KUALA LUMPUR UNIVERSITY
OF SCIENCE AND TECHNOLOGY
吉隆坡科技大学

**CSC 1017-SYSTEM ANALYSIS & DESIGN**
**AUGUST 2026**

# KLUST shuffle Bus system Full project

| | |
|---|---|
| NAME | 1: Ismail Abdirisaq Abdiqani<br>2: MONIR SAAD BIN<br>3: Ibrahim Zakarie abdiwali<br>4: Mohamed Ahmed Yusuf |
| MATRIC NO. | 1:<br>2:<br>3:<br>4: |
| DATE | 10/10/2026 |

**TABLE OF CONTENTS**


1. Executive Summary & System Overview....1
2. System Actors....1
3. Functional Requirements....1
4. Non-Functional Requirements....1
5. System Scope....1
   - 5.1 In Scope....1
   - 5.2 Out of Scope....1
6. Constraints & Ethical Considerations....1
   - 6.1 GPS Tracking Privacy....1
   - 6.2 Accessibility....1
7. Requirement Elicitation Methods & Evidence....1
   - 7.1 Methods Applied....1
   - 7.2 Survey Instrument & Administration....1
   - 7.3 Survey Findings....1
   - 7.4 Honest Interpretation of Findings....1
   - 7.5 Design Decision: Seat Reservation vs. Removal....1
   - 7.6 Existing System Context (Document Review)....1
8. Use Case Diagram....1
9. Use Case Specifications....1
   - 9.1 UC-01: Board Bus....1
   - 9.2 UC-02: Trigger SOS/Delay Alert....1
   - 9.3 UC-03: Manage Routes & Analytics....1
10. Innovation & Advanced Features....1
    - 10.1 AI-Driven Demand Forecasting....1
    - 10.2 Dynamic Route Optimization....1
    - 10.3 Secondary Idea (brief)....1
11. Diagrams....1
12. User Interface Prototypes....1
13. Screen 1 — Student Mobile App....1
14. Screen 2 — Driver Tablet Dashboard....1
15. Screen 3 — Admin Web Dashboard....1
16. Design Summary....1
17. APPENDIX A: AI PROMPT EVIDENCE....1

17.1 A.1 AI Tool Used....1
17.2 A.2 Prompt 1: Initial UI Kit Design....1
17.3 A.3 Prompt 2: Improved UI Design and Color Consistency....1
17.4 A.4 Prompt 3: Final UI Corrections....1
17.5 A.5 Summary of AI-Assisted Design Process....1

# 1. Executive Summary & System Overview

The current KLUST University campus shuttle service relies on one vehicle, using a fixed-loop route. It has relied solely on manual processes. As the university expands the number of vehicles to be in operation from 8 – 10 units, these manual processes will create significant operational risks. This document includes the Checkpoint 1 Deliverables for the KLUST CampusShuttle System, a data-driven fleet management ecosystem that connects Students, Drivers and Fleet Administrators and resolves four key issues currently affecting the service.

- Zero Passenger Visibility — students do not know where buses are or how many seats remain available.
- Blind Fleet Management — administrators have no analytic tools to evaluate demand and routes.
- Manual Cash Payments — collecting money by hand at each stop increases boarding time, administrative costs and creates an additional layer of complexity.
- Lack of Emergency Alerts — there is no method for drivers to notify administration immediately of a problem.

# 2. System Actors

| Actor | Type | Role |
|---|---|---|
| Student | Primary | Can view live bus capacity/location, pay digitally, receive alerts. |
| Driver | Primary | Can trigger SOS alert/manifest/bus status update. |
| Fleet Administrator | Primary | Manage route/driver/fleet and analytics |
| Payment Gateway | Secondary (external) | Processes digital payments |
| GPS / Location Service | Secondary (external) | Provide Live GPS Location Data. |
| SMS Gateway | Secondary (external) | Send Student SMS Notifications/Payments Confirmed via SMS |

# 3. Functional Requirements

Each of the requirements listed has an example of how they are explained plainly as well as references to the issue they resolve (Section 1) or evidence from which they were elicited (Section 7).

| ID | Requirement | Explanation | Problem fixed |
|---|---|---|---|
| FR-01 | The System will display real-time bus position on a live map that is available to students immediately without them having to log-in or take any other action. | Students can check exactly where a shuttle is before walking to the stop, with no sign in required first. | Zero Passenger Visibility |
| FR-02 | System shall show live remaining seat capacity per bus. | Shows how full a bus is in real time, so students know if it's worth waiting for. | Zero Passenger Visibility |
| FR-03 | Students shall be able to pay digitally (e-wallet/card) when boarding the bus. | Replaces cash collection with a fast digital payment at the point of boarding, no advance reservation needed. | Manual Cash Payments |

| | | | |
|---|---|---|---|
| FR-04 | The system will produce a receipt or transaction document electronically as soon as it receives successful payment from the rider. | The receipt will provide evidence of payment to the student and also provides an electronic document for the administrator to keep a record of all transactions. This eliminates the need for manual records. | Manual Cash Payments |
| FR-05 | The administrator should have access to a dashboard showing how many people are riding during peak hours based on analytics | The administrator will be able to see the times when there is a surge in riders (so they know exactly what time to schedule the buses). | Blind Fleet Management |
| FR-06 | The system will create congestion/route demand documents based on where riders get off. | The system will show which stops are constantly overcrowded, so the administrator can adjust routes accordingly. | Blind Fleet Management |
| FR-07 | The system will log historical ridership information so administrators can analyze trends. | This allows administrators to use this past information to make decisions about adding additional fleets or changing bus schedules. | Blind Fleet Management |
| FR-08 | The driver will be able to click one button to send an SOS alert. | The SOS alert will allow drivers to quickly send a message about an emergency situation while avoiding having to navigate through the menu options on their device. | Lack of Emergency Alerts |
| FR-09 | As soon as the driver sends out an SOS alert, the administrator and any affected students will receive notification via SMS or push notifications sent directly into the mobile app. | Even though students may not be currently logged onto the application, both the administrator and the students will be notified immediately. | Lack of Emergency Alerts |
| FR-10 | Driver shall be able to log a delay reason (e.g. traffic, breakdown, accident). | Captures why a delay happened, feeding into admin's incident records and future planning. | Lack of Emergency Alerts |
| FR-11 | Administrators should have the capability to reallocate or re-route a bus based on an alert. | Allows administrators to take action with regard to a breakdown or emergency in relation to a bus, through re-routing another. | Lack of Emergency Alerts |
| FR-12 | Administrators should be able to view real time data on the location, speed and capacity of each active bus through a single fleet view dashboard. | Provides administrators with one screen to view the entire fleet rather than having to check each individual bus. | Blind Fleet Management |
| FR-13 | Administrators should be able to export ridership and revenue reports (for example; CSV/PDF) . | Enables administrators to create documentation for University reporting or budget planning, that is not manually entered. | Blind Fleet Management |
| FR-14 | Administrators should be able to schedule drivers and their shifts. | Provides centralization of the scheduling of drivers as opposed to using a separate manual roster. | Blind Fleet Management |

| | | | |
|---|---|---|---|
| FR-15 | Admin shall be able to configure and update fixed routes, stops, and timetables. | Lets admin adjust the service directly as the fleet expands, instead of manual notices. | Blind Fleet Management |
| FR-16 | Admin shall be able to review historical SOS/delay incident logs for safety and compliance audits. | Creates an auditable safety record admin can refer back to if an incident needs review. | Lack of Emergency Alerts |

# 4. Non-Functional Requirements

| Category | ID | Requirement | Explanation |
|---|---|---|---|
| Performance | NFR-01 | GPS location updates shall refresh on the student app within 5 seconds. | Keeps the live map accurate enough to actually be useful for deciding when to leave. |
| Performance | NFR-02 | SOS alerts shall reach the Admin dashboard within 3 seconds of trigger. | Fast enough that admin can respond to an emergency almost as soon as it happens. |
| Security | NFR-03 | All payment transactions shall be encrypted. | Protects student payment data in transit standard practice for digital payment. |
| Security | NFR-04 | Student location data shall not be exposed to other students (aggregated bus location only). | Prevents any student from tracking another student's individual movements. |
| Security | NFR-05 | Driver and Admin accounts shall require role-based authentication. | Ensures only verified drivers/admin can access sensitive controls like SOS handling or route changes. |
| Scalability | NFR-06 | System shall support fleet expansion from 1 to 10 vehicles without architecture redesign. | The system needs to grow with the university's fleet plan, not be rebuilt for it. |
| Scalability | NFR-07 | System shall handle concurrent boarding/payment requests from up to 500 students during peak hours. | Prevents the system slowing down exactly when it's needed most. |
| Usability | NFR-08 | Student app shall be usable by first-time users without a tutorial. | Boarding needs to be fast and obvious, since students won't read instructions while a bus is waiting. |
| Usability | NFR-09 | System shall support a non-smartphone fallback via SMS-based check-in and payment confirmation through the SMS Gateway. | Preserves access for the minority of students without reliable smartphone/data access, now that cash is no longer available as an alternative. |

# 5. System Scope

## 5.1 In Scope

- Student mobile application, Driver dashboard, Admin web dashboard
- Real-time GPS tracking and live seat capacity display
- Digital-only boarding payment, including SMS-based fallback via SMS Gateway
- SOS / delay alert system
- Analytics, fleet overview, driver scheduling, and demand reporting

## 5.2 Out of Scope

- Cash handling and manual cash accounting (fully discontinued)
- Physical vehicle hardware / telematics installation
- Implementation of the route-optimization AI algorithm (design only, not build)

- Integration with KLUST's core LMS / student information system
- Insurance and legal claims processing

# 6. Constraints & Ethical Considerations

## 6.1 GPS Tracking Privacy

- The system tracks the bus, not the individual student, students are not personally geolocated once they board or when not actively using the app.
- Location history is retained only for a short window (24–48 hours) to prevent long-term movement profiling.
- The system must comply with Malaysia's Personal Data Protection Act (PDPA), as it processes personal data.

### 6.2 Accessibility

- Not all students own a smartphone or have reliable data access. Since cash payment has been removed from the system, NFR-09 requires an SMS-based fallback (via the SMS Gateway) so these students can still check in and pay for boarding preventing digital exclusion, not merely offering a convenience.
- The user interface must meet baseline accessibility standards (colour contrast, text sizing) for students with visual impairments.

## 7. Requirement Elicitation Methods & Evidence

To ensure the requirements above reflect genuine user needs rather than assumptions, the team applied a mixed-methods elicitation approach combining breadth (survey data) with depth (observation and document review).

### 7.1 Methods Applied

| Method | Target | Purpose |
|---|---|---|
| Questionnaire | KLUST student shuttle users | Quantify demand-side pain points and feature demand |
| Direct Observation | Live shuttle stop, peak hours | Capture real boarding/queue behaviour firsthand |
| Document Analysis | Existing transport policy / payment process | Identify formal constraints and current system state |

### 7.2 Survey Instrument & Administration

A 10-question Google Forms survey was distributed to the official KLUST shuttle bus student group chat. The survey was anonymous, required no login, and took approximately two minutes to complete. It collected 21 valid responses (N=21).

### 7.3 Survey Findings

| Finding | Result | Justifies |
|---|---|---|
| Missed a class due to shuttle issues | 90.4% (occasionally or often) | Establishes core problem legitimacy |
| Couldn't tell where the bus was | 71.4% (15/21) | FR-01 |
| Not informed about delays in advance | 47.6% (10/21) | FR-09 |
| Would use a live tracking app | 100% positive (Definitely/Probably) | FR-01, FR-02 |
| Delay/breakdown alerts rated important | 100% (Very/Somewhat important) | FR-09 |
| Own a smartphone with reliable campus data | 95.2% Yes, 4.8% Sometimes | Informs NFR-09 scope (see 7.4) |
| Cash payment caused boarding delays | 19% (4/21) | Supported the decision to remove cash entirely (see 7.6) |
| Reserving a seat in advance rated useful | 52.4% Very useful / 47.6% Not useful at all | Informed the decision to remove advance reservation from scope (see 7.5) |

## 7.4 Honest Interpretation of Findings

The Three most critical elements in our study and the ones that we needed to interpret carefully, as opposed to confirming our existing views were:

- Seat reservation was contested in the data (52.4% found it useful vs. 47.6% who did not), and one respondent specifically warned that reservation could cause disputes between students over who a seat "really" belonged to. Combined with checkpoint defense feedback, this directly informed the decision to remove advance seat reservation from the system entirely. Boarding is now first-come, first-served (UC-01: Board Bus), removing the dispute risk at its source rather than attempting to design around it.
- Smartphone ownership is near-universal (95.2%), which weakens a "high measured need" justification for a non-smartphone fallback. NFR-09 is retained regardless as an ethical inclusivity safeguard, it now specifically provides SMS-based check-in and payment access via the SMS Gateway for the remaining minority, since cash is no longer available as an alternative.
- Open-text responses repeatedly raised fleet size and schedule alignment to class timings (e.g. "align the bus time to students' class time") this was not a designed survey question, but directly supports the AI-Driven Demand Forecasting innovation proposed in Section 10.

## 7.5 Design Decision: Seat Reservation vs. Removal

Two design paths were considered for handling seat allocation, and are weighed explicitly here rather than the removal being treated as a default.

- Option A — Retain reservation: This would require defining a reservation window (how far in advance a seat can be booked before it opens up to walk-up students) and a cancellation policy (how late a student can cancel before losing the seat or being penalised), to prevent the exact dispute risk raised in the survey data.
- Option B — Remove reservation entirely: Boarding becomes first-come, first-served, paid digitally at the point of boarding (UC-01: Board Bus).

We chose Option b. Because KLUST's shuttle runs relatively short and frequent loop trips, instead of running scheduled long-distance routes, the next bus is usually only minutes away. Therefore, adding a reservation window adds additional scheduling complexity without providing a corresponding benefit. If we choose to retain reservation (Option a), then we will also need to actively monitor no shows and implement procedures for cancelling reservations, in addition to implementing processes for resolving disputed claims related to fairness. While retaining reservation (Option a) would add additional constraints to managing the risks associated with fairness, eliminating reservation (Option b) eliminates that risk at the root level rather than trying to manage it after the fact. This tradeoff, not just the results of the survey was the justification for removing fr-03 (reserve Seat) from the System instead of adding rules to retain it.

## 7.6 Existing System Context (Document Review)

Review of the current shuttle payment process confirmed that KLUST's shuttle currently accepts both cash and digital payment. Based on checkpoint defense feedback, and to fully resolve the Manual Cash Payments problem identified in Section 1, the redesigned system discontinues cash handling entirely: FR-03 makes digital payment the sole boarding method, integrated directly into the boarding flow (UC-01), with automatic digital receipts (FR-04) replacing manual cash accounting. NFR-09 preserves accessibility for students without reliable smartphone/data access through an SMS-based fallback via the SMS Gateway, rather than through cash.

## 8. Use Case Diagram

The diagram below models all primary and secondary actor interactions with the KLUST CampusShuttle System, including <<include>> and <<extend>> relationships, drawn in Visual Paradigm.

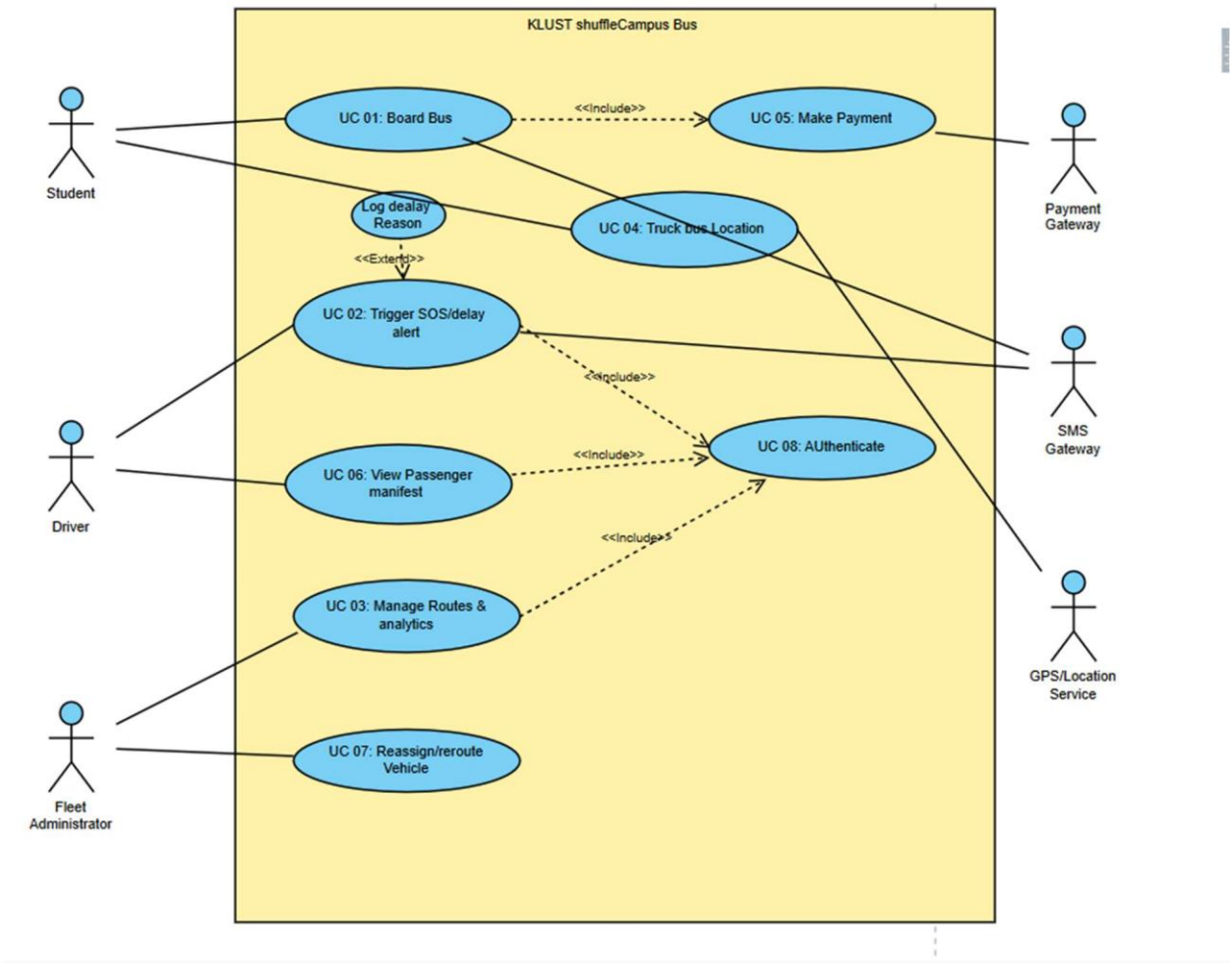



Figure 1: KLUST CampusShuttle System Use Case Diagram (Visual Paradigm)

# 9. Use Case Specifications

Full specifications for the three major use cases, as required by the assignment brief.

## 9.1 UC-01: Board Bus

| | |
|---|---|
| **Use case ID** | UC-01 |
| **Use case name** | Board Bus |
| **Description** | Student wants to board the shuttle bus and pay digitally, without needing an advance reservation. |
| **Actor** | Student (primary), Payment Gateway (secondary), SMS Gateway (secondary — for students without app access) |
| **Pre-condition** | Student is at the stop when the bus arrives; live seat capacity indicates space is available. |
| **Post-condition** | Payment is processed, a digital receipt is generated, and seat capacity is updated in real time. |
| **Basic flow** | 1. Student checks live bus location and seat capacity before walking to the stop.<br>2. Bus arrives at the stop.<br>3. Student boards; system identifies them via their account (app) or SMS-based check-in.<br>4. System prompts for digital payment.<br>5. Student completes payment.<br>6. System generates a digital receipt and sends confirmation (in-app or via SMS Gateway).<br>7. System decrements the live seat capacity count for that trip. |
| **Exceptions** | Step 1/2: Bus is at full capacity on arrival. System flags no space is available; student is advised to wait for the next bus.<br>Step 5: Payment fails. System allows a 1-minute retry window before the student is asked to exit or wait for the next bus.<br>Step 3: Student has no smartphone/app access. System falls back to SMS-based check-in and payment confirmation via the SMS Gateway. |

### 9.2 UC-02: Trigger SOS/Delay Alert

| | |
|---|---|
| **Use case ID** | UC-02 |
| **Use case name** | Trigger SOS/Delay Alert |
| **Description** | Driver wants to instantly alert Admin and passengers of an emergency or delay. |
| **Actor** | Driver (primary), Fleet Administrator and affected Students (notified via app and SMS Gateway) |
| **Pre-condition** | Driver is authenticated and assigned to an active trip. |
| **Post-condition** | Alert is logged and delivered to Admin and affected passengers in real time. |
| **Basic flow** | 1. Driver opens the dashboard during an active trip. |
| | 2. Driver taps the "SOS" button.<br>3. System captures current GPS location and timestamp.<br>4. System immediately pushes the alert to the Admin dashboard.<br>5. System notifies students tracking that route via the app, and non-app users via the SMS Gateway.<br>6. Admin acknowledges the alert and takes action (may trigger UC-07: Reassign/Reroute Vehicle). |
| **Exceptions** | Step 2: Driver selects "Delay" instead of SOS. System prompts the driver to log a delay reason (extends to Log Delay Reason).<br>Step 4: Network failure at trigger time. System queues the alert and retries transmission every 10 seconds until delivered. |

### 9.3 UC-03: Manage Routes & Analytics

| | |
|---|---|
| **Use case ID** | UC-03 |
| **Use case name** | Manage Routes & Analytics |
| **Description** | Admin wants to view demand analytics, oversee the live fleet, configure routes/schedules, and manage driver assignments. |
| **Actor** | Fleet Administrator |
| **Pre-condition** | Admin is authenticated. |
| **Post-condition** | Route/schedule is updated, driver assignments are set, and/or an analytics report is generated. |

| | |
|---|---|
| **Basic flow** | 2. Admin selects a view: Analytics, Fleet Overview, Driver Assignments, or Route/Timetable Configuration.<br>3. System displays the relevant data (e.g. peak-hour demand, live bus positions, current driver shifts).<br>4. Admin reviews the data and makes an adjustment (e.g. modifies a route, reassigns a driver, exports a report).<br>5. System validates and saves the changes.<br>6. System pushes any schedule/route updates to student and driver apps. |
| **Exceptions** | Step 5: Route or schedule conflict detected (e.g. overlapping timetable). System flags the conflict and blocks saving until resolved. |

# 10. Innovation & Advanced Features

Beyond the baseline requirements, two features are proposed in depth, selected because they extend use cases already modelled in this document and are supported by elicitation evidence, rather than being speculative additions.

## 10.1 AI-Driven Demand Forecasting

- Problem it solves: Fleet Administrators currently have no way to anticipate demand only to react to it after congestion occurs (Blind Fleet Management).
- How it works: Cross-references historical ridership data (FR-07) with KLUST's semester class timetable to predict surges around common class-end times, before they happen.
- Extends: UC-03 (Manage Routes & Analytics) — the analytics dashboard would surface predictive alerts rather than only historical reports.
- Evidence: Multiple open-text survey responses independently requested schedule alignment with class timings (Section 7.4), directly supporting this feature's real-world relevance.
- Feasibility: Scoped to KLUST's realistic 8–10 vehicle fleets a lightweight scheduling-pattern model, not an enterprise-scale prediction system.

## 10.2 Dynamic Route Optimization

- Problem it solves: Fixed loop routes cannot adapt to real-time disruptions such as an SOS/delay event.
- How it works: When UC-02 (Trigger SOS/Delay Alert) fires, the system suggests an AI-recommended reroute instead of requiring the Admin to decide manually from scratch.
- Extends: UC-07 (Reassign/Reroute Vehicle) turns a manual admin decision into an assisted one.
- Feasibility: Builds directly on infrastructure already required for FR-09/FR-11, keeping added complexity low.

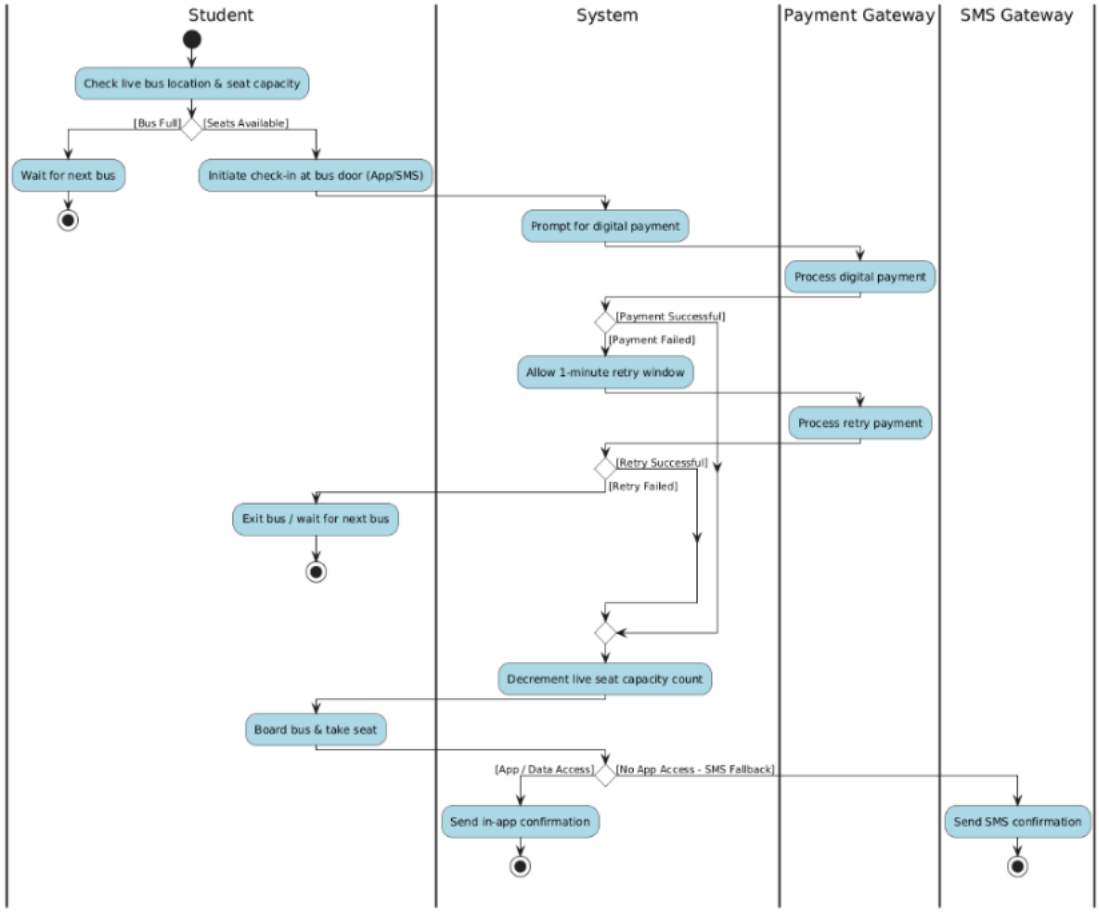
## 10.3 Secondary Idea (brief)

- Gamified off-peak ridership incentive: students earn redeemable points (e.g. campus café discounts) for boarding during off-peak trips, reducing overcrowding without forcing behavior change.

# 11. Diagrams

**Activity Diagram 1 : UC-01 Board Bus**

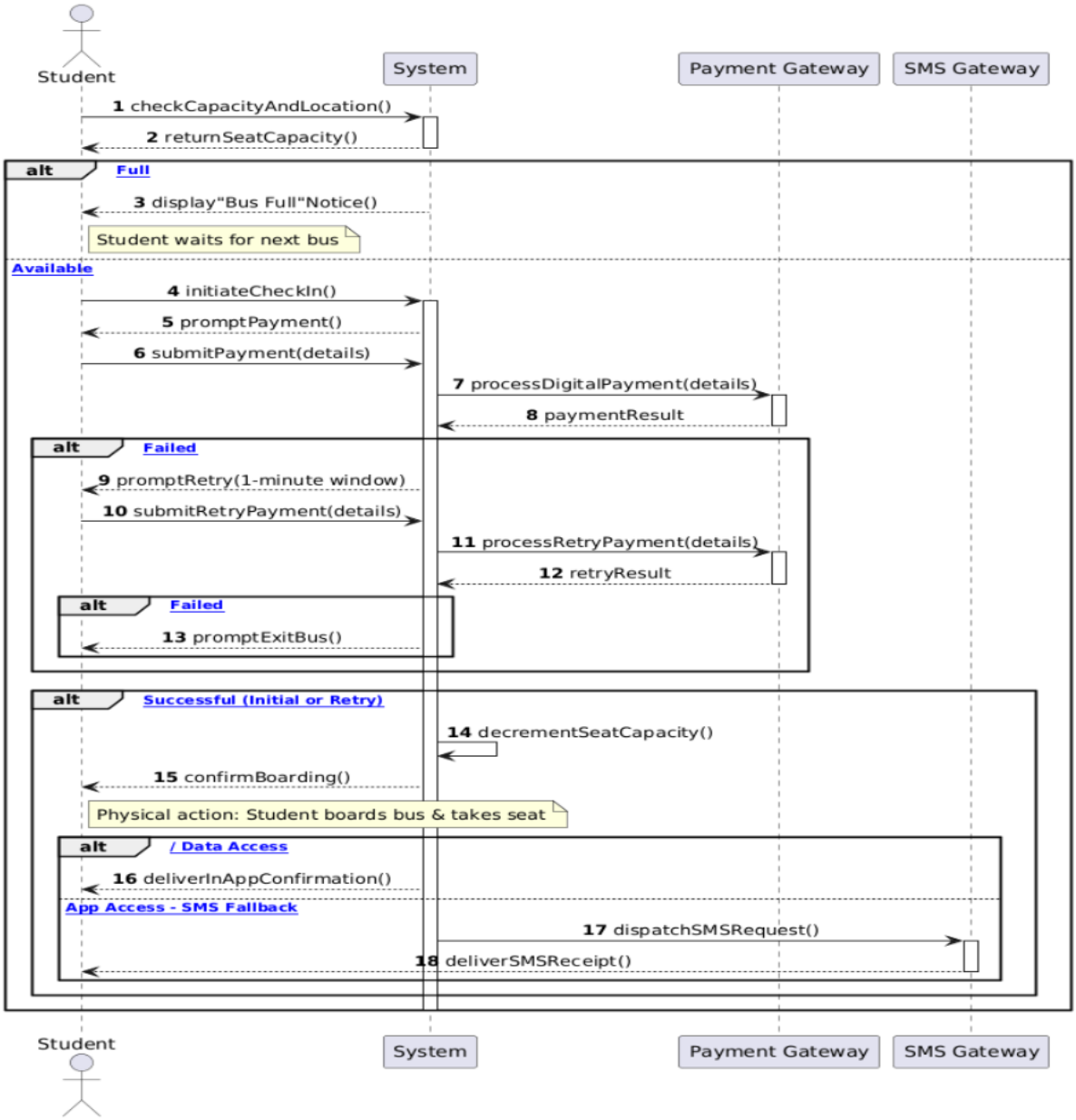
This activity diagram shows the step-by-step workflow for **UC-01 (Board Bus)** across four swimlanes: Student, System, Payment Gateway, and SMS Gateway. It starts with an upfront capacity check on the app so students don't waste time at the stop if the bus is full. If seats are available, the system processes digital payment and includes a 1-minute retry window if the transaction fails. Once payment succeeds, the system automatically decrements the live seat count and sends a confirmation receipt via app or SMS fallback.




**Sequence Diagram 1 : UC-01 Board Bus**

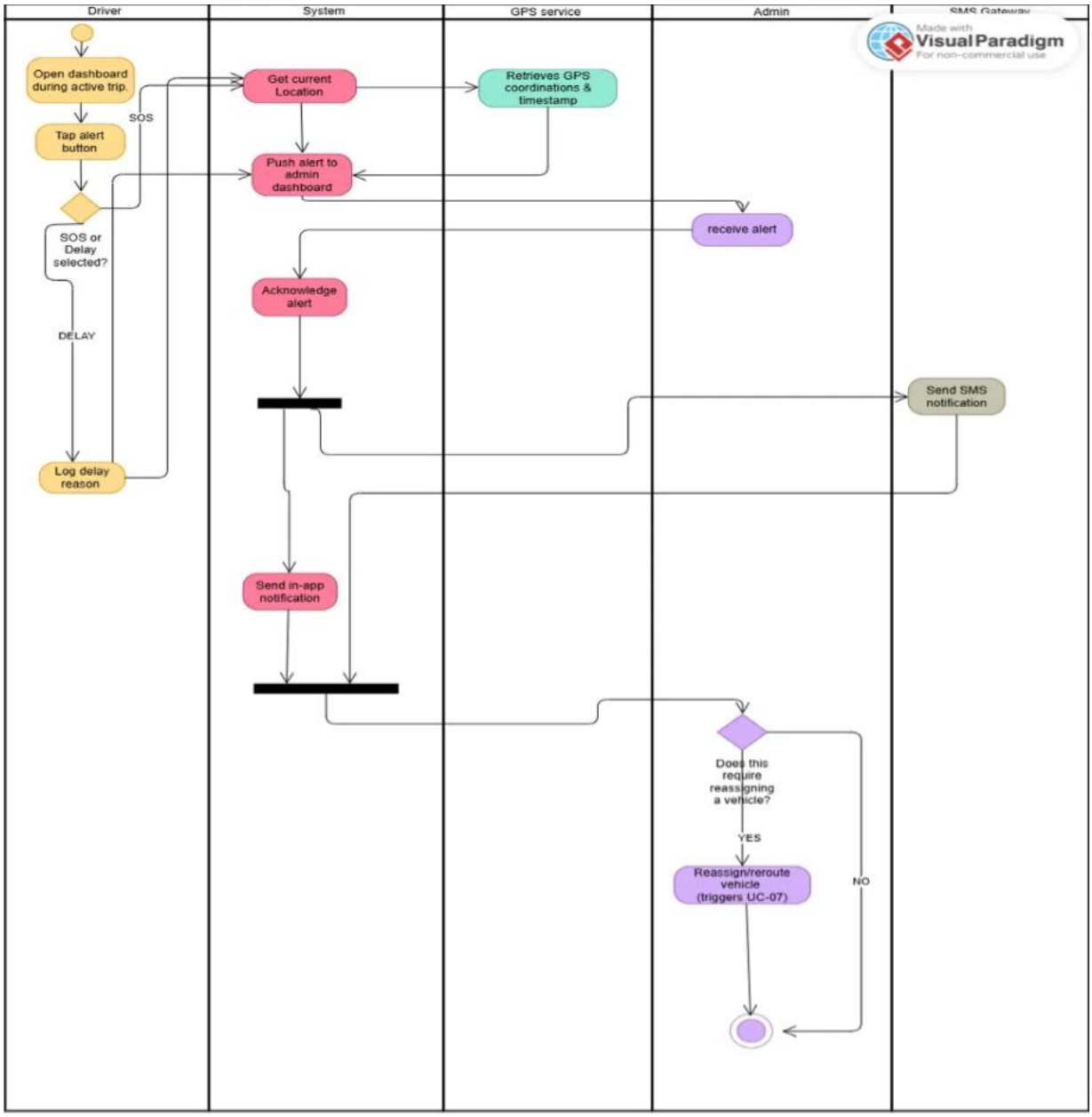
The below diagram demonstrates the interaction between entities that occurs as part of the process for UC-01 (Board Bus),

and represents the ordered set of events involved with this use case. The diagram utilizes a Boundary-Control-Entity (bce) model where the Student user interacts with the Boarding UI application; it then delegates the task of verifying whether there is sufficient seating available on the bus to both the boarding controller and the trip entity. After the Student has made their payment, the payment processing operation will be performed by an external payment Gateway. As long as the payment was processed successfully, the number of seats available on the bus will be decreased. Then, at some later point in time, the Notification Service will notify all of the students who boarded the bus either through receipt delivery within the app or through SMS if they do not have access to the app.



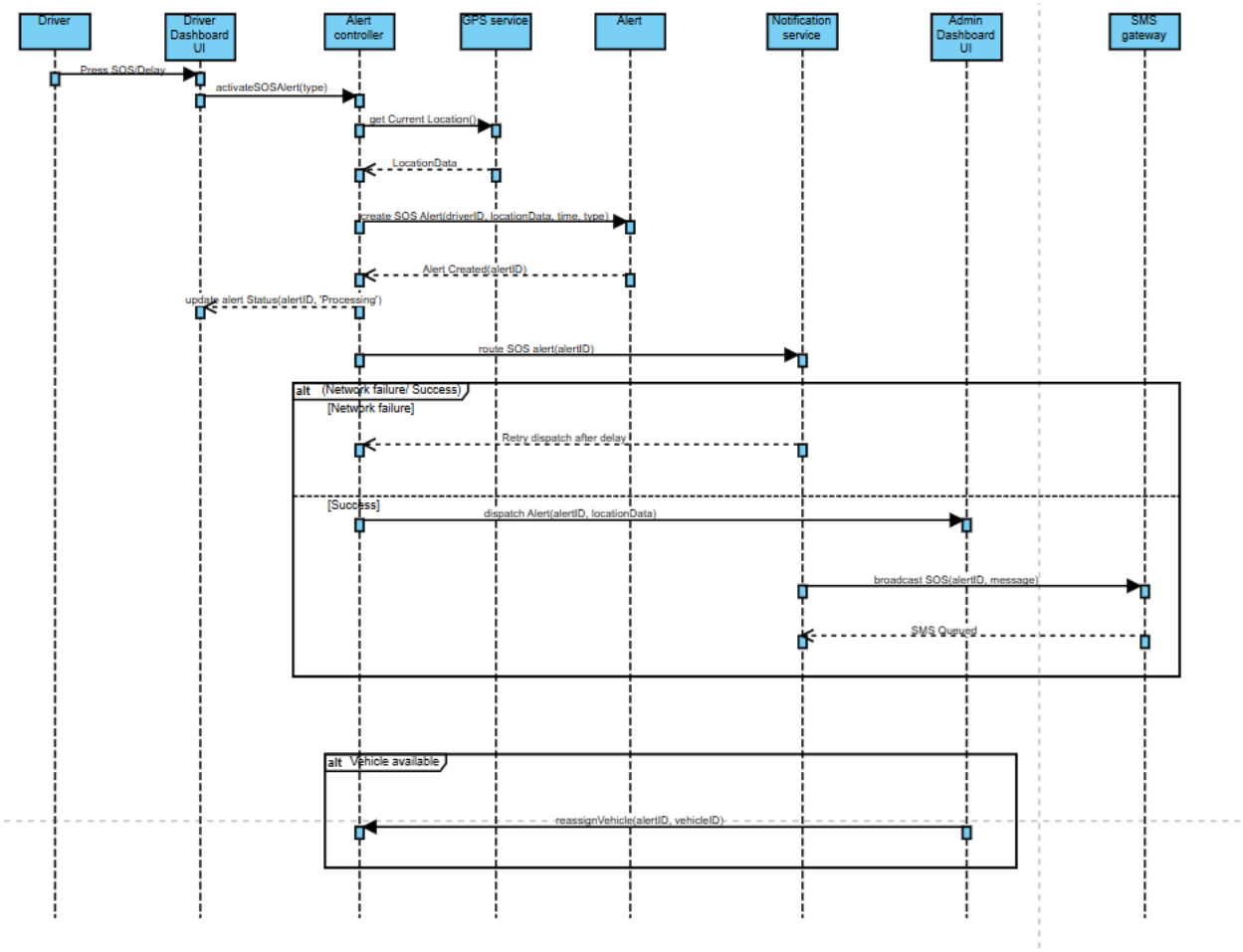

**Activity Diagram 2: UC-02 Trigger SOS/Delay Alert**

The activity diagram illustrated below represents the operational flow for uc-02 (Trigger SOS/Delay Alert), which involves several different entities including but not limited to the Driver, System, gps service, admin, and SMS Gateway. Upon entry into this flow by the driver, it will automatically capture the current location based upon gps data and log that the driver has encountered a delay or sos emergency. Immediately after capturing this information, an alert will be sent to the Admin Dashboard regarding the status of the driver's route. Additionally, students who were previously scheduled to ride on the same bus will receive notification messages via either their app or SMS about what happened. However, prior to sending these notifications, they must have been able to successfully determine whether the disruption caused by the emergency was severe enough to require that a new bus be assigned to continue transporting them.



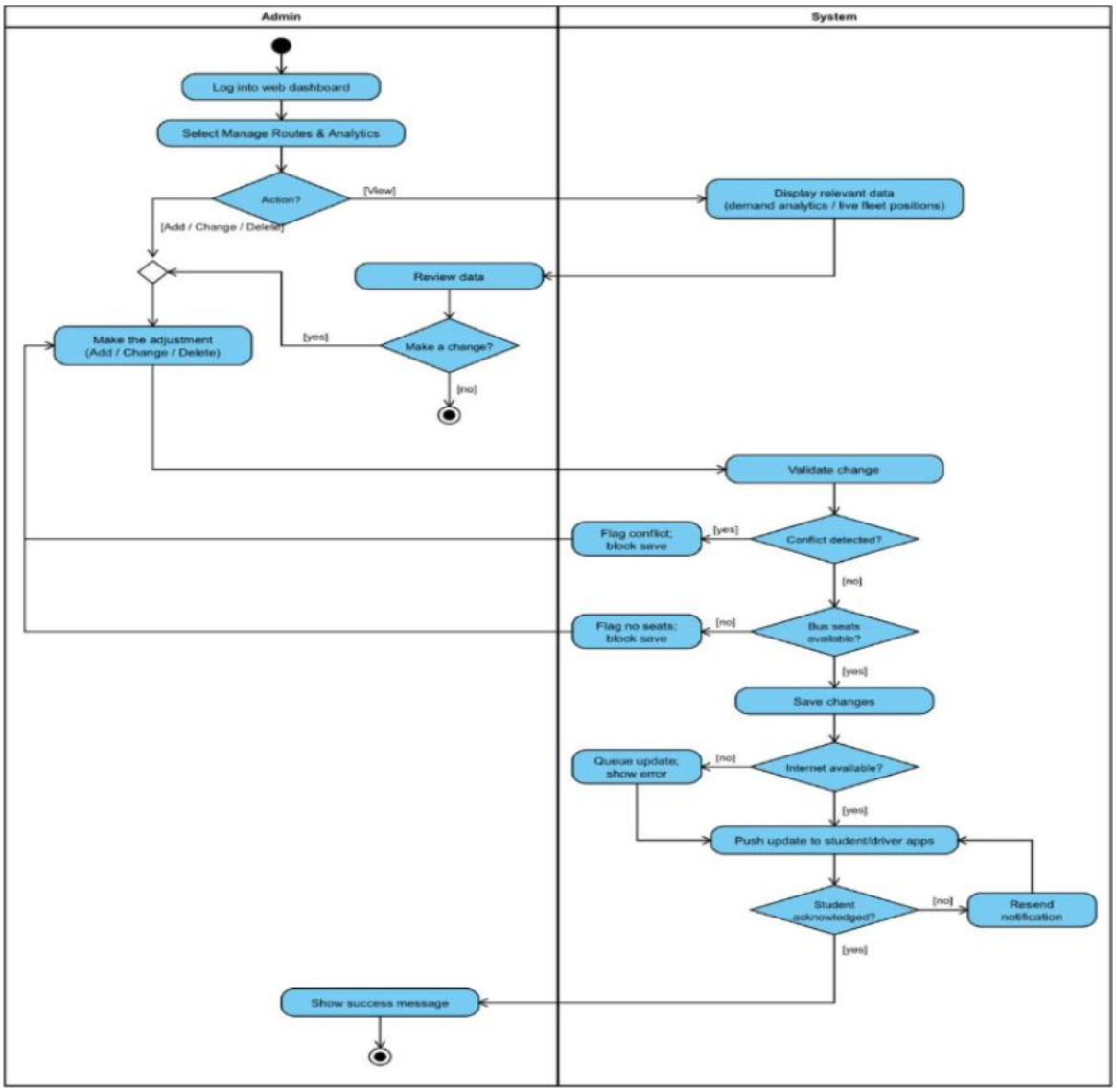
**Sequence Diagram 2: UC-02 Trigger SOS/Delay Alert**

The Sequence Diagram shown here illustrates how uc-02 (Trigger SOS/Delay Alert) operates. Specifically, it shows how the Alert Controller coordinates with the GPS Service in order to obtain gps coordinates. Then, based upon those coordinates, it generates an alert entity and modifies the driver dashboard ui accordingly. There is also an alternate branch in this diagram for cases where communication failures prevent successful delivery of alerts to the Admin Dashboard and/or SMS Gateway. Similarly, there is another alternate branch in this diagram for scenarios where the admin wishes to notify drivers of vehicles whose routes need to be reassigned due to disruptions caused by emergencies.



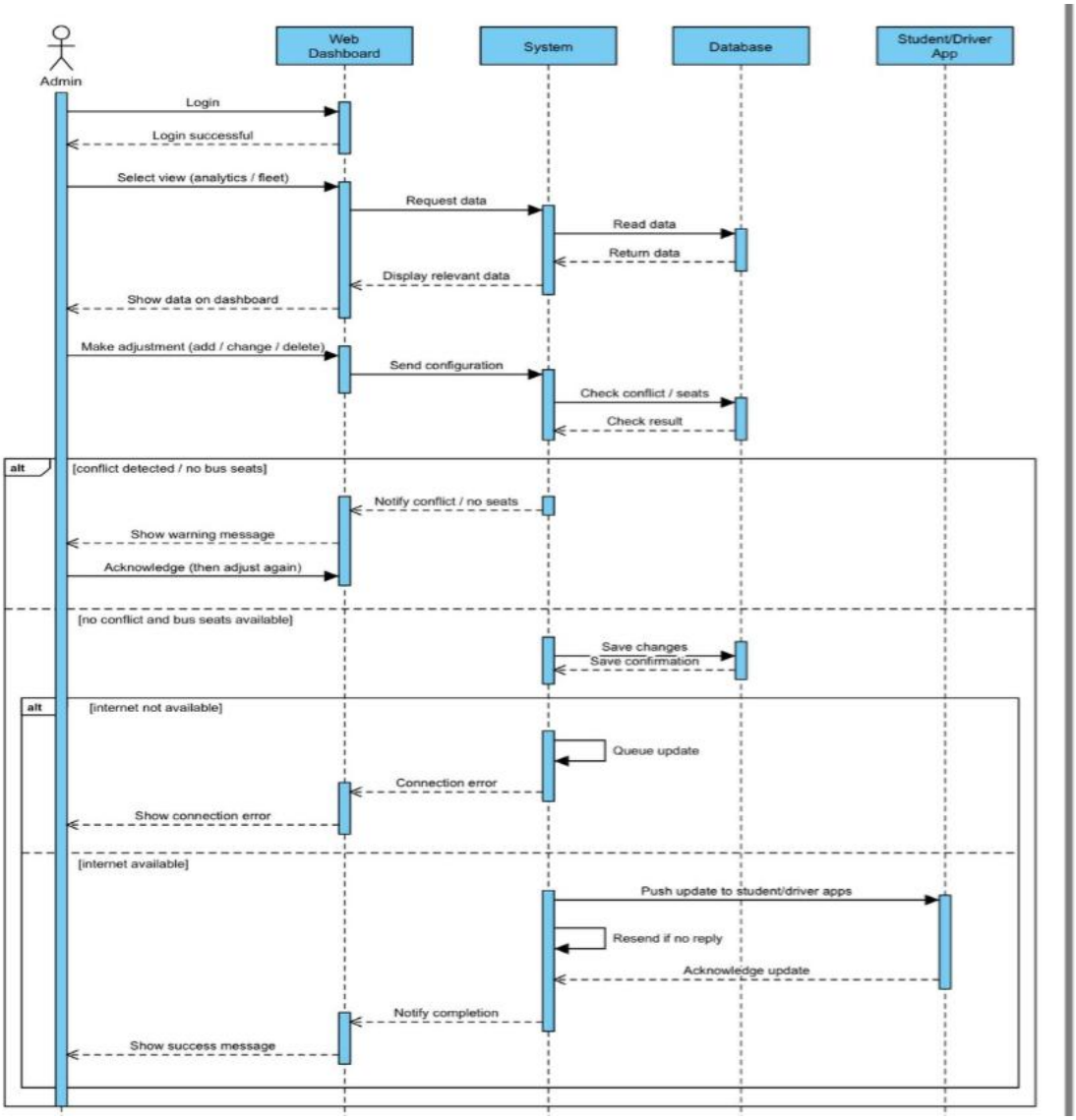
**Activity Diagram 3 : UC-03 Manage Routes & Analytics**

The above Activity Diagram shows the Administrative Workflow for UC-03 (Manage Routes & Analytics) that is divided into the Admin and System Swimlane. The diagram divides viewing read only information from configuring routes by adding new routes, modifying existing routes or deleting routes. Once an admin submits a change it will be validated through the system as well as checked for capacity/schedule conflict. If there are no conflicts, then the changes are sent to the database. Then those changes are pushed out in real time to both student and driver applications.



**Sequence Diagram 3 : UC-03 Manage Routes & Analytics**

The Above Sequence Diagram represents the interaction of different systems involved in UC-03 (Manage Routes & Analytics), which include the web dashboard user interface, System Controller, Database Entity and Mobile Apps. There is an alt fragment used to separate the scheduling conflict check from updating the database. Additionally, there is another alt fragment, also nested, that allows you to show queuing for offline app updates when disconnected from the Internet vs. real time syncing and acknowledging with Student and Driver Devices.



## 12. User Interface Prototypes

Interface designs for each of the three primary actors Student, Driver, and Fleet Administrator, were generated using Figma Make, an AI design tool, as permitted by the assignment brief. Each interface was designed to directly reflect the functional and non-functional requirements, use cases, and innovation features established in Checkpoints 1 and 2.

**Figma prototype link:** [https://www.figma.com/make/quETsRYrn1d1IsvmJlTsyo/KLUST-Shuttle-UI-Kit-Design?code-node-id=0-6&p=f&t=pK6QmTO6IlAYkEtU-0&fullscreen=1](https://www.figma.com/make/quETsRYrn1d1IsvmJlTsyo/KLUST-Shuttle-UI-Kit-Design?code-node-id=0-6&p=f&t=pK6QmTO6IlAYkEtU-0&fullscreen=1)

# 13. Screen 1 — Student Mobile App

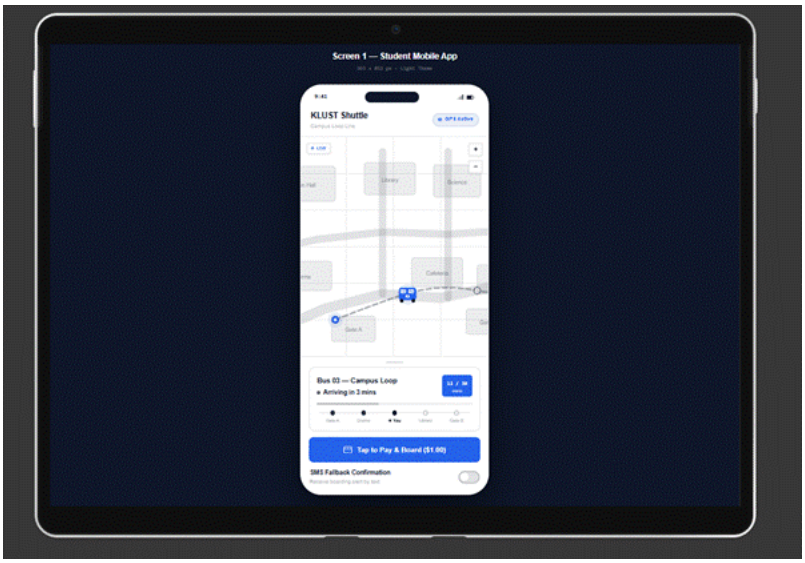



*Figure 2: Student Mobile Interface — live bus tracking and digital boarding (Figma Make)*

The live map sits in front and center, since the main question a student has is where the bus is and whether it is worth walking to the stop. Below it, arrival time and free seats are shown clearly, as that information directly drives the student's decision. Boarding is a single tap pay and board together, fully digital, with no cash option anywhere in the flow.

A notable design choice is the SMS Fallback Confirmation toggle. Not every student has reliable data access on campus, so instead of hiding that fallback in the backend, it was made a visible setting the student controls themselves directly surfacing the system's accessibility requirement as a real interface feature.

# 14. Screen 2 — Driver Tablet Dashboard

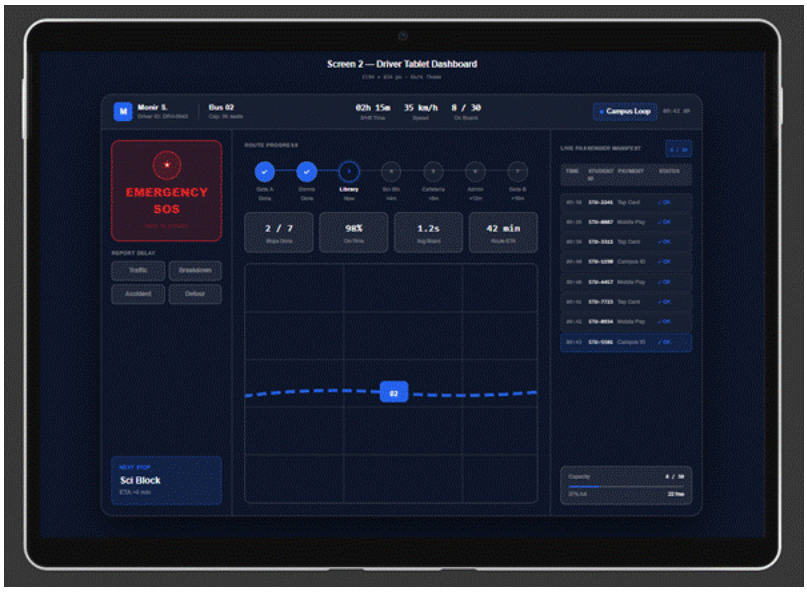



*Figure 3: Driver Tablet Dashboard — emergency SOS, delay reporting, and live manifest (Figma Make)*

Built for a tablet mounted in the bus, the layout is designed for quick glances rather than careful reading. The Emergency SOS button is the largest and most visually dominant element on the screen, with no menus to navigate during an emergency. Four delay-reason options (Traffic, Breakdown, Accident, Detour) let the driver report routine issues just as quickly.

The centre of the screen tracks route progress and live stats, while the passenger manifest on the right shows every boarded student's payment method consistently card or mobile payment, reflecting the system's fully cashless design.

## 15. Screen 3 — Admin Web Dashboard

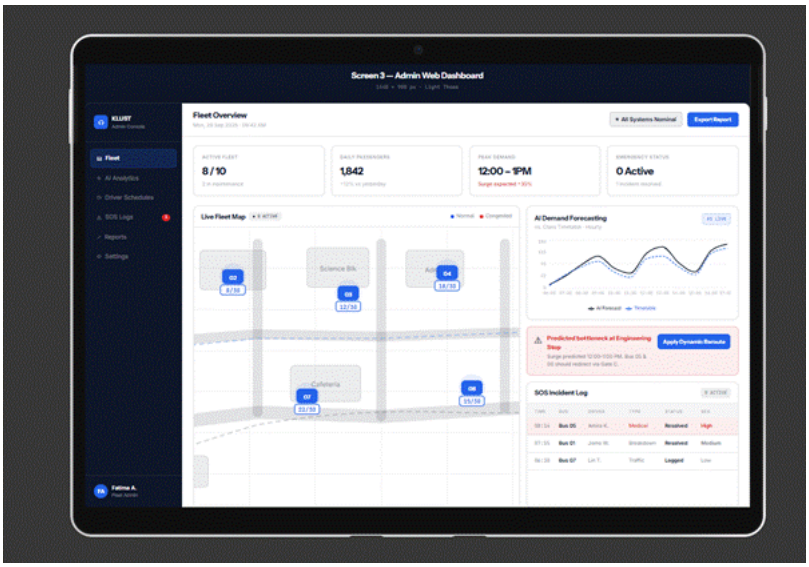



*Figure 4: Admin Web Dashboard — fleet overview, AI demand forecasting, and incident log (Figma Make)*

Designed as a full desktop console, since the Fleet Administrator manages the whole fleet rather than a single bus. The top row gives an at-a-glance fleet health check, and the live map shows every active bus with its current capacity.

Two panels on this screen directly operationalise the Innovation features proposed in Checkpoint 1: an AI Demand Forecasting chart comparing predicted ridership against the class timetable (Innovation — AI-Driven Demand Forecasting), and a predictive bottleneck alert with a one-click "Apply Dynamic Reroute" action (Innovation — Dynamic Route Optimization). Rather than existing only as written proposals, both innovations are shown here as features an administrator

could genuinely act on. The SOS Incident Log at the bottom maintains a full audit trail of every alert raised in the system.

## 16. Design Summary

- Live, real-time data is surfaced wherever a user decision depends on it (bus location, seat capacity, fleet status).
- The cashless payment model is reflected consistently  no cash option appears on any of the three screens.
- Both proposed Innovation features are represented as working interface elements, not only as written proposals.
- Accessibility  (SMS fallback) is designed as a visible, user-controlled setting rather than an invisible backend rule.

## 17. APPENDIX A: AI PROMPT EVIDENCE

### 17.1 A.1 AI Tool Used

**Tool:** Figma Make

**Project:** KLUST  Shuttle — Campus Shuttle System UI Kit

**Purpose:** To generate and refine three connected user interface prototypes: the Student Mobile App, Driver Tablet Dashboard, and Admin Web Dashboard.

### 17.2 A.2 Prompt 1: Initial UI Kit Design

Design a 3-screen UI kit for "KLUST  Shuttle", a campus shuttle ecosystem, using a color palette of deep navy blue (#0D253A), electric blue (#007AFF), status green (#34C759), and alert red (#FF3B30). Screen 1: Student Mobile App (393x852 px) - Top bar: "KLUST Shuttle" title with a pulsing green "GPS  Active" status pill. - Canvas: Vector campus map with shuttle route line and live bus icon "Bus 03". - Bottom sheet: Card showing "Bus 03 — Campus Loop Line", ETA "Arriving  in 3 mins", and green capacity badge "12 / 30 Seats Available". - Buttons: Primary CTA "Tap to Pay & Board ($1.00)" with payment icons, and a toggle for "SMS Fallback Confirmation". - Pop-up modal: Receipt showing green checkmark, "Boarding Confirmed", transaction ID, timestamp, and updated seat count. Screen 2: Driver Tablet Dashboard (1194x834 px, Dark Mode) - Tactical dark UI (#121212) with high contrast and large glanceable buttons. - Header bar: Driver name "Monir S.", "Bus 02 (Cap: 30)", shift timer "02h 15m", speed "35 km/h". - Left panel: Massive, prominent red "EMERGENCY  SOS" button with yellow hazard outline, and quick delay action chips [Traffic], [Breakdown], [Accident]. - Center panel: Horizontal route progress bar tracking completed, current, and upcoming stops. - Right panel: "Live Passenger Manifest" table showing Time, Student ID, Payment Type, and green "Verified" status tags. Screen 3: Admin Web Dashboard (1440x900 px) - Desktop SaaS layout with dark navy sidebar (#0F172A) containing menu links (Fleet, AI Analytics, Driver Schedules, SOS Logs). - Top row: 4 metric summary cards for Active Fleet (8/10), Daily Passengers, Peak Demand Alert, and Emergency Status. - Main left (60%): Interactive campus map with live GPS pins and capacity badges for all 8 active shuttle buses. - Main right (40%): "AI Demand Forecasting vs. Class Timetable" line chart predicting hourly surges, a "Dynamic Route Optimization" alert banner with an "Apply Dynamic Reroute" button, and an SOS incident log table.

### 17.3 A.3 Prompt 2: Improved UI Design and Color Consistency

Design a cohesive 3-screen UI kit for "KLUST  Shuttle" strictly restricted to 4 colors: Slate Navy (#0F172A), Pure White (#FFFFFF),  Electric  Blue (#2563EB), and Signal Red (#DC2626).

Screen 1: Student Mobile App (393x852 px, Light Theme)

- Canvas: Pure White (#FFFFFF) background with subtle Slate Navy (#0F172A) route lines and a live Electric Blue (#2563EB) bus marker "Bus 03".

- Top Bar: Slate Navy text "KLUST Shuttle" with an Electric Blue "GPS  Active" status pill.

- Bottom Card: Pure White card with Slate Navy title "Bus 03 — Campus Loop", "Arriving in 3 mins", and an Electric Blue capacity badge "12 / 30 Seats Available".
- Actions: Primary CTA button in solid Electric Blue (#2563EB) reading "Tap to Pay & Board ($1.00)", and a Slate Navy toggle for "SMS Fallback Confirmation".
- Pop-up Overlay: Pure White receipt modal with an Electric Blue checkmark, Slate Navy text "Boarding Confirmed", transaction ID, and updated seat count.

Screen 2: Driver Tablet Dashboard (1194x834 px, Dark Theme)

- Canvas: Full Slate Navy (#0F172A) background with high-contrast Pure White (#FFFFFF) typography for glanceable driving usability.
- Header Bar: Pure White text showing "Driver: Monir S.", "Bus 02", Shift Timer "02h 15m", and Speed "35 km/h".
- Left Panel: Massive, high-visibility Signal Red (#DC2626) "EMERGENCY SOS" button, above Slate Navy outline buttons for quick delays [Traffic], [Breakdown], [Accident].
- Center Panel: Minimal horizontal route progress bar using Electric Blue (#2563EB) for active stops and Pure White for upcoming stops.
- Right Panel: "Live Passenger Manifest" table on Slate Navy surface with Pure White text and Electric Blue "Verified" status tags.

Screen 3: Admin Web Dashboard (1440x900 px, Light Theme)

- Navigation: Slate Navy (#0F172A) left sidebar with Pure White menu text (Fleet, AI Analytics, Schedules, SOS Logs).
- Main Body: Pure White (#FFFFFF) background containing 4 top stat cards in Pure White with Slate Navy text.
- Left Section (60%): Minimalist campus map with Electric Blue (#2563EB) bus pins. Congested buses flagged with a Signal Red (#DC2626) badge.
- Right Section (40%): Clean Slate Navy line chart for "AI Demand Forecasting", a Signal Red alert banner reading "Predicted bottleneck at Engineering Stop", an Electric Blue button reading "Apply Dynamic Reroute", and a clean SOS log table.

### 17.4 A.4 Prompt 3: Final UI Corrections

Make two changes across the KLUST Shuttle UI Kit: 1) On Screen 1 (Student Mobile App), change the boarding button price from '$1.00' to 'RM 1.00' to match Malaysian Ringgit currency used throughout the rest of the system. 2) On Screen 2 (Driver Tablet Dashboard), the live route map panel in the center is currently rendering empty/blank — fix it so it displays the live route map with the bus position and stops, consistent with how the map renders on Screen 1 and Screen 3.

### 17.5 A.5 Summary of AI-Assisted Design Process

The KLUST Shuttle UI Kit evolved because of three rounds of prompts. Round one prompted the overall structure (three screen systems) and the essential functions of the UI Kit. Round two focused on refining the visual aspects by creating a consistent color scheme throughout the entire application and defining a light theme and a dark theme for each interface within the application. Round three improved upon two specific areas: showing the boarding cost in Malaysian Ringitts; fixing the blank route map on the driver tablet dashboard.

Thus, there are three distinct examples that exemplify an iterative approach to designing an interface where the

original UI ideas were improved to increase visual consistency, accurate representation of currency, and the display of route data.