Simple Use Cases (Low Complexity)

Simple use cases involve one main actor, minimal business rules, read-only or single-step operations, and few classes in the class diagram.

---

1. View Attendance History (Student)

Description:

Allows a student to view their past attendance records for enrolled courses.

Why it is Simple:

Single actor (Student)

No data modification

Simple retrieval operation

Limited number of classes (Student, AttendanceRecord, Controller)

No validation logic or multi-step flow

---

2. View Real-Time Attendance (Faculty Member)

Description:

Allows a faculty member to view attendance status during an active session.

Why it is Simple:

Single actor (Faculty Member)

Only displays current attendance data

No complex calculations or workflows

Small class diagram with direct interaction

---

3. View Attendance Policy (Student)

Description:

Enables students to view institutional attendance rules and regulations.

Why it is Simple:

Static information retrieval

No dependencies on other subsystems

No business rules or validations

Minimal classes (Policy, Student, UI)

---

4. Register Biometric Device (FMS Admin)

Description:

Allows an administrator to register a biometric device into the system.

Why it is Simple:

One actor (Admin)

Single operation (store device data)

No interaction with attendance logic

Limited attributes and methods in the class diagram

Moderate Use Cases (Medium Complexity)

Moderate use cases include multiple validation steps, business rules, and coordination between several classes, but without heavy data processing.

---

5. Record Attendance Using QR Code (Student)

Description:

Students scan a QR code to mark their attendance.

Why it is Moderate:

Multiple validations (QR validity, session time, enrollment)

Interaction between several classes (QRToken, Session, Attendance)

One main actor but multiple system checks

Medium-sized class diagram

---

6. Generate Attendance Warning List (Academic Affairs)

Description:

Generates a list of students whose attendance is below the allowed threshold.

Why it is Moderate:

Business rules (attendance percentage thresholds)

Requires calculations and filtering

Uses multiple classes (Student, AttendanceRecord, Rule)

More logic than simple retrieval

---

7. View Attendance Statistics (Faculty Member)

Description:

Displays attendance percentages and summaries for a course.

Why it is Moderate:

Requires aggregation and calculation

Multiple data sources involved

Still limited to one actor

Moderate interaction between entities

8. Manage Attendance Session (Faculty Member)

Description:

Allows faculty to open and close attendance sessions.

Why it is Moderate:

Time-based logic

State changes (open/close session)

Requires validations

Medium-level class interactions

Complex Use Cases (High Complexity)

Complex use cases involve multiple actors, advanced business rules, external systems, data analysis, and large class diagrams.

9. Record Attendance Using Fingerprint

Description:

Students record attendance using biometric fingerprint authentication.

Why it is Complex:

Integration with biometric hardware

Biometric matching algorithms

Multiple actors (Student, Device, System)

Security and accuracy requirements

Large class diagram with controllers, services, and entities

---

10. Analyze Attendance Trends (Academic Affairs)

Description:

Analyzes historical attendance data to identify patterns and trends.

Why it is Complex:

Advanced data analysis

Aggregation across semesters and courses

Requires reporting logic and filtering

Heavy processing and multiple class dependencies

---

11. Generate Low Attendance Report

Description:

Produces official reports identifying students with low attendance.

Why it is Complex:

Cross-course analysis

Institutional rules enforcement

Report formatting and validation

Large number of interacting classes

---

12. Generate End-of-Semester Attendance Report

Description:

Creates a comprehensive attendance summary at the end of the semester.

Why it is Complex:

Combines data from many sessions

Multiple calculations and validations

Formal report generation

High data volume and dependencies