Employee Attendance System
Group 17

Team Members

Aishwarya Ajay Venkatesha(E425P738)
Kotur Guna Pragna(S987P265)
Dennis Pham (T353F647)

1. Requirements workflow

A Requirements workflow is a state transition model in which you specify the state that artifacts of a specific type can be in, as well as the actions that users can take to transfer them from one state to another.

Use case diagram

A use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system in the Unified Modeling Language (UML). You'll need a collection of specialized symbols and connectors to construct one. A good use case diagram can assist your team in discussing and representing:

Interactions between your system or application and people, organizations, or external systems and Goals that your system or application assists those entities in achieving

Actors: Users who interact with a system are referred to as actors. A person, an organization, or an external system that interacts with your application.

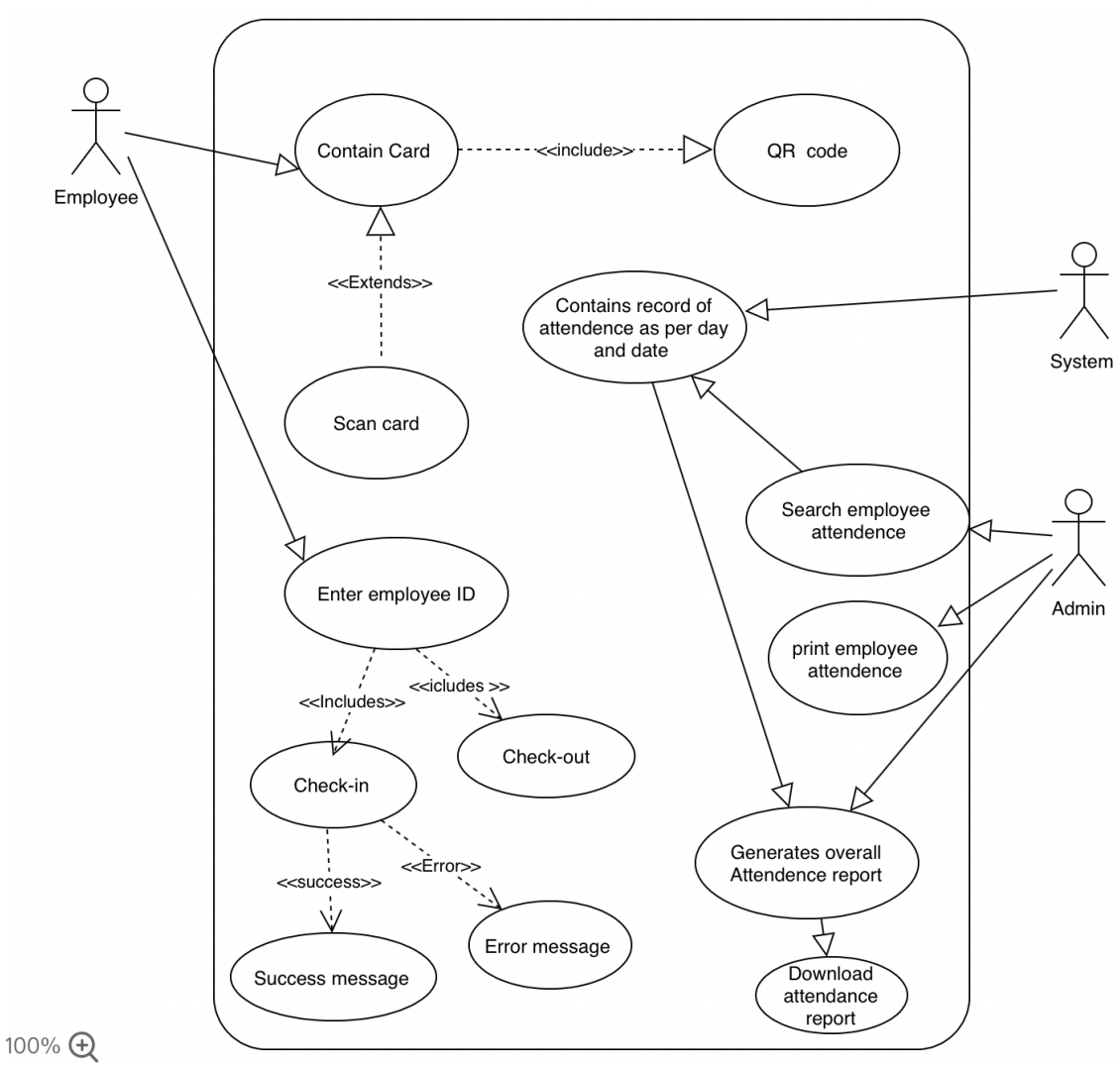
Here we have Employee, System and Admin as Actors.

System: A precise sequence of activities and interactions between actors and the system is referred to as a system. <<include>>, <<extend>>, System boundary, Data flow arrow

We use the sequence of use cases here for scanning the employee ID and using the Id by the Admin for keeping the timesheet and the report of the attendance entry.

Goals: The most common use case's final outcome. The activities and alternatives utilized to achieve the goal should be described in an effective diagram.

Here the final output or the goal includes the Generating the overall attendance report.



100%

Use case description

In this diagram there are two actors and one system is involve

- Employee
- Admin
- System

Here we have the use cases for Employee which has

1) Scan the QR code and Scan the Employee ID card
2) Enter employee id for the check-in and check-out. If check-in processes correctly then the success msg is processed or else the error message is produced.

The Admin use cases include:

- Search employee attendance
- Print employee attendance
- Generates overall attendance report
- Download attendance report

The System use case includes the records as per the date, time and day.

2. Analysis workflow

2.1 Structured System Analysis

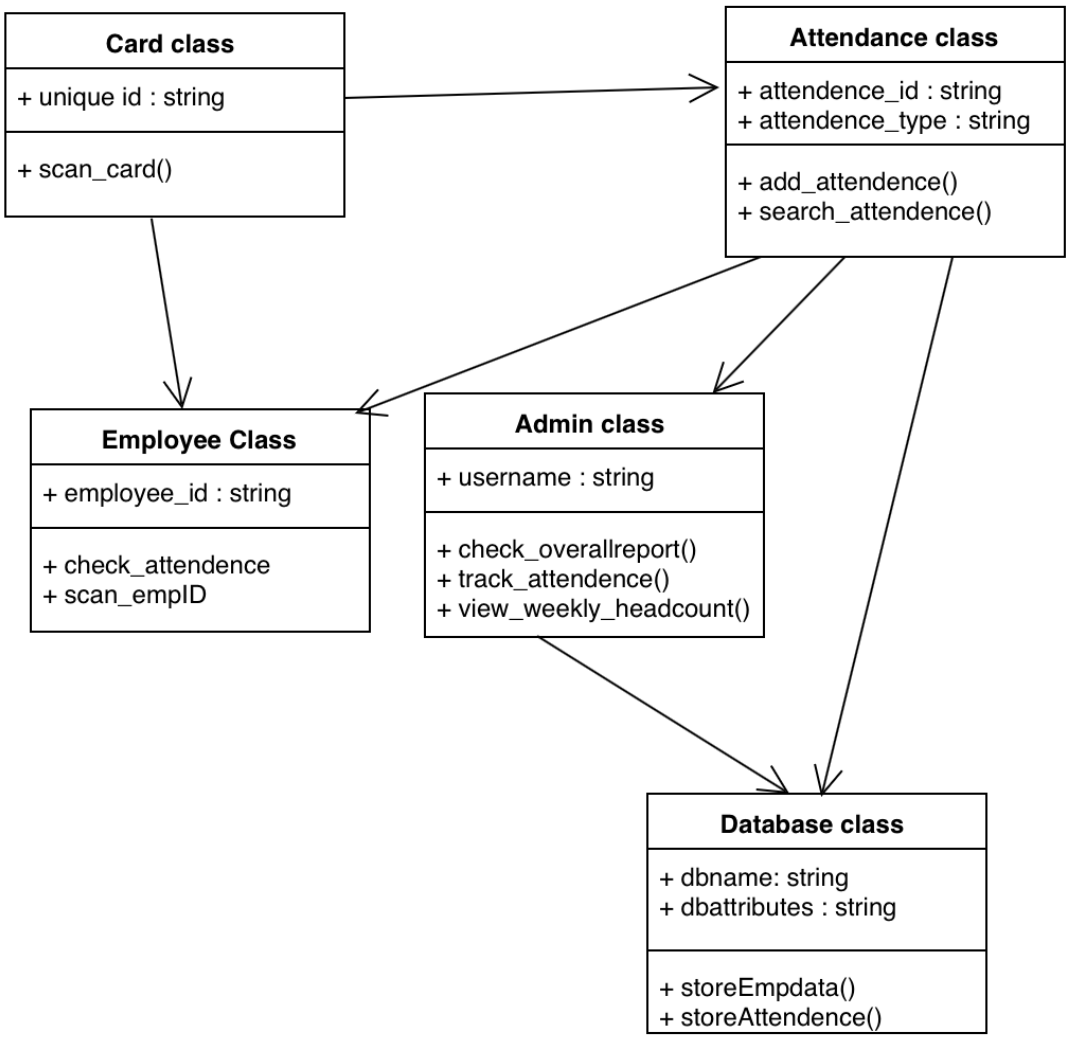
Data Flow diagram

OR

2.2 Object-Oriented Analysis

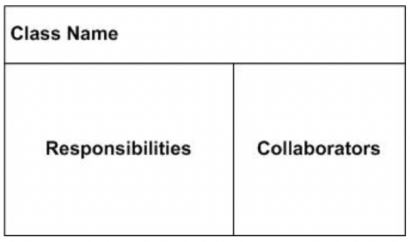
The system requirements are specified, the classes are recognized, and the links between classes are identified during the system analysis or object-oriented analysis phase of software development. The object-oriented analysis is formed with the principles such as information domain and the representation of behaviour and to expose more detail, data, functional, and behavioral models are split. The essence of the problem is represented by early models, while implementation details are provided.

Class diagram



CRC cards

A Class Responsibility Collaborator (CRC) model is made up of three portions segregated from normal index cards, as shown in the diagrams below. A duty is anything that a class understands or does, and a collaborator is another class with which a class interacts to fulfill its responsibilities.

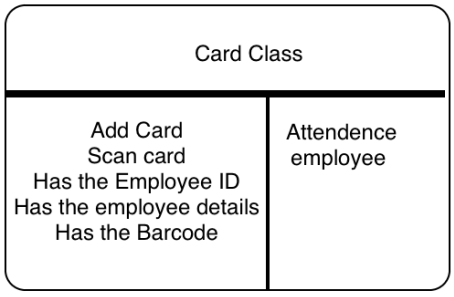


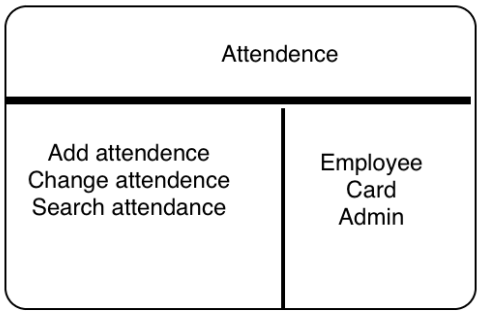
Class: A class is a group of things that are comparable in some way. A person, place, thing, event, or notion that is significant to the system at hand is referred to as an object.

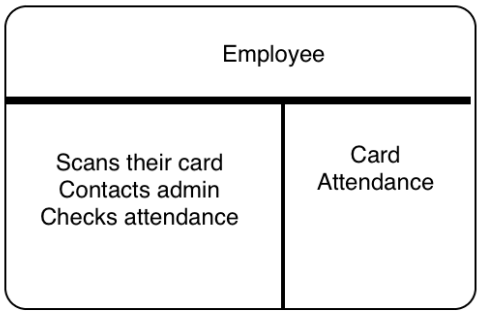
Classes, in our project, would represent Card, Attendance, Database, Admin and Employee in a university system.

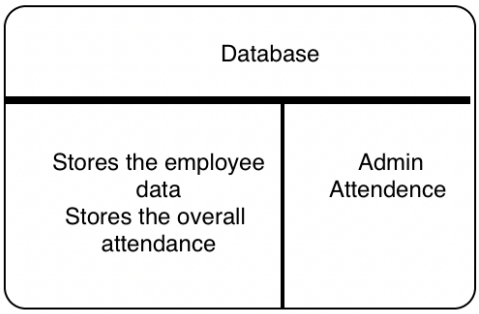
Responsibility : Anything that a class knows or performs is considered a responsibility. Students, for instance, have scan cards, employee details , and Barcode . These are the items that a Card class is aware of. Employees can also use this card for attendance and scanning barcodes for attendance. These are the activities of an Employee. A Card class duties are defined by what it knows and does.

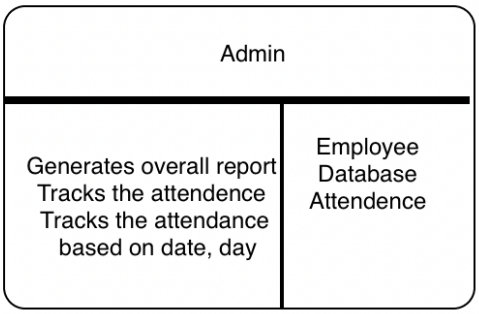
Collaborators : A request for information or a request to accomplish something can be used to initiate collaboration. For instance, consider the card. A request for information is made by the Admin to the attendance report , asking if an Employee ID is available. The Admin checks the timesheet where the attendance is recorded.









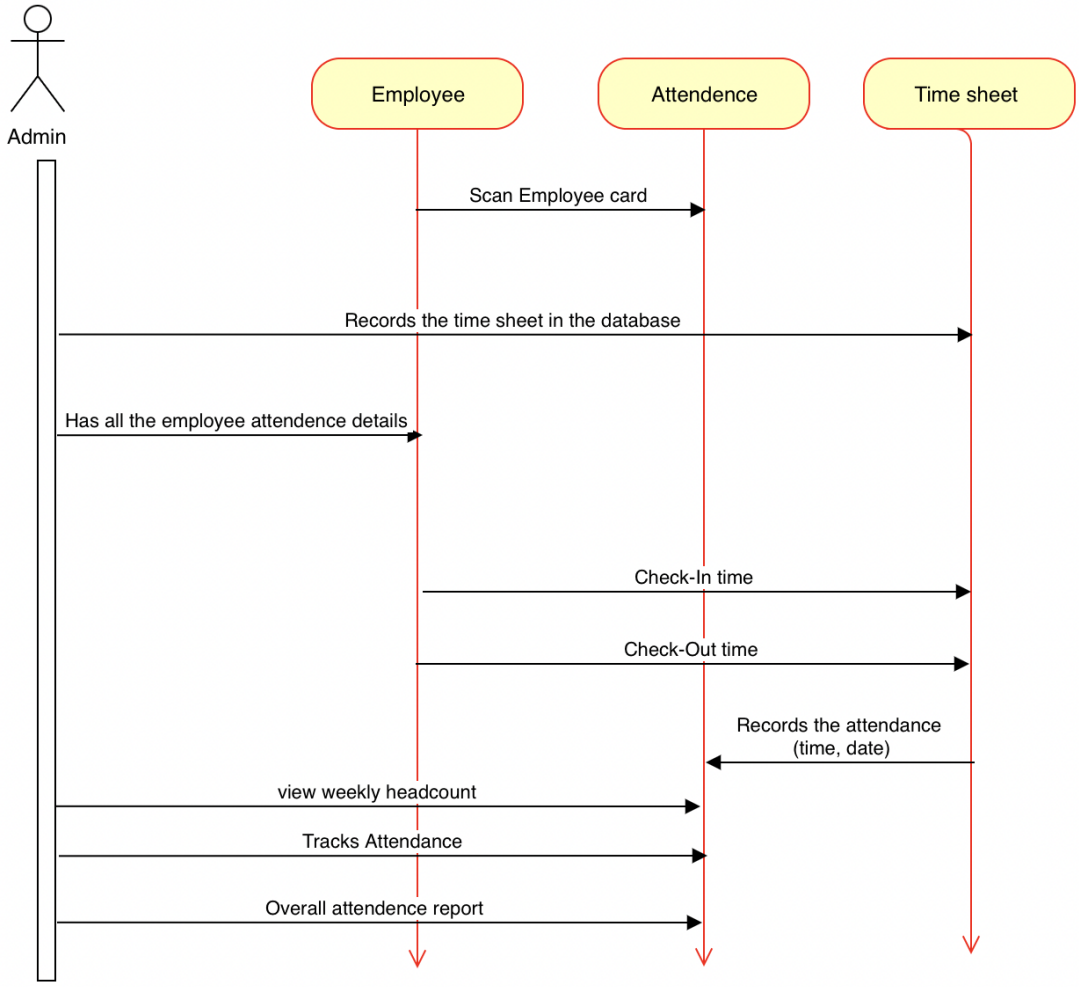


Sequence Diagram

A sequence diagram is a sort of interaction diagram that shows how a group of items interacts and in what order. Software engineers and business experts use these diagrams to understand the requirements for a new system or to describe an existing process. Event diagrams and event scenarios are other names for sequence diagrams.

Sequence diagram has the Activities Card, employee, Timesheet and Attendance

Actor here is the Admin and has the activation box. The messages are passed between the activities. Asynchronous and Synchronous message symbol.



3. Design workflow

3.1 Data Flow Analysis

Module Diagram

Module descriptions

Pseudocode

OR

3.2 Object-Oriented Design

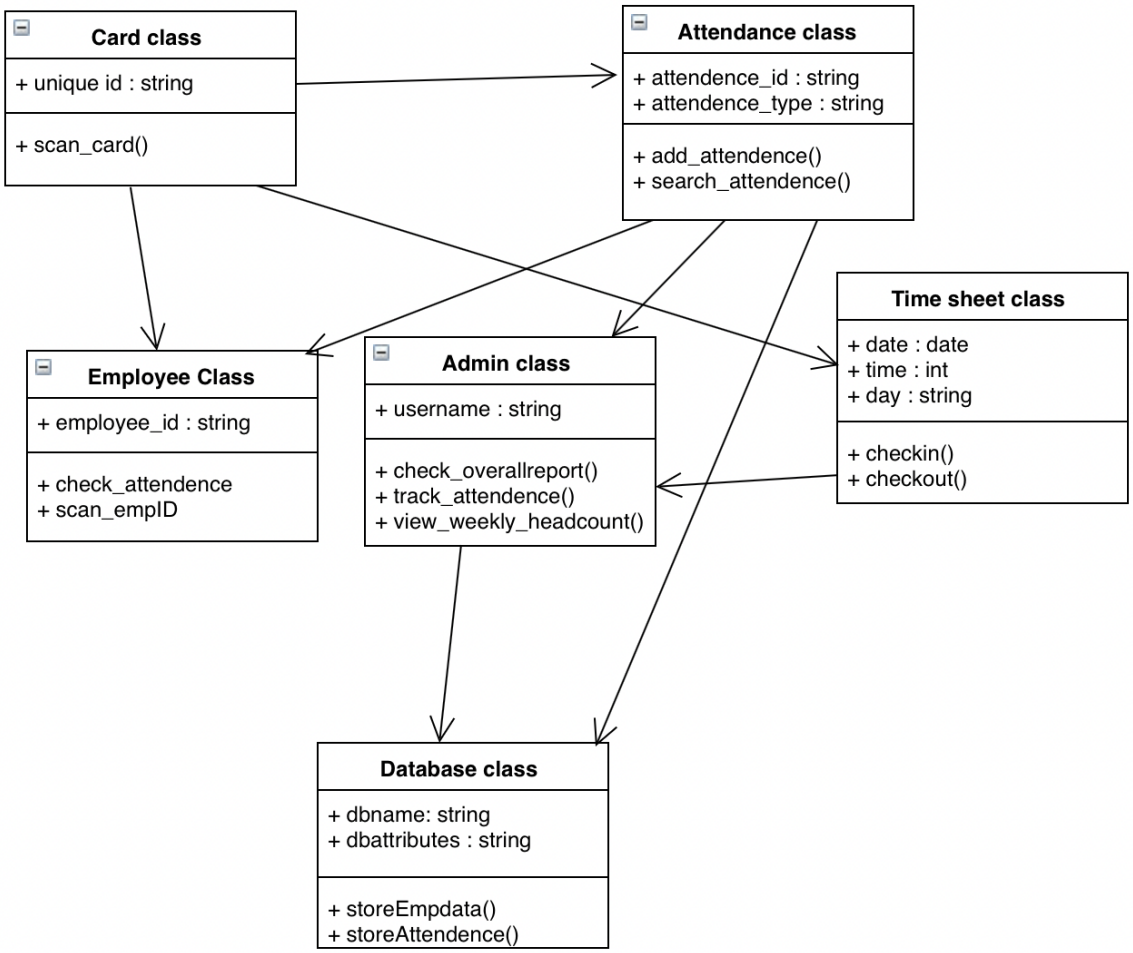
The conceptual model is subsequently evolved into an object-oriented model utilizing object-oriented design after the analysis phase. The analysis model's technology-independence are mapped onto implementing classes, constraints are established, and interfaces are built in OOD, resulting in a solution domain model.

The stages of object–oriented design are as follows: Definition of the system's context, System architecture design , The objects in the system are identified, Design model construction and Object interface specification

Completed Class Diagram

We have the all the main classes here in our Employee attendance system such as Card, Attendance, Database Admin and Employee.

The new class here is the Time-sheet class which has the date, time and day as the inputs. This class has the checks-in and checks-out methods. The time when the employee scans the entry and the exit is recorded.



Pseudocode

4. Implementation workflow:

Source code

5. Running results and snapshots