Revisions

| Version | Primary Author(s) | Description of Version | Date Completed |
| --- | --- | --- | --- |
| 1.0 | Hesham | Categories added, drafted up examples for objects | 18/04/2025 |
| 1.1 | Nickleirsch | Drafted sources of requirements | 19/04/2025 |
| 1.2 | Lim Xin Yee | Drafted development context objects | 21/04/2025 |
| 1.3 | Nickleirsch | Explanation for material context objects | 22/04/2025 |
| 1.4 | Nickleirsch | Explanation for immaterial context objects | 22/04/2025 |
| 1.5 | Danesh Veran | Drafted more immaterial and development context objects | 23/04/2025 |
| 1.6 | Danesh Veran | Explanation for development context objects | 23/04/2025 |
| 1.7 | Hesham | Added a source of requierement | 24/04/2025 |
| 1.8 | Lim Xin Yee | Added a source of requirements | 24/04/2025 |
| 1.9 | Hesham | Fixed formatting | 24/04/2025 |

Context objects are elements in and around the system that influence its design and behaviour. We can divide them into two main categories – material and immaterial.

**Material Context Objects:**

|  |  |  |
| --- | --- | --- |
| **Object** | **Sources** | **Explanation** |
| Students | Project Guidelines (Document) | expected to be the primary users of the system; access academic data, receive notifications, view attendance, etc. |
| Lecturers | Project Guidelines (Document) | educators who will enter academic records, view student data, communicate updates, etc. |
| Administrators | Project Guidelines (Document) | manage system access, maintain academic and financial records. |
| Parents | Project Guidelines (Document) | receive academic and financial updates via SMS and view performance summaries. |
| University Database Servers | Existing System | store academic records, attendance, billing info, and user profiles. |

1. Campus: physical environment where the system is used — affects network access and system availability.
2. Network Infrastructure: enables system access across campus (Wi-Fi, LAN, internet connectivity, etc).

**Immaterial Context Objects:**

|  |  |  |
| --- | --- | --- |
| **Object** | **Sources** | **Explanation** |
| Academic Processes | Existing System | workflows like course registration, grading, and attendance tracking. |
| Communication Protocols | Technical Specification (?) | rules that define how messages are sent and received (e.g., email, SMS format, response handling). |
| Administrative Processes | Existing System | includes billing, scheduling, reporting, and system maintenance routines. |
| SMS/Notification Services | Technical Specification (?) | third-party or internal services used to deliver timely messages (alerts, reminders) to users. |
| API Integrations | Technical Specification (?) | Interfaces that connect the portal with the campus management system, SMS gateway, etc. |
| Authentication and Access Control Policies | Technical Specification (?) | define how users securely log in (e.g., via single sign-on or OTP), session timeouts and password rules. |
| Notification Prioritization and Escalation Logic |  | logic flow that determines the urgency and format of alerts. For example, critical alerts (e.g., payment deadlines, academic probation) are sent via SMS + portal, while general reminders (e.g., campus events) are sent via portal only. |
| Localization and Language Settings |  | optional configurations to support multi-language capabilities and regional time/date formats, especially for institutions with international students or multi-lingual local populations. |

1. User Roles and Permissions: which define what each user (student, parent, etc.) can see or do within the portal.

**Development Context Objects:**

|  |  |  |
| --- | --- | --- |
| **Object** | **Sources** | **Explanation** |
| Technical Environment | Project Guidelines (Document) | The hardware, software, and platforms used to develop, test, and deploy the portal. |
| IT/Development Team |  | responsible for designing, coding, testing, and maintaining the portal (developers, system admins, and project managers). |
| Testing Tools & Methods |  | verifying the portal’s functionality, performance, and security (e.g., unit testing, automated testing, UAT). |
| Budget & Timeline Constraints |  | cost limits and deadlines that influence scope, development speed, and resource allocation. |
| Version Control System |  | managing code changes, track development history, and support collaboration among developers. |
| Deployment Strategy |  | how and when the portal is rolled out to users, including testing phases, rollback plans, and live release timelines. |
| Maintenance and Support Plan |  | outlines post-launch support, including updates, bug fixes and scheduled maintenance. |
| Security Policies and Protocols |  | rules ensuring secure coding, user authentication and data encryption. |
| Integration Testing and Staging Environments |  | dedicated testing systems for integrations (e.g., APIs, SMS Gateway) and ensuring system stability. |

1. Technical Environment: The hardware, software, and platforms used to develop, test, and deploy the portal.
2. IT/Development Team: responsible for designing, coding, testing, and maintaining the portal (developers, system admins, and project managers).
3. Testing Tools & Methods: verifying the portal’s functionality, performance, and security (e.g., unit testing, automated testing, UAT).
4. Budget & Timeline Constraints: cost limits and deadlines that influence scope, development speed, and resource allocation.
5. Version Control System: managing code changes, track development history, and support collaboration among developers.
6. Deployment Strategy: how and when the portal is rolled out to users, including testing phases, rollback plans, and live release timelines.
7. Maintenance and Support Plan: outlines post-launch support, including updates, bug fixes and scheduled maintenance.
8. Security Policies and Protocols: rules ensuring secure coding, user authentication and data encryption.
9. Integration Testing and Staging Environments: dedicated testing systems for integrations (e.g., APIs, SMS Gateway) and ensuring system stability.

**Sources of Requirements**

Stakeholders (students, parents, lecturers, admin) – understand what they need

University policies (rules)

System Project Guidelines

Current system documentation – look at what constraints the current system has