Republic of the Philippines
DAVAO ORIENTAL STATE UNIVERSITY
Guang-guang, Dahican, City of Mati, Davao Oriental
Faculty of Computing, Data Sciences, Engineering and Technology
Information Technology Program

ITC 130 – Applications Development in Emerging Technologies

# PROJECT X: Automated Attendance Tracking System: High Level Use Diagram

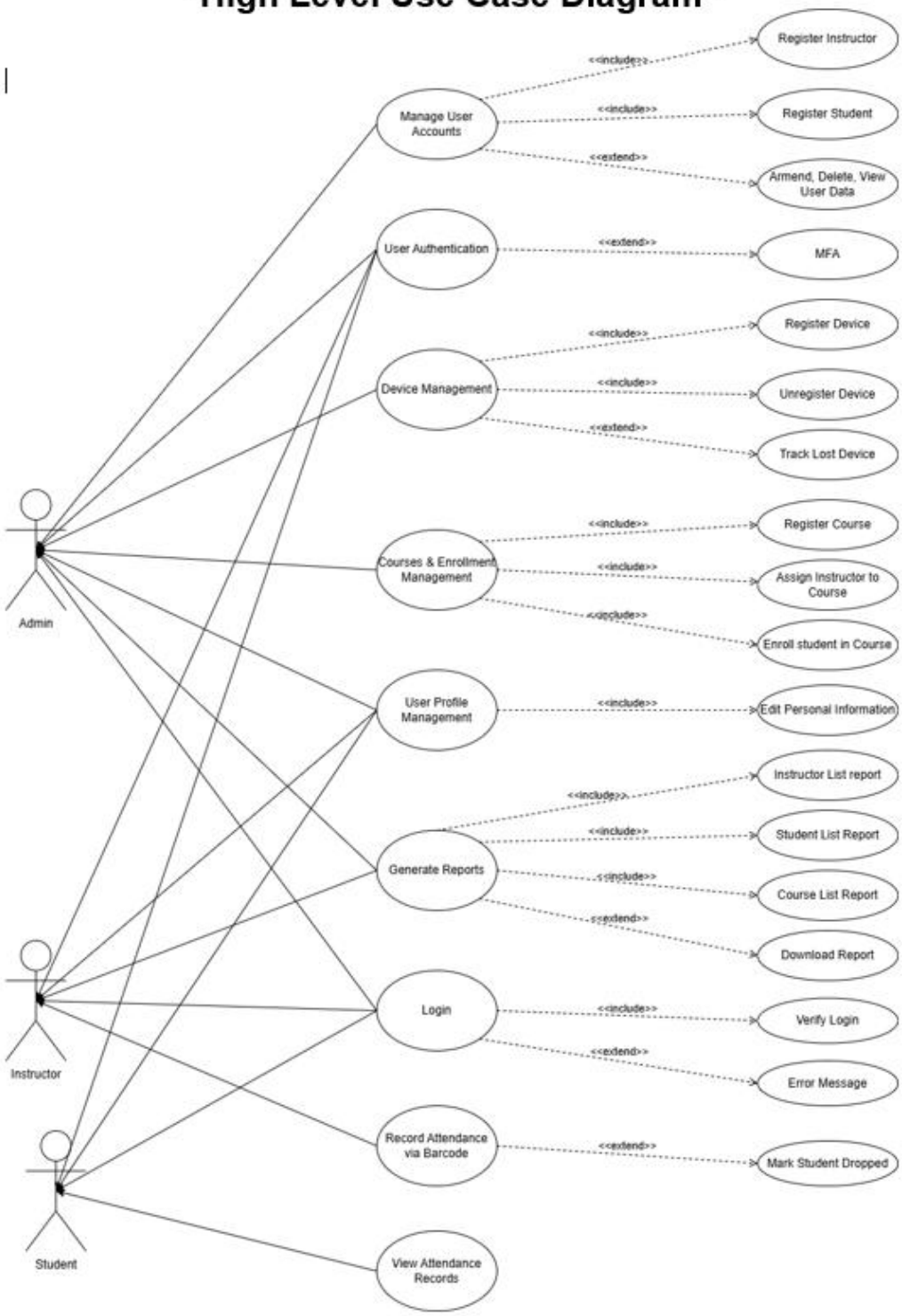
PRESENTED BY:

NEIL ROY G. OMONGOS

WENONA MARIE M. MONTEMAYOR

SHAMAIAH LEE CADUT

# "High Level Use Case Diagram"



## Overview

The **High-Level Use Case Diagram** represents the main functions (use cases) of the **Automated Attendance Tracking System** and shows how different types of users (actors) interact with the system. It focuses on the key roles: **Admin**, **Instructor**, and **Student**, each having specific responsibilities and access rights.

## Actors and Their Roles

1. **Admin**

   The Admin has the most control in the system. They manage users, courses, devices, and reports. They also handle system-level actions like user authentication and data updates.
2. **Instructor**

   Instructors can record attendance, view reports, and interact with student data relevant to their assigned courses.
3. **Student**

   Students have limited access. Their main actions are logging in and viewing their own attendance records.

## Main Use Cases and Descriptions

1. **Manage User Accounts**

   - *Includes:* Register Instructor, Register Student
   - *Extends:* Amend, Delete, View User Data
   - The Admin creates new accounts and can update or delete existing user information.

2. **User Authentication**

   - *Extends:* MFA (Multi-Factor Authentication)
   - Ensures that all users (Admin, Instructor, Student) securely log in using MFA for added security.

3. **Device Management**

   - *Includes:* Register Device
   - *Extends:* Unregister Device, Track Lost Device
   - Admin manages the devices used for attendance tracking. Only registered devices can be used for scanning.

4. **Courses and Enrollment Management**

   - *Includes:* Register Course, Assign Instructor to Course, Enroll Student in Course
   - Admin creates course records, assigns instructors, and enrolls students in the right subjects.

5. **User Profile Management**

   - *Includes:* Edit Personal Information, Instructor List Report
   - Users can update their own information, and Admins can view instructor-related reports.

6. **Generate Reports**

   - *Includes:* Instructor List Report, Student List Report, Course List Report
   - *Extends:* Download Report
   - Admin and Instructor can generate and download reports for monitoring attendance and managing course data.

7. **Login**

   - *Includes:* Verify Login
   - *Extends:* Error Message
   - All users must log in to access the system. The system verifies credentials and shows error messages for invalid login attempts.

8. **Record Attendance via Barcode**

- *Extends:* Mark Student Dropped
- Instructors scan student barcodes to record attendance. If a student has 3 consecutive absences, the system may mark them as dropped.

**9. View Attendance Records**

- by students so they can view their own attendance history.

**Real-World Example**

Let's say **Instructor A** opens the app and scans the QR code shown by **Student B**. The system:

- Verifies that the device is registered
- Matches the student to the course and instructor
- Logs the attendance with time and date
  If **Student B** misses three classes in a row, the system marks them as "Dropped" automatically.

Meanwhile, the **Admin** can download attendance reports for all students in that course and update student details if needed.

**Conclusion**

This High-Level Use Case Diagram provides a clear view of the **main operations** and **user interactions** in the system. It ensures that:

- **Admins** manage the system structure
- **Instructors** handle attendance and teaching
- **Students** track their own records

It also emphasizes **security** through features like device registration and multi-factor authentication, making the system **efficient**, **secure**, and **user-friendly**.