in event planning and resource allocation. Ultimately, the Campus Event Check-in System seeks to create a smooth, user-friendly experience for students while supporting the university’s goals of efficient event management and financial accountability.

* 1. **Scope**

The Campus Event Check-in System is designed to manage and streamline the entire process of event attendance and payment on a university campus. The system will allow students to check in to various campus events using their unique Student ID, which will be authenticated against the university’s existing student information system. This ensures that only authorized students can register their attendance at events.

In addition to attendance tracking, the system will facilitate payment processing for events that require fees. Payments can be made on-site or online through integration with the university’s secure payment gateway, allowing for seamless financial transactions. Event organizers and administrative staff will be able to monitor real-time attendance data and payment statuses through a dedicated dashboard, improving the accuracy of reports and simplifying event management tasks.

The system’s scope includes support for different types of events held on campus, ranging from small student club activities to large university-wide functions. It will generate attendance reports and payment summaries to help organizers and university departments with event planning and financial reconciliation. However, the system is limited to campus events approved and managed by the university and does not extend to events hosted outside of this context.

Furthermore, the system is designed to operate within the constraints of the university’s IT infrastructure, relying on the availability of the student database and payment gateway. It will not support third-party payment systems or external user authentication methods. The focus is on improving operational efficiency, data accuracy, and user convenience within the university event management environment.

* 1. **Product overview**

The Campus Event Check-in System is a centralized software platform designed to facilitate the management of event attendance and payment on a university campus. It connects students, event organizers, and administrative staff through a single interface that integrates with the university’s student information system and payment gateway.

Students use the system to check in to events quickly by scanning or entering their Student ID, ensuring secure and accurate verification of their attendance. For events that require payment, the system enables seamless transaction processing through the university’s existing payment infrastructure, supporting both on-site and online payments.

Event organizers benefit from real-time attendance tracking and reporting tools, allowing them to monitor event participation and manage ticketing efficiently. Administrative staff have access to dashboards for overseeing event management, generating detailed attendance and financial reports, and maintaining system configurations.

By bringing these functions together, the system improves operational efficiency, enhances data accuracy, and provides a better experience for all users involved in campus events.

* + 1. **Product perspective**

The Campus Event Check-in System functions as a middleware platform that integrates with the university’s existing systems. It connects to the student information system to authenticate users via their Student IDs and to the payment gateway to process event-related payments securely.

This integration ensures that student attendance and payment data remain consistent across systems and reduces the need for manual data entry. The system is designed to operate within the university’s IT infrastructure, relying on these external services for key functionalities.

Users will access the system primarily through a web-based interface, supporting multiple roles such as students, event organizers, and administrators. This approach allows the system to fit smoothly into the existing university environment without requiring significant changes to other systems.

A diagram of a college event

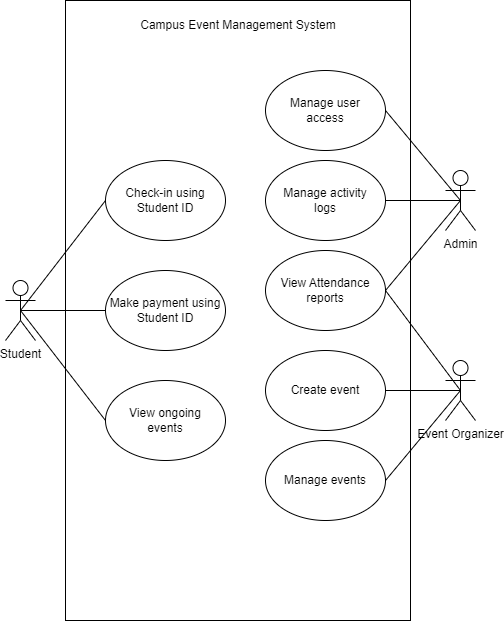
AI-generated content may be incorrect.

**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.

****

**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**

# **Requirements**

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

# **Verification**

# **Appendices**

* 1. **Assumptions and dependencies**
  2. **Acronyms and abbreviations**