JoseniCare
USE CASE DIAGRAM

JoseniCare

System
- Send automated alerts and notifications
- Perform data backup and recovery
- Auto-award gamification stamps
- Generate system analytics reports
- Trigger health trend updates
- Detect potential outbreaks
- Log all major user activities

Administrator
- Manages user accounts
- Configure system settings (security, access rules, audit trails)
- Set and update gamification criteria
- Access admin dashboard

USJ-R Tertiary Students
- Book and view clinic / dental appointments
- Submit daily health survey
- View digital health record
- View appointment history
- Report accidents through SSD emergency channel
- Interact with Jose the chatbot
- Track gamification progress

Security & Safety Department
- Receive Emergency Reports
- Access Allergy Information (Limited View)
- Manage Emergency Reports

Clinic Staff
- Manage Appointment Scheduling
- Track Anonymized Survey Results
- View Doctor's prescriptions/treatment notes
- Manage Resource Dashboard (Inventory)
- View Health Dashboards
- Export or print patient consultation records

Doctor
- View Student Appointments
- Access and Update Digital Health Records of Patient
- View Anonymized Health Trends (Dashboard)
- View Medicine & Resource Utilization Dashboard

SYSTEM USE-CASES

ACTOR

UML Use Case Description: Send Automated Alerts and Notifications

USE CASE #1:	**SEND AUTOMATED ALERTS AND NOTIFICATIONS**
Primary Actor:	System
Goal	The system automatically sends notifications to users (students, clinic staff, or SSD) based on events such as appointment reminders, survey prompts, or emergency updates.
Preconditions	• User accounts and notification preferences are active.
Postconditions	• Users receive appropriate and timely alerts.
Trigger	Detect event passed to System.
Main Flow	1. The System detects an event that requires a notification, such as: • Upcoming appointment • Health survey reminder • SOS confirmation or status update 2. The System generates the appropriate notification and sends it via: • In-app alert • University e-mail 3. The user receives the notification and can act on it as needed.
Alternative Flows	**A1. Notification service unavailable:** If the system cannot send notifications (in-app or email), it retries and logs the failure. **A2. User opted out:** If the user has disabled notifications, the alert is not sent but logged for tracking purposes. **A3. Invalid event detected:** If the event triggering the notification is malformed or missing, the system skips sending and logs the issue.

UML Use Case Description: Perform Data Backup and Recovery

USE CASE #1:	**PERFORM DATA BACKUP AND RECOVERY**
Primary Actor:	System
Goal	The system automatically performs periodic backups of critical health and appointment data to prevent data loss.
Preconditions	• Backup schedule and storage configuration are active.
Postconditions	• Data backups are successfully stored. • Recovery option is available in case of system failure.
Trigger	Scheduled event every 8 pm from Monday to Saturday.
Main Flow	1. The System initiates a scheduled backup according to the configured interval. 2. The System compresses the relevant data and stores it securely in the designated backup storage. 3. The System generates a confirmation log indicating the success or failure of the backup.
Alternative Flows	**A1. Backup storage unavailable:** If storage cannot be accessed, the system retries and alerts the administrator. **A2. Data compression error:** If compression fails, the system logs the error and continues with uncompressed backup if possible. **A3. Confirmation log generation fails:** If log cannot be created, system retries and sends alert to administrator.

UML Use Case Description: Auto-Award Gamification Stamps

USE CASE #1:	**AUTO-AWARD GAMIFICATION STAMPS**
Primary Actor:	System
Goal	The system automatically awards points or stamps to students who meet certain health participation milestones (daily surveys).
Preconditions	• Gamification rules are set by the administrator.
Postconditions	• Student profiles are updated with new rewards or progress points.
Trigger	Daily scheduled gamification checking per batch.
Main Flow	1. The System checks the student's activity logs including: • Daily health survey submissions • Claiming of gamification reward 2. The System validates each action against the configured reward criteria • (HP (Health Points) per check-in • Milestone achievements 3. The System automatically awards the corresponding points or HP stamps and records them in the student's gamification profile. 4. The System updates the student's dashboard and reward progress in real-time.
Alternative Flows	**A1. Activity not logged:** If the student activity is missing or corrupted, no points/stamps are awarded, and an error is logged. **A2. Reward criteria misconfiguration:** If the reward rules are invalid, system skips awarding and alerts the administrator. **A3. Database save failure:** If points/stamps cannot be saved, retry occurs; if still failing, alert is sent to admin.

UML Use Case Description: Generate System Analytics Reports

USE CASE #1:	**GENERATE SYSTEM ANALYTICS REPORTS**
Primary Actor:	System
Goal	The system compiles reports on user activity, health data trends, and system performance for administrative and medical review.
Preconditions	• System databases contain sufficient data for report generation.
Postconditions	• Reports are available for viewing or export by authorized personnel.
Trigger	System detects new data submissions.
Main Flow	1. The System aggregates relevant data from various modules (Student health surveys, appointments, gamification activity, weather API). 2. The System processes the data and summarizes key metrics, such as: • Trends in student health (common symptoms, consultation frequencies) • Resource usage (medicine/dental supplies, inventory trends) • Gamification statistics (points earned, check-in streaks) • Noted Weekly Weather reports 3. The System stores the generated reports and makes them accessible through dashboards for authorized users (Doctors, Nurses, Administrators).
Alternative Flows	**A1. No data available:** If no data exists for the requested period, the system displays “No data available” in the dashboard. **A2. Report generation error:** If data processing fails, the system logs the error and optionally retries the report generation. **A3. Unauthorized access:** If a non-authorized user attempts to view reports, the system denies access and logs the attempt.

UML Use Case Description: Trigger Health Trend Updates

USE CASE #1:	**TRIGGER HEALTH TREND UPDATES**
Primary Actor:	System
Goal	The system periodically updates health dashboards to reflect new survey inputs, appointment results, and clinic data.
Preconditions	• Latest user data and health inputs are available.
Postconditions	• Health dashboards display updated and accurate insights.
Trigger	System receives new records.
Main Flow	1. The System scans for new student survey submissions or appointment records. 2. The System recalculates relevant health statistics, such as: • Trends in reported symptoms • Average consultation reasons • Attendance patterns for check-ins or appointments 3. The Dashboard visuals are updated in real time, ensuring authorized users (Doctors, Nurses, Administrators) see the latest metrics.
Alternative Flows	**A1. No new records detected:** If no new surveys or appointments are found, the dashboard remains unchanged. **A2. Update calculation fails:** If recalculation of statistics fails, system logs the error and keeps previous metrics. **A3. Dashboard refresh error:** If dashboard fails to update, the system retries and logs the error if unsuccessful.

UML Use Case Description: Detect Potential Outbreaks

USE CASE #1:	**DETECT POTENTIAL OUTBREAKS**
Primary Actor:	System
Goal	The system monitors aggregated health data to detect unusual symptom patterns that may indicate a potential outbreak.
Preconditions	• Active collection of daily health surveys and clinic records.
Postconditions	• An alert is sent to clinic and admin users if a potential outbreak is detected.
Trigger	System detects exceedance in defined spike levels in symptoms utilized for analysis.
Main Flow	1. The System analyzes anonymized student health survey trends over time. 2. The System identifies abnormal spikes or unusual patterns in symptoms or illness reports. 3. When a potential outbreak is detected, the System sends an alert to authorized personnel (Doctors, Nurses, or Administrators) for further review and action.
Alternative Flows	**A1. Insufficient data:** If survey data is too sparse for meaningful trend analysis, no alert is generated. **A2. False positive detection:** If abnormal spikes are detected but later invalidated (e.g., data entry error), the system retracts the alert and logs the correction. **A3. Notification delivery failure:** If alert cannot be sent to personnel, the system retries and logs the failure for admin review.

UML Use Case Description: Log All Major User Activities

USE CASE #1:	**LOG ALL MAJOR USER ACTIVITIES**
Primary Actor:	System
Goal	The system automatically records significant actions such as logins, record updates, and emergency reports for accountability and security purposes.
Preconditions	• Logging module is enabled.
Postconditions	• All activities are stored in the system audit log.
Trigger	Event-driven (Users interacts with system and logs new activities)
Main Flow	1. The System detects a major event, such as: • User login/logout • Record creation, modification, or deletion • Changes in system settings or permissions 2. The System records the activity details, including: • User who performed the action • Timestamp of the action • Type of action performed 3. The logs are made available for administrator review to ensure accountability and system security.
Alternative Flows	**A1. Logging service unavailable:** If the log service fails, the system temporarily queues events until logging is restored. **A2. Unauthorized action:** If an unpermitted action is attempted, the system blocks the action, logs it, and alerts the administrator. **A3. Corrupted activity record:** If activity details cannot be saved due to data corruption, system logs the error and continues with subsequent events.

ADMIN USE-CASES

ACTOR

UML Use Case Description: Manage Use Accounts

USE CASE #1:	**MANAGE USER ACCOUNTS**
Primary Actor:	Administrator
Goal	To create, modify, or deactivate user accounts for students, clinic staff, SSD, and doctors.
Preconditions	• Administrator is logged into the system. • System access is granted to admin-level users. • A user management module is available.
Postconditions	• User accounts are successfully created, updated, or deactivated. • Account changes are logged in the audit trail.
Trigger	Administrator selects “User Management” from the admin panel.
Main Flow	1. The system displays a list of all registered user accounts. 2. The Administrator selects “Add,” “Edit,” or “Deactivate” for a specific user. 3. The Administrator enters or updates the required user details including: • First and Last Name • Role [Student | Clinic Staff] • University email • Account status [Active | Deactivated] 4. The system validates the inputs and saves the changes. 5. A confirmation message appears, and the update is recorded in the system’s audit trail for accountability.
Alternative Flows	**A1. Invalid Input:** Display “Missing or invalid user details.” **A2. Action Failed:** Display “Unable to update user account. Please retry.” **A3. Mismatched role:** Display “Access denied.” **A4. Unauthorized Access:** Display “You do not have permission to modify these settings.”

UML Use Case Description: Configure System Settings (Security, Access Rules, Audit Trails)

USE CASE #2:	**CONFIGURE SYSTEM SETTINGS (SECURITY, ACCESS RULES, AUDIT TRAILS)**
Primary Actor:	Administrator
Goal	To configure system-level settings including access roles, security parameters, and audit logging.
Preconditions	• Administrator is logged into the system. • A configuration module is available. • Admin has system privileges.
Postconditions	• System configurations are updated and saved. • New access and security settings take effect immediately.
Trigger	Administrator selects “System Configuration” from the admin panel.

Main Flow	1. The system displays configurable options such as security settings, access permissions, and audit parameters.
	2. The Administrator updates the necessary configurations:
	  • Role permissions
	  • Password policies
	  • Audit parameters
	3. The system validates all modified settings.
	4. The Administrator reviews and confirms the changes.
	5. The system applies the new configurations and logs the update in the audit trail.

Alternative Flows	**A1. Unauthorized Access:** Display “You do not have permission to modify these settings.” **A2. Configuration Error:** Display “Failed to apply settings. Please try again.”

UML Use Case Description: Set and Update Gamification Criteria

USE CASE #3:	**SET AND UPDATE GAMIFICATION CIRTERIA**
Primary Actor:	Administrator
Goal	To set, modify, or update gamification rules, such as HP stamp points, event rewards, and participation thresholds.
Preconditions	• Administrator is logged into the system. • A Gamification module is accessible. • Existing reward configurations are available.
Postconditions	• Gamification rules are updated in the system. • Changes are reflected in student app interfaces and dashboards.
Trigger	Administrator selects “Gamification Settings” from the admin panel.

Main Flow	1. The Administrator accesses “Gamification Settings.”
	2. The system displays the current HP stamp, badge, and reward configurations.
	3. The Administrator updates the desired criteria, such as:
	  • Points awarded per daily check-in
	  • Weekly event or milestone rewards
	4. The system validates all modified inputs and saves the updated settings.
	5. The students’ app interfaces automatically reflect the new gamification criteria.

Alternative Flows	**A1. Invalid Configuration:** Display “Invalid reward or point value.” **A2. Save Failed:** Display “Unable to update gamification settings.”

UML Use Case Description: Access Admin Dashboard

USE CASE #4:	**ACCESS ADMIN DASHBOARD**
Primary Actor:	Administrator
Goal	To view overall system analytics, including user activity logs, health statistics, and audit trails.
Preconditions	• Administrator is logged into the system. • Data sources (users, health surveys, activity logs) are available.
Postconditions	• Dashboard metrics are displayed. • Admin may export or review data insights.
Trigger	Administrator selects “Admin Dashboard” from the main menu.

Main Flow	1. The system loads the Admin Dashboard.
	2. The dashboard displays key data, including:
	  • Total user count
	  • System uptime
	  • Recent activity logs
	  • Active modules and components
	3. The Administrator filters or refreshes the displayed reports as needed.
	4. The system updates the dashboard visuals in real-time.
	5. The Administrator may export analytics or logs for record keeping or auditing.

Alternative Flows	**A1. No Data Available:** Display “No data to display at this time.” **A2. Dashboard Error:** Display “Failed to load dashboard data.”

STUDENT USE-CASES

ACTOR

JoseniCare
USE CASE DIAGRAM

UML Use Case Description: Book and View Clinic/Dental Appointments

USE CASE #1:	**BOOK AND VIEW CLINIC/DENTAL APPOINTMENTS**
Primary Actor:	USJ-R Tertiary Students
Goal	To schedule and view clinic or dental appointments conveniently through the JoseniCare app.
Preconditions	• Student is logged into the app. • Clinic system is online and has available appointment slots.
Postconditions	• Appointment is successfully booked and recorded in the system. • Confirmation and details are displayed to the student.
Trigger	Student selects the “Book” option in the application under the Schedule Appointment section.
Main Flow	1. Student opens the JoseniCare app. 2. Student selects the option to “Book” an appointment. 3. System asks which campus the student is located [BASAK | MAIN | QUADRICENNTENIAL]. 4. System asks if the student is booking a dental session or general check-up → [DENTAL | CHECK-UP]. a. If Dental, System will ask follow up questions such as: for [CLEANING | DENTAL FILLING (pasta)] b. If Check-up, System will ask: for [Event (Sportsfest, Intramurals) | General] 5. After student answers the follow-up questions, System will show available and taken slots through visual calendar. 6. Student will select available slot; Specify [Date & time] and enter confirmation. 7. System confirms availability and saves the appointment if true. 8. Student views confirmation details showing date, time, and appointment purpose.
Alternative Flows	**A1. No Available Slot:** System will inform the student and suggest the nearest available schedule instead. **A2. Connection Error:** Alert “Unable to process booking. Please try again later.”

UML Use Case Description: Submit a Daily Health Survey

USE CASE #2:	**SUBMIT A DAILY HEALTH SURVEY**
Primary Actor:	USJ-R Tertiary Students
Goal	To submit daily self-health assessments for wellness tracking and potential outbreak detection.
Preconditions	• Student is logged into the app. • Clinic system is online and survey form is open for the current date.
Postconditions	• Survey data is recorded in the system. • Health status is updated in the Health Dashboard.
Trigger	Student selects the “Check-in” option in the application under the Daily Health Survey section.
Main Flow	1. Student opens the JoseniCare app. 2. Student selects the option to “Check-in” their Daily Health Survey. 3. The system displays the Terms and Conditions before accessing the submission form. 4. The system prompts the student for a subjective physical assessment, asking them to rate their general health from 1 to 5 (Healthy → Sick). i. If the rating is between 3-5, System will ask for temperature, symptoms, severity and duration. ii. If the rating is between 1-2, the system provides a positive message acknowledging the student’s good health and asks them to confirm the absence of common symptoms. 5. After completing the questions, student confirms and submits their survey. 6. System will validate report and then save to database if true. 7. The student’s check-in progress is automatically updated and visually reflected (e.g., via progress stamp or indicator)
Alternative Flows	**A1. Failed Confirmation for Terms and Conditions:** Student will be asked to agree to answer the survey form. **A2. Submission Error:** Display “Failed to submit form. Please try again.” **A3. Report Mismatch:** Display “It seems there’s a small inconsistency. You rated your health as good but also reported some symptoms. Please review your rating to ensure it reflects how you feel.” **A4. Duplicate Submission:** Display “You’ve already completed today’s survey.”

JoseniCare
USE CASE DIAGRAM

UML Use Case Description: View Digital Health Record

USE CASE #3:	**VIEW DIGITAL HEALTH RECORD**
Primary Actor:	USJ-R Tertiary Students
Goal	To access personal medical and dental records digitally.
Preconditions	• Student is logged into the app. • Students' digital health record exists in the system. • Has set their biometric login detail. Either [Fingerprint | Facial] recognition.
Postconditions	• Health data are displayed securely
Trigger	Student selects the "View" option in the application under the Personal Records section for Digital Health Record.

Main Flow	1. The student opens the JoseniCare app and navigates to the "Personal Records" page.
	2. The student selects the option to "View" their Digital Health Record.
	3. The system prompts the student to complete biometric login for verification.
	4. The system retrieves the student's stored medical and dental records from the database.
	5. The system displays the student's patient information as recorded by the clinic, including:
	  • Name
	  • School ID
	  • Department
	  • Course & Year
	  • Noted Allergies
	  • History of previous illnesses or accidents
	6. The system summarizes and visualizes health data sourced from the student's submitted Health Surveys and clinic appointments, such as:
	  • Frequency of specific symptoms (e.g., how many times cough, colds, fever occurred)
	  • Average duration and severity of past conditions
	  • Health trends over time (e.g., improved/stable/worsening)
	  • Prescribed medicines
	7. The student can interact with the visual summary (e.g., view charts or detailed breakdowns for each health category).

Alternative Flows	**A1. Failed Biometric login:** Prompt retry or alternative login. **A2. No Health Record found:** Display "Sorry, you have no digital health record yet. Please inquire the Clinic READS for assistance." **A3. Data retrieval error:** System displays "Unable to load records. Please try again later.

UML Use Case Description: View Appointment History

USE CASE #4:	**VIEW APPOINTMENT HISTORY**
Primary Actor:	USJ-R Tertiary Students
Goal	To view the list and details of past clinic and dental appointments.
Preconditions	• Student is logged into the app. • Students' appointment history record exists in the system.
Postconditions	• Appointment history is displayed securely
Trigger	Student selects the "Appointment History ->" option in the application under the Personal Records section.
Main Flow	1. The student opens the JoseniCare app and navigates to the "Personal Records" page. 2. The student selects the button option "Appointment History ->". 3. The system prompts the student to complete biometric login for verification. 4. The system retrieves the student's stored medical and dental appointment history information from the database, including: • Date & time • Purpose for appointment • Doctor in Charge • Campus Clinic selected 5. The system displays the list of past appointments in chronological order (most recent first). 6. The student may select a specific appointment to view consultation notes or any further updates, such as follow-up check-ups or prescriptions issued by the doctor.
Alternative Flows	**A1. Failed Biometric login:** Prompt retry or alternative login. **A2. No Appointment History found:** Display "You have no recorded clinic appointments yet." **A3. Data retrieval error:** System displays "Unable to load appointment history. Please try again later.