**System Documentation**

for

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

**Task 4**

**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:** | 1. **MAY 2025** |

# **Elicitation Execution**

Execute: Conduct elicitation sessions using your planned techniques.

1. **Techniques**
   1. **Techniques 1: Interview**

Interviewer: Melvin Woo

Roles: Student

|  |  |  |  |
| --- | --- | --- | --- |
| **Interview Questions** | **Feature** | **Insight Gained** | **Kano Category** |
| 1. How important is it for you to log in using your Student ID instead of creating a new account? | Student Authentication (SSO via SIS) | Students prefer official university login for ease and security. | Must-Have |
| 2. Would you prefer scanning a QR code to check in at events over manual sign-ins? | QR Code-Based Check-In | QR codes are favored for speed and convenience. | Performance |
| 3. Is receiving a confirmation email after registration and payment important to you? | Email Notification & Payment Receipt | Confirmation builds trust and ensures transaction transparency. | Must-Have |
| 4. How helpful would it be to get automatic reminders for events you've signed up for? | Calendar Sync / Notifications | Students often forget events; reminders improve attendance. | Delighter |
| 5. Would you like your personal details to autofill during event registration? | SIS Integration for Autofill | Avoids repetitive entry, improves user experience. | Must-Have |
| 6. Would you find it useful to store and download participation certificates in the system? | Digital Certificate Vault | Certificates are useful for resumes and portfolios. | Delighter |
| 7. Do you prefer paying for events through UPI, credit/debit card, or e-wallets? | Integrated Payment Gateway | Mobile-friendly payments are essential for students. | Performance |
| 8. Would it be useful to view your past event attendance in your profile? | Event History Log | Tracking participation helps students manage involvement. | Delighter |
| 9. How would you feel if check-in failed due to internet issues at the venue? | Offline Check-In Mode | Offline functionality is expected for reliability. | Must-Have |
| 10. Would personalize event suggestions based on your interests improve your experience? | Offline Check-In Mode | Not expected but would increase engagement. | Delighter |

**Outcomes:**

|  |  |  |
| --- | --- | --- |
| **Feature** | **Kano Category** | **Justification** |
| Student ID Login (SSO) | Must-Have | Students expect seamless login using university credentials; builds trust. |
| QR Code-Based Check-In | Performance | Significantly speeds up the entry process; preferred over manual sign-in. |
| Email Confirmation & Receipts | Must-Have | Essential for user confidence in payment and registration tracking. |
| Calendar Sync / Event Reminders | Delighter | Unexpected but appreciated; improves attendance and time management. |
| Autofill Student Details via SIS | Must-Have | Repetition is frustrating; students expect pre-filled forms for speed. |
| Digital Certificate Vault | Delighter | Storing downloadable proof of participation adds long-term value. |
| Integrated Payment Gateway | Performance | Students value flexible and mobile payment methods like UPI and e-wallets. |
| Attendance History Log | Delighter | Helps students track involvement; adds transparency and reflection value. |
| Offline Mode for Check-In | Must-Have | Reliability is expected regardless of internet conditions at the event site. |
| Personalized Event Recommendations | Delighter | Enhances engagement; not required, but positively influences event discovery. |

* 1. **Techniques 2: Observation**

**Article 01**

Authors: Aditi Chaturvedi, Krishna Sharma, Akshat Dua, Aastha Gupta

Publisher: IJRASET Journal for Research in Applied Science and Engineering Technology

Link:<https://www.ijraset.com/best-journal/cuevents-a-comprehensive-event-management-system-for-university>

**Objective:**

The primary goal of this study is to design and implement CU-Events, a centralized event management platform tailored for universities. This system aims to address common challenges in campus event management such as inefficient manual registration processes, poor attendance tracking, and lack of centralized data for event organizers. CU-Events seeks to streamline event creation, participant registration, and real-time check-in using student IDs, all while incorporating features to handle payment integration for paid events.

**Methodology:**

The system is developed based on a modular, three-tier architecture including the presentation layer (front-end), business logic layer (back-end), and data layer (database). The front-end utilizes React.js for creating an interactive user interface that supports easy navigation for both students and organizers. The back-end is built on Node.js and Express.js, ensuring efficient processing of requests and robust API services. A MySQL database manages event details, user profiles, attendance, and payment records. The study involved iterative development with continuous feedback from students and campus event managers to refine usability and performance.

**Findings:**

CU-Events demonstrated marked improvements in administrative efficiency by automating registration and attendance tracking. The integration of student ID scanning reduced entry bottlenecks during events. Payment integration allowed for smooth transactions for paid events without the need for separate payment portals. User surveys indicated increased satisfaction due to the system's ease of use and quick access to event information. The system also facilitated detailed analytics for event attendance and revenue, assisting campus officials in making data-driven decisions for future event planning.

**Article 02**

Authors: Sun Yang, Lixia Wen

Publisher: Open Journal of Social Sciences

Link: <https://www.scirp.org/journal/paperinformation?paperid=101114>

**Objective:**

This research aims to develop a unified virtual payment system tailored for campus environments that consolidates various payment services, including event fee collection, dining, transportation, and other campus-related payments. The objective is to improve campus financial operations by creating an intelligent, secure, and convenient payment platform that can integrate seamlessly with student ID cards and mobile applications.

**Methodology:**

The study employed a three-layer architectural design: an infrastructure layer to provide the foundational hardware and network environment; a core framework layer to handle payment processing, authentication, and security; and a service platform layer delivering user-facing payment applications. The implementation leverages microservices architecture deployed via Docker containers and Kubernetes for scalability. Spring Cloud supports service discovery and fault tolerance. The research included prototype development and simulated testing with mock payment transactions under high concurrency to evaluate system stability and response time.

**Findings:**

The virtual payment system effectively supports multiple payment methods, including QR codes, NFC, and direct integration with student ID cards. The system shows high scalability and fault tolerance, handling thousands of transactions concurrently without downtime. It also provides robust security measures such as encrypted communication and two-factor authentication. Campus administration reported a significant reduction in manual billing errors and improved financial tracking. Students benefited from a one-stop payment interface, reducing the need to carry multiple cards or cash, thereby enhancing convenience and safety.

**Article 03**

Authors: Ibad Ali, Bhushanwar, Atulbahi Vaghela, Khodifad, Dayaram Patil, Pandwal

Publisher: ResearchGate

Link:<https://www.researchgate.net/publication/389201749_The_Campus_Pay_Offline_Payment_System_Using_RFID_Technology_for_Campus_Transactions>

**Objective:**

This study focuses on the development of an offline campus payment system that leverages RFID-enabled student ID cards to facilitate fast, secure, and reliable transactions in areas with limited or unstable internet connectivity. The system targets campus facilities such as event venues, cafeterias, and bookstores, allowing students to make payments without the delays or disruptions caused by network failures.

**Methodology:**

The system architecture integrates RFID readers at transaction points and a local database that records payments temporarily when offline. Transactions are encrypted and stored securely, then synchronized with a centralized cloud database once internet connectivity is restored. The team developed a prototype system using Arduino microcontrollers, RFID tags embedded in student IDs, and a custom-built POS terminal. Field testing was conducted on a university campus to monitor transaction speed, error rates, and user acceptance under varying network conditions.

**Findings:**

Results showed the offline RFID system reduced transaction times significantly compared to traditional cash or card payments, especially in high-traffic areas during event check-ins. Users reported high satisfaction due to ease of use and reliability. The system maintained data integrity with no reported loss of transaction records after synchronization. Campus administrators highlighted improved operational continuity even during network outages, making the system ideal for remote or infrastructure-limited environments. The study concluded that RFID-based offline payments offer a scalable and cost-effective solution for campus financial transactions.

**Observation Technique Tables**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Article Title** | **Observation Type** | **Observation Context** | **Tools/Tech Used** | **Parameters Observed** | **Findings/Outcomes** |
| **CU-Events: A Comprehensive Event Management System for University** | System Usage & User Behavior | Observing system interaction during event registration and check-in phases using student ID and payment modules | React.js (UI), Node.js, Express.js, MySQL, Student ID Scanner | Registration time, check-in speed, user satisfaction, payment success rates | Reduced entry bottlenecks; user satisfaction increased; smooth payment experience; enabled event data analytics for planners |
| **Development of Unified Virtual Payment System in Campus Environment** | Simulated System Performance | Observing virtual payment platform under stress conditions with mock transactions | Spring Cloud, Docker, Kubernetes, Student ID/NFC integration | Transaction throughput, system response under load, authentication success rate | High scalability; supported high concurrency; secure transaction handling; improved financial tracking; users appreciated simplified digital payments |
| **The Campus Pay: Offline Payment System Using RFID Technology** | Field-Based Functional Test | Observing real-world transactions in offline conditions across campus (cafeterias, bookstores, events) | RFID Tags, Arduino, Local DB, POS Terminal, Sync to Cloud | Transaction time, data synchronization accuracy, network dependency, error rates | Reliable in low-connectivity zones; fast transaction times; no data loss post-sync; high user and admin satisfaction; ideal for infrastructure-limited areas |

**Key Observation**

|  |  |  |
| --- | --- | --- |
| **Aspect** | **Common Trend Across Articles** | **Kano Category** |
| **Student ID Integration** | All systems utilized student ID cards (with RFID or digital scan) for check-ins and payments, streamlining access and increasing accuracy. | Must-Be |
| **Check-in Speed & Efficiency** | Real-time or offline systems reduced bottlenecks at event entries and service points, enhancing the overall user experience. | One-Dimensional |
| **Payment Integration** | Systems offered digital or offline payment integration, reducing dependency on cash and external systems. | Must-Be |
| **Offline Functionality (RFID Systems)** | RFID-based systems continued to function during internet outages, syncing data when back online—crucial in unstable network environments. | Attractive |
| **System Usability (UI/UX Design)** | User interfaces were designed for ease of use; users expressed satisfaction with system navigation and functionality. | One-Dimensional |
| **Analytics & Reporting Tools** | Event organizers and admins used analytics for attendance tracking and revenue analysis, improving data-driven decision-making. | Attractive |
| **Multi-Payment Options (NFC, QR, ID)** | Support for various payment modes (e.g., QR code, NFC, student ID) gave users flexibility and choice. | Attractive |
| **Scalability & Performance** | Systems handled high transaction volumes smoothly due to modular and scalable architectures (e.g., microservices, cloud computing). | One-Dimensional |
| **Security & Data Integrity** | All systems used encryption and secure authentication; no data loss was reported, even during offline syncing. | Must-Be |
| **User Satisfaction & Adoption** | Students and admins showed high satisfaction, especially with simplified processes and faster transactions, leading to higher system adoption. | One-Dimensional |

* 1. **Techniques 3: Brainstorming**

**Questions:**

|  |  |
| --- | --- |
| **Prompt (Question)** | **Goal** |
| What are the problems with current campus event check-in methods? | Identify pain points and justify the need for a new system. |
| How can student IDs be used to simplify event check-ins? | Explore methods to automate and speed up the check-in process. |
| What types of payment should be integrated? | Define payment flexibility and cater to user preferences. |
| Should the system operate offline? | Determine the necessity of offline capabilities. |
| What data is needed at check-in and how can privacy be ensured? | Define data requirements and ensure privacy compliance. |
| What benefits can this system provide to event organizers? | Identify administrative advantages and added value. |
| What kind of user interface would work best for students and staff? | Improve usability for both user groups. |
| How should users receive payment confirmations? | Clarify how payment status and confirmations are communicated. |
| What platforms should the system support? | Define technical scope and accessibility across devices. |
| How can the system be tested before launch? | Plan for reliability testing and user feedback collection. |

**Outcomes:**

|  |  |  |
| --- | --- | --- |
| **Outcome (Idea)** | **Kano Category** | **Notes** |
| Replace manual check-ins with automated scanning via student IDs. | Must-Be | Manual processes are slow, error-prone, and frustrating for users. |
| Use RFID/QR/NFC-enabled student IDs for real-time scanning and attendance tracking. | One-Dimensional | Links attendance to student profiles and speeds up entry. |
| Support debit/credit cards, mobile wallets, and prepaid campus accounts. | One-Dimensional | Multiple options improve convenience and centralized tracking. |
| Use offline-capable RFID readers with local storage and sync when online. | Attractive | Essential for events with poor network connectivity. |
| Collect minimal data with encryption and access control. | Must-Be | Privacy compliance is critical; minimize and protect data. |
| Provide an admin dashboard for attendance, payments, and analytics. | Attractive | Facilitates data-driven decisions and reduces manual effort. |
| Mobile-first UI for students; web dashboard for staff. | One-Dimensional | Intuitive design boosts adoption and reduces training. |
| Auto-generate digital receipts and QR check-in tickets sent by email/app. | Must-Be | Confirmation reassures users and speeds entry validation. |
| Support web (admin), mobile app (students), and optional kiosks. | Attractive | Cross-platform increases accessibility and convenience. |
| Pilot test at campus events; stress test concurrency; collect user feedback. | Must-Be | Identifies bugs and improves UX before full rollout. |