



# **LOW LEVEL DESIGN AND IMPLEMENTATION DOCUMENT**

## **Smart Classroom Environment**

**UE18CS390B – Capstone Project Phase – 2**

*Submitted by:*

<b>Naik Bhavan Chandrashekhar</b>	<b>PES2201800047</b>
<b>Akshaya Visvanathan</b>	<b>PES2201800089</b>
<b>Atharva Moghe</b>	<b>PES2201800131</b>
<b>Akhil S Kumar</b>	<b>PES2201800137</b>

Under the guidance of:

**Dr. Annapurna D**  
Professor, CSE  
PES University

**August - December 2021**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**  
**FACULTY OF ENGINEERING**  
**PES UNIVERSITY**

(Established under Karnataka Act No. 16 of 2013)

Electronic City, Bengaluru – 560 100, Karnataka, India

**TABLE OF CONTENTS**

1. Introduction	3
1.1. Overview	3
1.2. Purpose	3
1.3. Scope	3
2. Design Considerations, Assumptions and Dependencies	3
3. Design Description	4
3.1. Master-Class Diagram	4
3.2. Attendance System	4
3.2.1 Description	4
3.2.2 Use-Case Diagram	5
3.2.3 Class Diagram	6
3.2.4 Sequence Diagram	11
3.3. Energy Conservation System	11
3.3.1 Description	11
3.3.2 Use-Case Diagram	12
3.3.3 Class Diagram	13
3.3.4 Sequence Diagram	15
3.4. Packaging and Deployment Diagram	16
4. Proposed Methodology/Approach	17
5. Further Exploration Plans and Timelines	18
Appendix A: Definitions, Acronyms and Abbreviations	18
Appendix B: References	19
Appendix C: Record of Change History	19
Appendix D: Traceability Matrix	20

**Note:**

<b>Section 1</b>	<b>Common for Prototype/Product Based and Research Projects</b>
<b>Section 2 &amp; 3</b>	<b>Applicable for Prototype / Product Based Projects.</b>
<b>Section 4</b>	<b>Applicable for Research Projects.</b>
<b>Appendix</b>	<b>Provide details appropriately</b>

### 1. Introduction

#### 1.1. Overview

The Low-Level Design Document highlights the concept and necessary detail for implementing an IoT-based enhanced learning environment. A combination of use case, class and sequence diagrams provides a detailed idea behind the product.

#### 1.2. Purpose

These diagrams show the relationship between the users and the admins and the cloud platform(system) and the interaction between them. It also gives a great idea about our project.

#### 1.3. Scope

We are planning to firstly implement this project in our own campus and then extend it to other institutions.

### 2. Design Constraints, Assumptions, and Dependencies

#### - Server Systems in the Institutions:

Our project, to an extent, depends upon how the server system exists at a particular institution.

#### - Existing Wiring in Institutions:

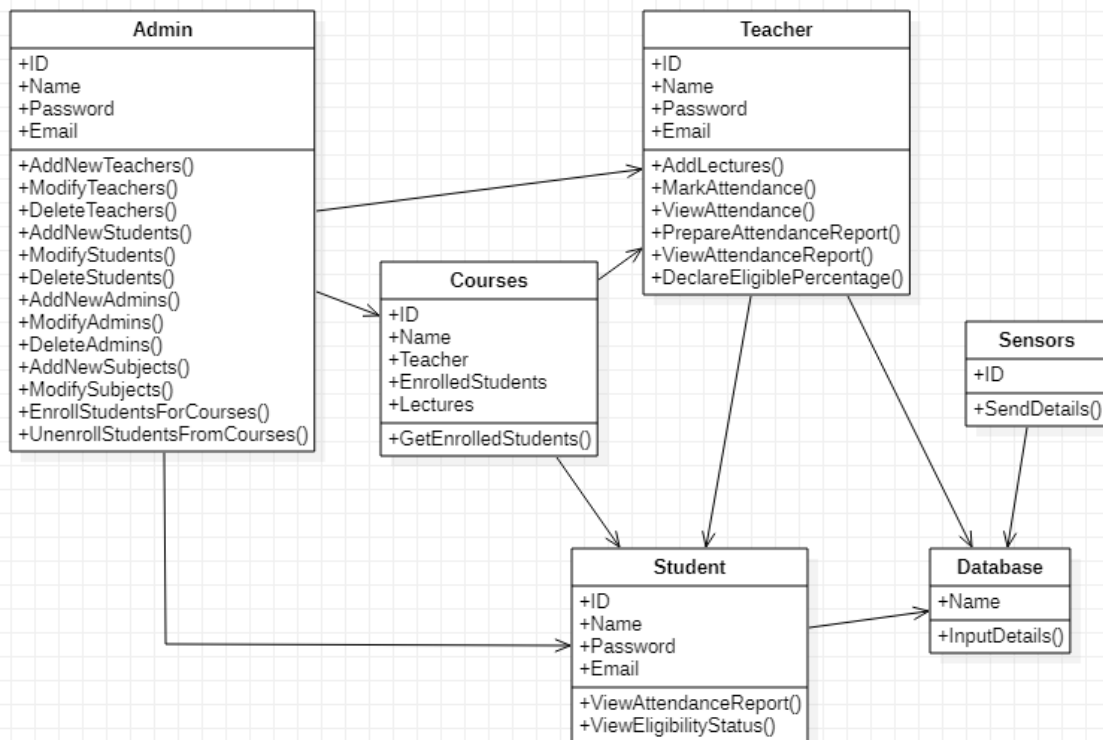
It depends which wiring system exists at the institution because our project would work on almost every existing wiring system unless it's very old.

#### - Authenticity of Collected Data:

Due to the present scenario, we must fabricate the data required in our project.

### 3. Design Description

#### 3.1. Master Class Diagram

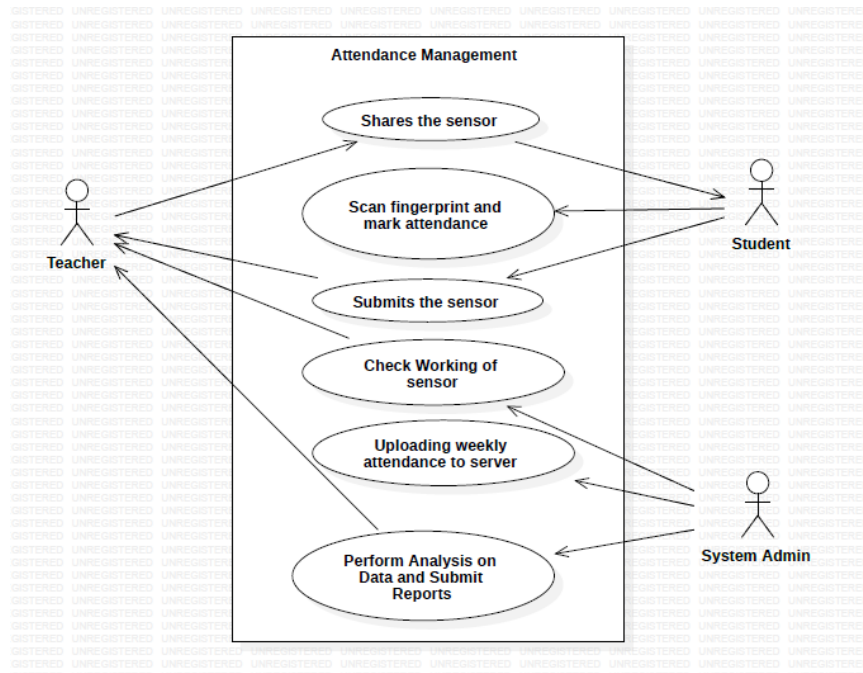


#### 3.2. Attendance System

##### 3.2.1. Description

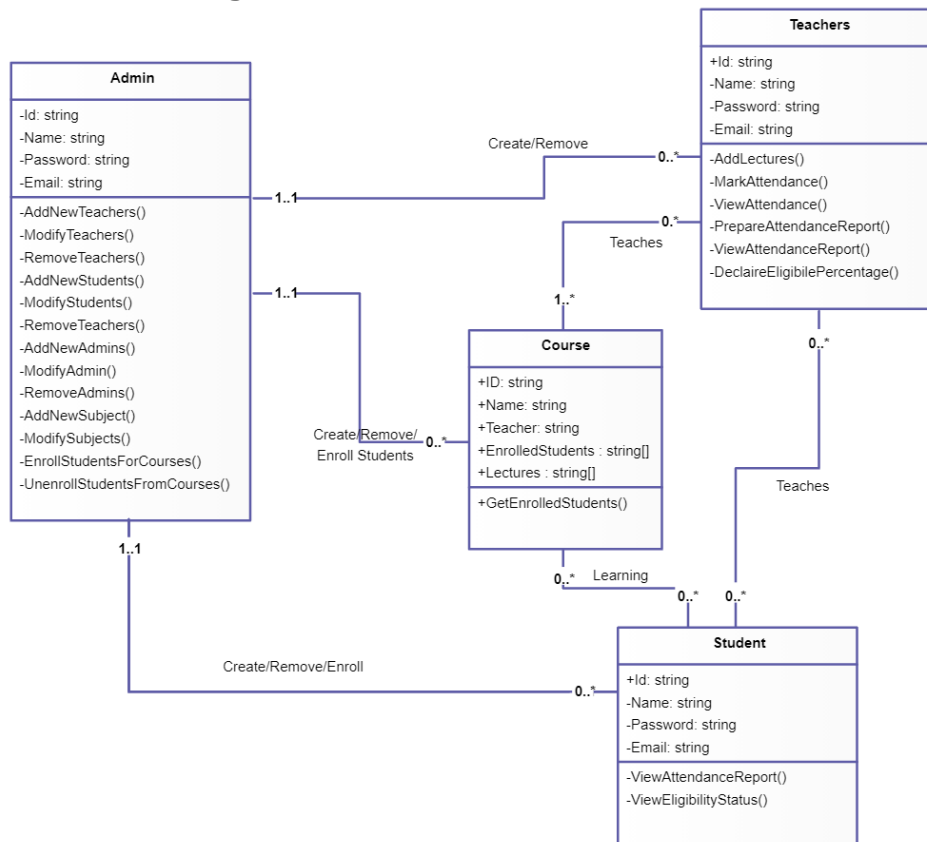
The first module of our project focuses on creating a semi-automatic attendance system rather than the regular manual method of teachers calling out names and the students responding to it. We felt that this manual method wasted lots of time during class hours thereby being a concern for both students and teachers in this teaching-learning process. We came up with a foolproof fingerprint scanner method to save time and discrepancy for the same. The complete approach to this is mentioned in Section-4.

### 3.2.2. Use Case Diagram



Use Case Item	Description
Check sensor working	The system admin checks the working of the fingerprint scanner and assigns one scanner to one teacher or one class.
Shares the sensor	The teacher enters the classroom, then activates and shares the fingerprint scanner with the students.
Scan the fingerprint and mark attendance	The students scan their respective fingerprints and the attendance for that class is marked as "Present".
Submit the sensor	The students submit the sensor back to the teacher after everyone has completed the process.
Uploading weekly attendance to server	The attendance data is uploaded weekly to the institution server.
Perform Data Analysis and Submit Reports	System Admin performs Data Analysis on the uploaded data and finally sends a complete report to the teacher regarding student's attendance.

### 3.2.3. Class Diagram



#### 3.2.3.1. Admin

##### 3.2.3.1.1. Class Description

This class is for the set of all system admins whose major role is to conduct this attendance process very smoothly. They basically carry out all the backend functions required here.

##### 3.2.3.1.2. Data members

Data Type	Data Name	Access Modifiers	Initial Value	Description
String	ID	Public	-	The ID assigned to the admin.
String	Name	Public	-	The Name of the admin.
String	Password	Private	-	The login password of the admin.
String	E-mail	Private	-	The mail-ID of the admin for login.

**3.2.3.1.3. Add New Teachers**

This function adds new teachers to the institution database on their successful enrollment. We input all the different details of the teachers into the database and the database stores it for future uses.

**3.2.3.1.4. Modify Teachers**

This function modifies the existing values to new values for a particular teacher. We input the value that needs to be changed along with the new value and the changes are then reflected on the institution database.

**3.2.3.1.5. Remove Teachers**

This function deletes the data related to a teacher who has left the institution. We input the ID and Name of that teacher and their details are cleared from the database.

**3.2.3.1.6. Add New Students**

This function adds new students to the institution database on their successful enrollment. We input all the different details of the students into the database and the database stores it for future uses.

**3.2.3.1.7. Modify Students**

This function modifies the existing values to new values for a particular student. We input the value that needs to be changed along with the new value and the changes are then reflected on the institution database.

**3.2.3.1.8. Remove Students**

This function deletes the data related to a student who has left the institution. We input the ID and Name of that student and their details are cleared from the database.

**3.2.3.1.9. Add New Admins**

This function adds newly appointed admins to the institution database on their successful enrollment. We input all the different details of the admins into the database and the database stores it for future uses.

**3.2.3.1.10. Modify Admins**

This function modifies the existing values to new values for a particular admin. We input the value that needs to be changed along with the new value and the changes are then reflected on the institution database.

#### **3.2.3.1.11. Remove Admins**

This function deletes the data related to an admin who has left the institution. We input the ID and Name of that admin and their details are cleared from the database.

#### **3.2.3.1.12. Add New Subjects**

This function adds newly added subject of the curriculum to the institution database. We input all the details regarding the new subject to the database and it is stored for future uses.

#### **3.2.3.1.13. Modify Subjects**

This function is used to modify a part of existing subject to some new values. We input the value that needs to be changed along with the new value and the changes are then reflected in the database.

#### **3.2.3.1.14. Enroll Students from Course**

This function is used to enroll a student for a particular course on their successful registration and eligibility. We input Student's ID, Name and Subject ID and Name and the values are stored in the database.

#### **3.2.3.1.15. Unenroll Students from Course**

This function is used to unenroll the student from an existing course they chose. We input the required values and that row is removed from the existing database.

### **3.2.3.2. Teachers**

#### **3.2.3.2.1. Class Description**

The teacher is given dashboard permissions to view and manage the attendance and given regular updates as to when the students come to class. The teacher is also given notifications in case of any outliers and can also edit attendance as and when necessary.

#### **3.2.3.2.2. Data Members**

<b>Data Type</b>	<b>Data Name</b>	<b>Access Modifiers</b>	<b>Initial Value</b>	<b>Description</b>
String	ID	Public	-	The ID assigned to the teacher.
String	Name	Public	-	The Name of the teacher.
String	Password	Private	-	The login password of the teacher.
String	E-mail	Private	-	The mail-ID of the teacher for login.



**3.2.3.2.3. Add Lectures**

This function adds lectures assigned for that teacher as per the institution schedule or timetable. The teacher can then refer to it to prepare for those lessons.

**3.2.3.2.4. Mark Attendance**

This function marks the teacher's attendance since they will scan their fingerprints first to assign themselves "Present".

**3.2.3.2.5. View Attendance**

This function allows the teacher to view the attendance of any student assigned to them. All the teachers would have to do is input the student's name and subjects, and they would get all the details.

**3.2.3.2.6. Prepare Attendance Report**

This function allows the teacher to form a report regarding a particular student's attendance which can be used for future uses.

**3.2.3.2.7. View Attendance Report**

This function allows the teacher to view the attendance report that they had originally created. It would be present in the institution database.

**3.2.3.2.8. Declare Eligible Percentage**

This function is assigned to the teacher if they want to increase or decrease the eligible percentage for attendance for their classes. The default value for this would be 85%.

**3.2.3.3. Students**

**3.2.3.3.1. Class Description**

The students have permissions to only log in the attendance via fingerprint under the teacher's discretion when the teacher so allows it. The student has no further permissions and if he/she wishes to edit her attendance due to any issue/manual logging, they can only approach the teacher and the teacher can do so under her discretion.

### 3.2.3.3.2. Data Members

Data Type	Data Name	Access Modifiers	Initial Value	Description
String	ID	Public	-	The ID assigned to the student.
String	Name	Public	-	The Name of the student.
String	Password	Private	-	The login password of the student.
String	E-mail	Private	-	The mail-ID of the student for login.

### 3.2.3.3.3. View Attendance Report

This function allows the student to view their own attendance report. They can even select a particular subject to know their attendance status in that subject.

### 3.2.3.3.4. View Eligibility Status

This function allows the student to view the eligibility attendance percentage for that subject set by the teacher.

## 3.2.3.4. Course

### 3.2.3.4.1. Class Description

This class refers to the set of all courses offered by the institution to their enrolled students.

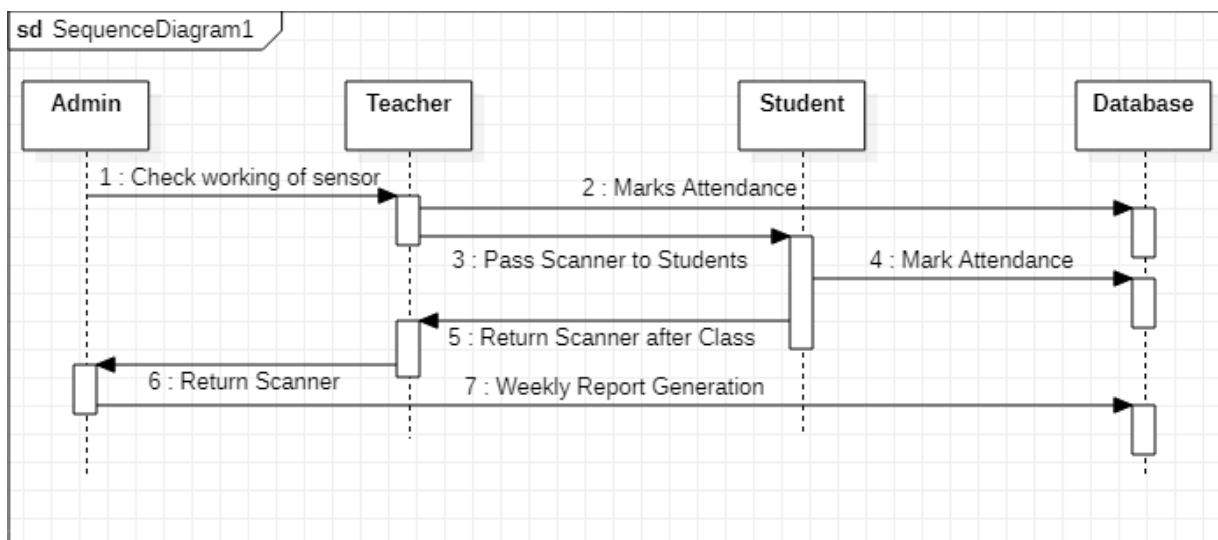
### 3.2.3.4.2. Data Members

Data Type	Data Name	Access Modifiers	Initial Value	Description
String	ID	Public	-	The ID assigned to the course.
String	Name	Public	-	The Name of the course.
String	Teacher	Public	-	The teacher teaching that course.
String []	Enrolled Students	Public	[ ]	The set of students enrolled for course.
String []	Lectures	Public	[ ]	The set of lectures for this course

### 3.2.3.4.3. Get Enrolled Students

This function returns the set of all students who are enrolled for that course. We input the year and the course name, and then the set of all students enrolled for that course pops up on our screen as output.

### 3.2.4 Sequence Diagram

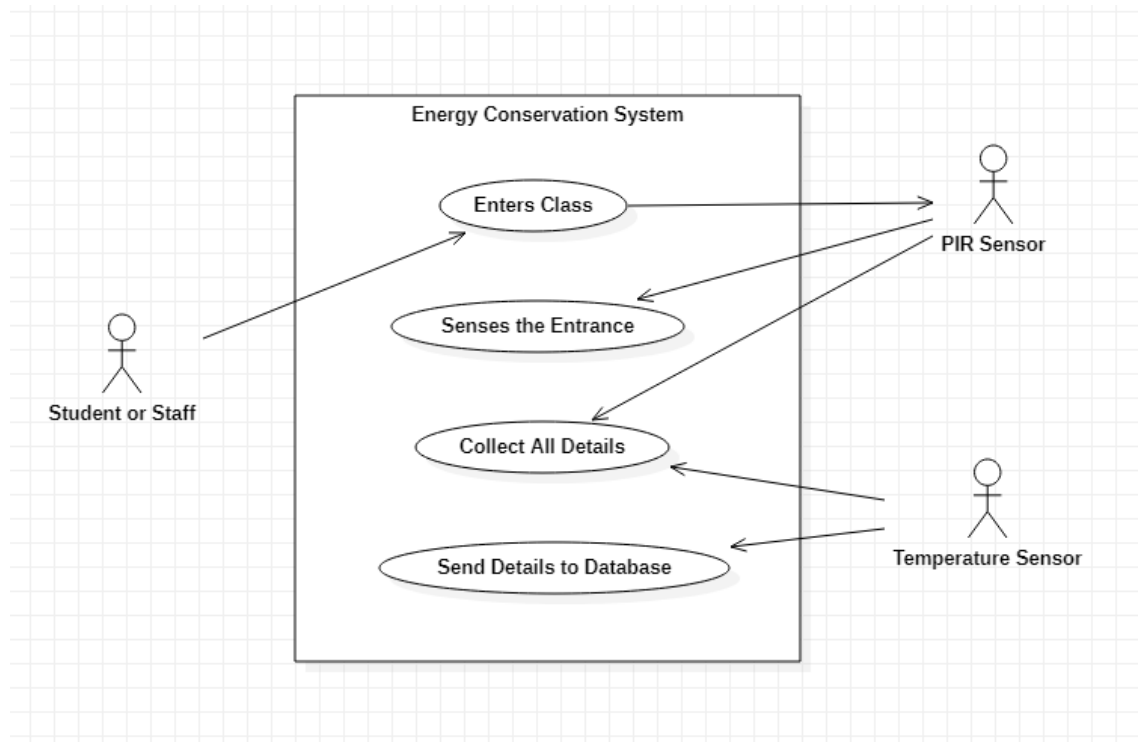


## 3.3. Energy Conservation System

### 3.3.1. Description

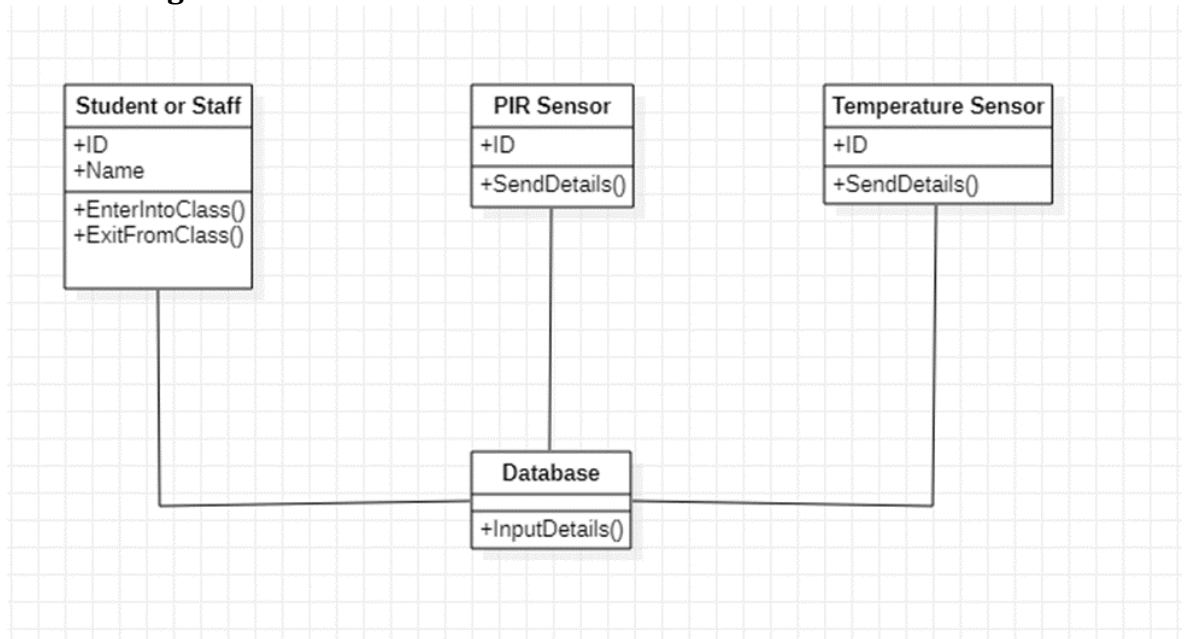
The second module of our project focuses on creating an automatic energy conservation system. We observed that a lot of classes had fans and lights left switched ON even after the classes were over and it took long for a single person to switch it off manually. Hence, we decided to make an automated system for the same. The complete approach to this is mentioned in Section-4.

### 3.3.2. Use Case Diagram



Use Case Item	Description
Enters Class	The Student/Staff enters the classroom.
Senses the Entrance	The PIR Sensor senses the Student/Staff entrance and then sends signal.
Collect All Details	The Sensors collects all the different details available in the classroom.
Send Details to Database	The Sensors then sends all these collected details to the database.

### 3.3.3. Class Diagram



#### 3.3.3.1. Student or Staff

##### 3.3.3.1.1. Class Description

This class is assigned for the set of all people who enter the classroom. They might be students or any of the members of the staff as well.

##### 3.3.3.1.2. Data members

Data Type	Data Name	Access Modifiers	Initial Value	Description
String	ID	Public	-	The ID assigned to the person.
String	Name	Public	-	The Name of the person.

##### 3.3.3.1.3. Enter Class

This function is for the students or staff members when they enter the classroom, and the sensors are activated.

##### 3.3.3.1.4. Exit From Class

This function is for the students or staff members when they exit the classroom, and the sensors are then deactivated.

### **3.3.3.2. PIR Sensor**

#### **3.3.3.2.1. Class Description**

The PIR Sensor sensors the student or staff member walking into the classroom and sends a signal to the temperature and humidity sensor. It then sends the data to the database.

#### **3.3.3.2.2. Data Members**

<b>Data Type</b>	<b>Data Name</b>	<b>Access Modifiers</b>	<b>Initial Value</b>	<b>Description</b>
String	ID	Public	-	The ID assigned to the PIR Sensor.

#### **3.3.3.2.3. Send Details**

This function is assigned to the PIR sensor which sends a signal to the other sensors and sends the data to the institution database.

### **3.3.3.3. Temperature and Humidity Sensor**

#### **3.3.3.3.1. Class Description**

The temperature and humidity sensor receives a signal from the PIR sensor and then sends the temperature and humidity values to the database.

#### **3.3.3.3.2. Data Members**

<b>Data Type</b>	<b>Data Name</b>	<b>Access Modifiers</b>	<b>Initial Value</b>	<b>Description</b>
String	ID	Public	-	The ID assigned to the Sensor.

#### **3.3.3.3.3. Send Details**

This function is assigned to the Temperature and Humidity sensor which sends a signal and data to the institution database.

### 3.3.3.4. Database

#### 3.3.3.4.1. Class Description

This class refers to the database where all the institution data is stored.

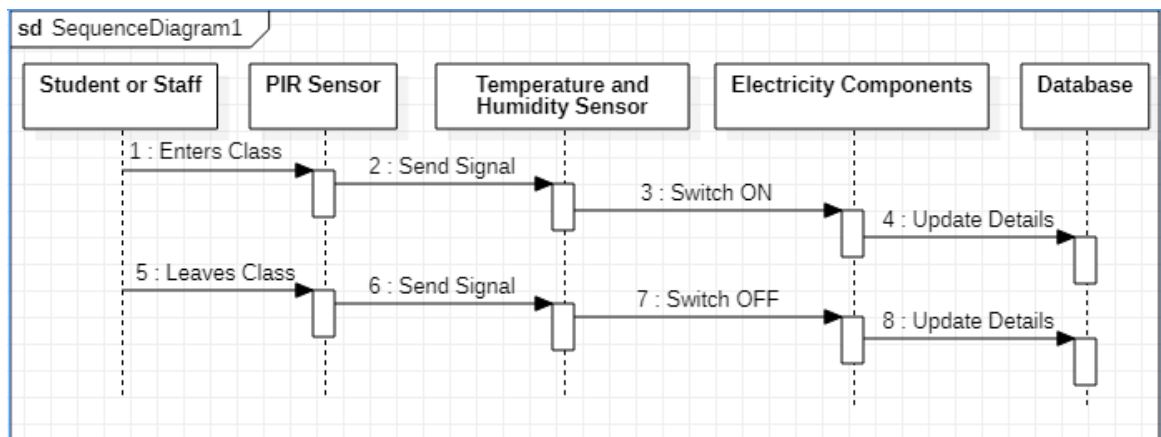
#### 3.3.3.4.2. Data Members

Data Type	Data Name	Access Modifiers	Initial Value	Description
String	Name	Public	-	The Name of the Database.

#### 3.3.3.4.3. Input Details

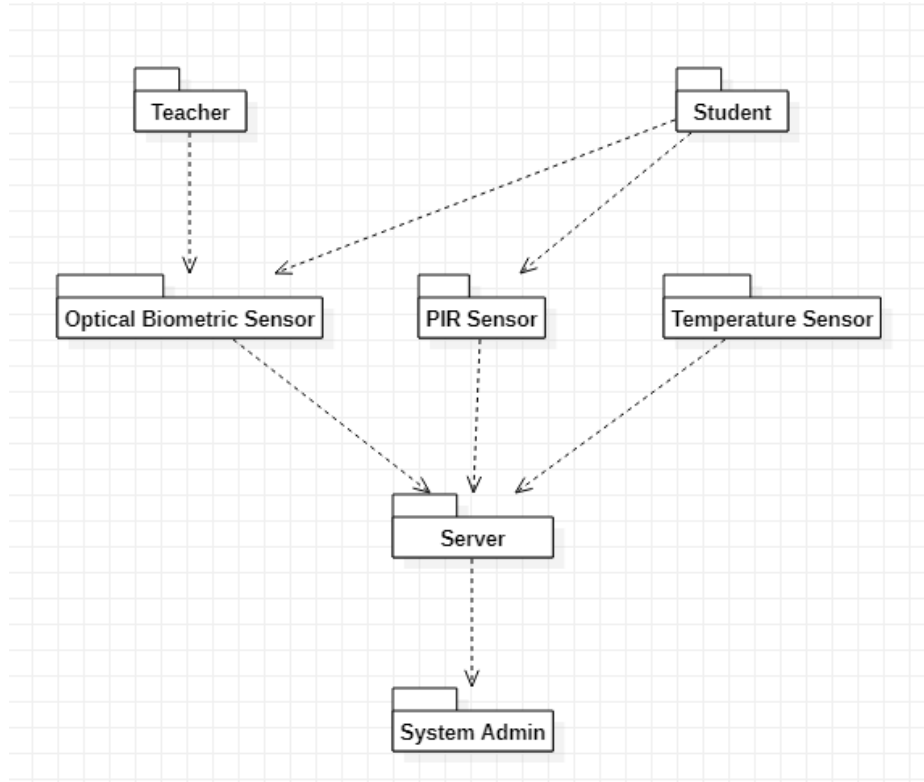
This function inputs the final details sent by all the different sensors into the institutions database.

### 3.3.4. Sequence Diagram

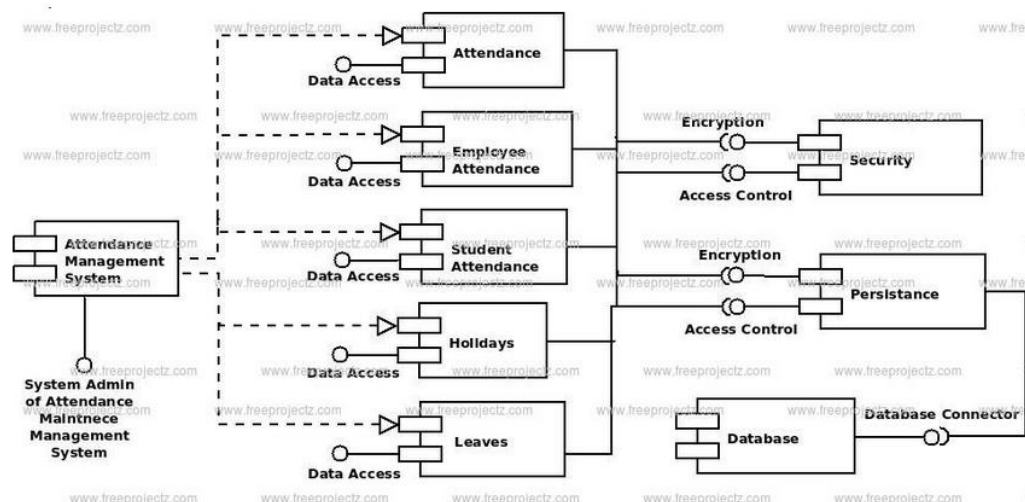


### 3.4. Packaging and Deployment Diagrams

- Packaging Diagram:

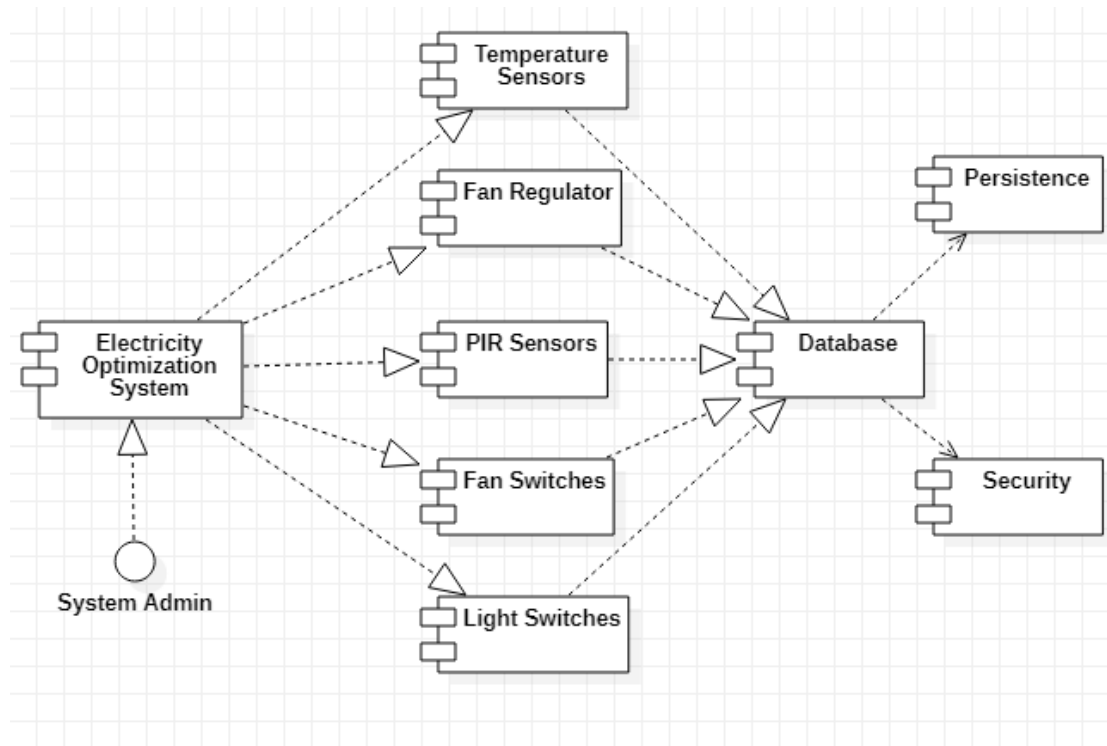


- Deployment Diagram (Attendance System)





### - Deployment Diagram (Energy Conservation System)



## 4. Proposed Methodology / Approach

### - Attendance System:

- Attendance can be taken electronically by means of a biometric optical fingerprint scanner.
- Security and integrity can be ensured by making the biometric module portable and modular - a small phone sized module carried by the teachers.
- The teacher can pass around/have each of the students scan their prints and register their attendance with no manual intervention.

### - Energy Conservation System:

- Spatial sensors placed at the edges of classrooms will notify the system of movement and activity in the room.
- Edge computed algorithms ensure that the lights and fans are turned on only at specific portions of the room in case of a large classroom/hallway.
- In case of manual fans, temperature monitors are used to add a level of cost-effective automation.

**5. Timeline:**

TASKS	1	2	3	4	5	6	7	8	9	10	11	12
Literature Survey												
Propose Architecture												
Desired Changes												
Assembling Sensors												
Attendance System Working												
Field Testing -I												
Electricity Saving Working												
Field Testing -II												
Integration with cloud server												
AI Model Build and Analysis												
Final Testing												
Demo of the Project												
Report Preparation												

**Appendix A: Definitions, Acronyms and Abbreviations**

- Capacitive Fingerprint Scanner: uses capacitance to gauge the depth of the finger and collect the fingerprint.
- Relay: A relay is an electrically operated switch. It consists of a set of input terminals for a single or multiple control signals, and a set of operating contact terminals.
- Thermostat: A thermostat is a regulating device component which senses the temperature of a physical system and performs actions so that the system's temperature is maintained near a desired setpoint.
- Microcontroller: A microcontroller is a small computer on a single metal oxide-semiconductor integrated circuit chip.

## **Appendix B: References**

- M. Cață, "Smart university, a new concept in the Internet of Things," 2015 14th RoEduNet International Conference - Networking in Education and Research (RoEduNet NER), 2015, pp. 195-197, doi: 10.1109/RoEduNet.2015.7311993.
- Q. Miao, F. Xiao, H. Huang, L. Sun and R. Wang, "Smart attendance system based on frequency distribution algorithm with passive RFID tags," in Tsinghua Science and Technology, vol. 25, no. 2, pp. 217-226, April 2020, doi: 10.26599/TST.2018.9010141.
- Suresh S., H. N. S. Anusha, T. Rajath, P. Soundarya and S. V. P. Vudatha, "Automatic lighting and Control System For Classroom," 2016 International Conference on ICT in Business Industry & Government (ICTBIG), 2016, pp. 1-6, doi: 10.1109/ICTBIG.2016.7892666.
- S. Bhattacharya, G. S. Nainala, P. Das and A. Routray, "Smart Attendance Monitoring System (SAMS): A Face Recognition Based Attendance System for Classroom Environment," 2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT), 2018, pp. 358-360, doi: 10.1109/ICALT.2018.00090.

## **Appendix C: Record of Change History**

#	Date	Document Version No.	Change Description	Reason for Change
1.	15/09/2021	1	First Draft	First Version
2.	16/09/2021	2	Diagrams	Added Diagrams
3.	17/09/2021	3	Index	Updated Index

**Appendix D: Traceability Matrix**

<b>Project Requirement Specification Reference Section No. and Name.</b>	<b>DESIGN / HLD Reference Section No. and Name.</b>	<b>LLD Reference Section No. Name</b>
3.4: General Constraints, Assumptions and Dependencies	3.3: Constraints, Assumptions and Dependencies	2: Constraints, Assumptions and Dependencies