‘Onset’
Task Management

Executive Summary

Task management is the process of managing a task through its life cycle: from planning, testing and tracking, to reporting on the outcome. Onset, a task management tool has been prepared based on available data, forecasts provided by users and other management tools. This system is supported to eliminate and in some cases reduce the hardships faced by this existing system. Moreover this system is designed for the particular need of any company to carry out operations in a smooth and effective manner.

No formal knowledge is needed for the user to use this system . Thus, by this all it proves it is user friendly. Every organization , whether big or small , has challenges to overcome and manage the information of tasks, events, files, updates. Every management system has different user needs, therefore we design exclusive task management systems that are adapted to your managerial requirements. This is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of information and details for your future goals. Also, for those busy executives who are always on the go, our systems come with remote access features, which will allow you to manage your workforce anytime, at all times. These systems will ultimately allow you to better manage resources

Table of Contents

1. **Introduction**

1.a. **Project Introduction**

In a company where the hierarchy of employees spans over thousands, managing the work with them is a difficult job. And in an environment where several jobs are done simultaneously, picking the right person for the job is also a difficult task, as you are not aware of their availability. This online task management system application is designed for such an environment where the work is divided into groups of employees and during division the employees are selected to be part of the work in hand.

A powerful task management software gives the best tools when it comes to creating tasks, organizing, assigning, tracking, and reporting on team projects and the required tasks. You'll have all the information in one place visible to the entire team. A task management tool also lets you see your task and project progress visually and in real time.

So, to make this properly functional for everyone to access easily, we are going to develop a web-based task management system named "Onset".

2. **Background of the Study**

2.a. **Project Background**

Task management is the process of managing a task through its life cycle: from planning, testing and tracking, to reporting on the outcome. It involves managing all aspects of a task, from its status and priority to the time spent, people involved, and finally, financial and other resources needed

This online task management system software being web based is easily accessible from any corner of the company as every machine is part of a LAN network. The reason why it is made as a web application rather than a window based application is for the types of users. Administrator (Highest authority), Manager (working group Community). The activities underlying these sections are as per the company policies

3. **Objective**

3.a. **Primary Objective**

There are no set criteria for selecting the tool but it all depends on the type of organization, their needs, requirements, and work processes.A range of features such as ease of use, security, expenses, and reliability can significantly affect your choice when selecting the appropriate tool.

3.b. **Secondary Objective**

- Plan, Organize, Track all projects in one place.
- Allows you to set goals, keep track of deliverables, and manage deadlines.
- Track the project schedule using a calendar.
- Offers comprehensive, customizable schedules to facilitate uninterrupted workflows.
- Customizable options help employees work with what he/she finds comfortable.

4. **Methodology**

4.a. **Process Model**

We choose to use the Waterfall model as it is simple yet effective model for our application's scope

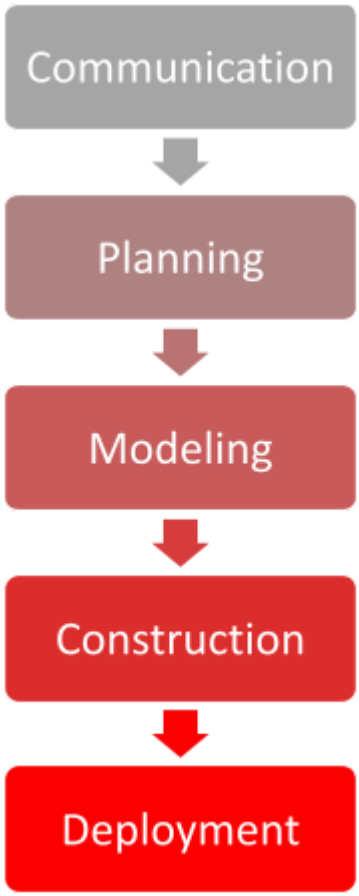


Fig: Waterfall Model

Here's an overview of how the waterfall model can be applied to a task management system web application OnSet, with a focus on communication, planning, modeling, construction, and deployment:

1. **Communication:** In this phase, the project team establishes communication channels and protocols to ensure effective communication throughout the project. This includes identifying stakeholders, defining communication methods, and establishing reporting structures.

2. **Planning:** In this phase, the project team defines the project scope, objectives, and deliverables. A project plan is created, which includes project schedules, budgets, and resource requirements.

3. **Modeling:** In this phase, the project team creates a detailed design of the task management system web application. This includes defining the user interface, database schema, system architecture, and functionality.

4. **Construction:** In this phase, the system is developed based on the design specifications. This includes coding, testing, and integrating various modules. The project team ensures that the system meets the requirements defined in the modeling phase.

5. **Deployment:** In this phase, the completed task management system web application is deployed to the production environment. This includes configuring the server, installing the software, and ensuring that the system is working as intended. The project team also conducts user training and ensures that the end-users can access and use the system effectively.

Throughout each phase, the project team must document progress, report any issues, and adjust the project plan as necessary. In the waterfall model, each phase must be completed before moving onto the next one, and the final task management system web application is not delivered until the deployment phase is complete. It is essential to ensure that the requirements are accurately defined in the modeling phase to avoid costly rework or delays later in the project. Effective communication among team members, stakeholders, and end-users is crucial to ensure the successful completion of the project.

5. **The Project**

5.a. **Communication**

5.a.i **Organization Visit**

For the organization visit, we visited two organizations.

a) Agni Systems Limited
b) Yokogawa India Limited

In Agni Systems Limited, we met with Farhana Haque, Head of Sales & Marketing. We discussed our project and its plan thoroughly. Then we asked her about her concern and feedback about our project. To quote her,

'I think this is a great initiative. When the organization becomes bigger, it gets muddled up. Who's doing what and why etc. Currently, we are using Microsoft Team which is serviceable at best. It doesn't have a dedicated task tracker and to track tasks, we need to always monitor their screen via meetings which is tedious for both parties. Also to see what job I have remaining or not is a quality of life feature. Currently, Onset is going in the right direction.'

She also showed concern about runtime of the application as due to load shedding and connectivity issues, the server might not be running always which might hamper the workings. She put most emphasis on Task Assignment and Task Tracking of our application as communication is not that important when there are other applications just for that. She also liked the Calendar and Todo list feature as it is tedious to sit through thousands upon thousands of emails just to see if there is a meeting tomorrow or not.

Overall, the feedback was positive as she was looking forward to the deployment of this application and wanted to test the application for the organization.

In Yokogawa India Limited, we met with Saude Al Faishal Ahmed, Operation & Maintenance Engineer. After hearing our plans, the response was mostly positive. Although he said that It might be possible to abuse the report system to flag an individual wrongly. He was also concerned about fake id and on the internet, it is easy to impersonate others. So he said that there should be a verification system for both managers and employees. He also said that Task tracking and assigning should be fully-fledged and realized as it is lacking from other applications.

Overall, the response was positive as he is certain that this application will be beneficial for all kinds of organizations in the long run.

5.b. **Planning**

5.b.i. **PERT**

PERT stands for "Program Evaluation and Review Technique," and it's a project management tool used to plan, schedule, and control complex projects like task management systems. A PERT chart is a graphical representation of a project's timeline, showing the dependencies between tasks and the critical path.

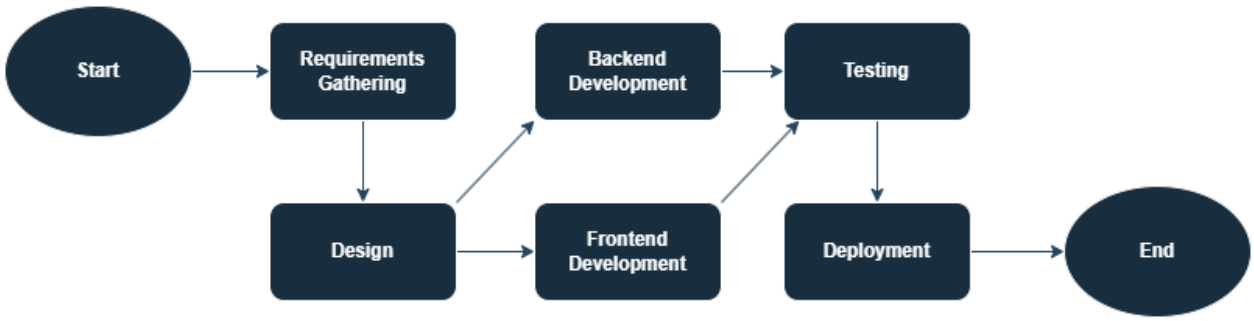
So, the PERT table for our project OnSet will be containing three columns about Task, Predecessors and Duration(weeks) which is given below :

Task	Predecessors	Duration (Weeks)
Requirements Gathering	-	2
Design	Requirements Gathering	3
Frontend Development	Design	6
Backend Development	Design	8
Testing	Frontend Development, Backend Development	4
Deployment	Testing	2

PERT Table

In this table, the first column lists each task that must be completed to develop and deploy the task management system web application. The second column specifies the predecessors for each task, indicating which tasks must be completed before work can begin on the current task. The third column lists the duration estimates for each task in weeks.

So, The PERT chart based on predecessor is given below :



5.b.ii. **CPM**

CPM or the Critical Path Method is an algorithm used in project management that is used to schedule project activities. The critical path refers to the longest stretch of the activities, and a measure of them from start to finish.

To calculate the critical path from the Table-1 provided earlier, we first need to calculate the earliest start time (EST), earliest finish time (EFT), latest start time (LST), and latest finish time (LFT) for each task:

Task	Predecessors	Duration (Weeks)	EST	EFT	LST	LFT
Requirements Gathering	-	2	0	2	0	2
Design	Requirements Gathering	3	2	5	2	5
Frontend Development	Design	6	5	11	5	11
Backend Development	Design	8	5	13	9	17
Testing	Frontend Development, Backend Development	4	13	17	13	17
Deployment	Testing	2	17	19	17	19

CPM Table

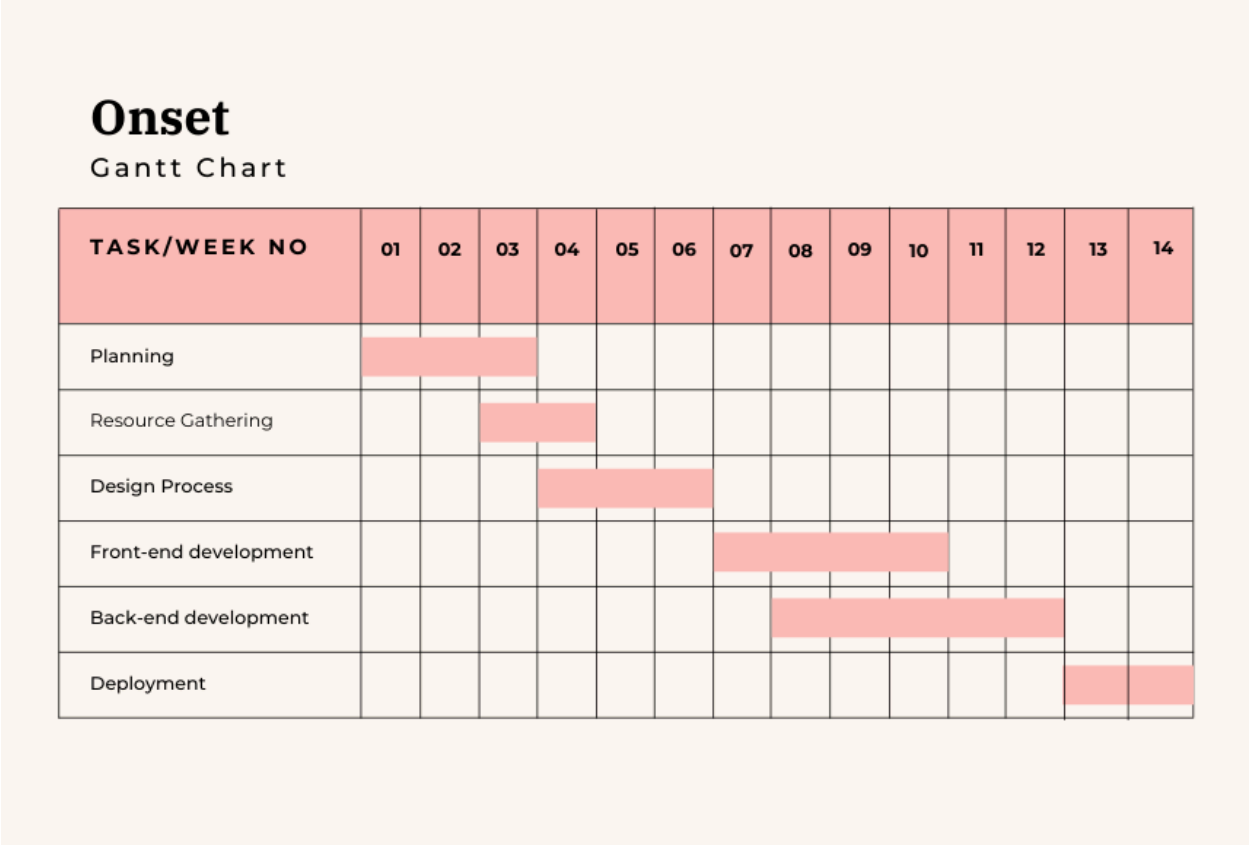
Finally, we can identify the critical path as the longest path through the project that includes tasks with no flexibility in their scheduling. In this case, the critical path is:

Requirements Gathering → Design → Frontend Development → Testing → Deployment

So the critical path for this project is Requirements Gathering, Design, Frontend Development, Testing, and Deployment, with a total project duration of

2 + 3 + 6 + 4 + 2 = 17 weeks, which is the minimum amount of time required to complete the project. Any delay in one of these tasks will result in a delay in the overall project timeline.

5.b.iii. **Gantt Chart**



5.b.iv. **Project Planning**

Our Project On-Set is a web application designed for such an environment where the work is divided into groups of employees and during the course of division the employees are selected to be part of the work and there will be a manager who will guide the employees through online accordingly.

To create this application, we have identified some vital stages which will guide us through the whole project development period. Those stages are:

Database Design: Database design is the process of producing a detailed data model of a database for the application. This data model will contain all the needed logical and physical design choices and physical storage parameters needed to generate a design for our application. Here we will use MSSQL which is a set of hosting services for any type of application. It offers real-time hosting of databases, content, social authentication, and notifications, or services, such as a real-time communication server. With the help of this, the employees will be able to know what task he/she will have to do in a certain time-period and the manager also can keep track of clients demand and employees working processes and so on.

UI Design: UI design is mainly designing the application's visual experience with the needs of the user in mind. UI design mainly aims to create attractive, simple to use and scalable applications. So, managers and employees can have a better communication environment.

Backend Coding: Back-end development is the process of creating the server-side logic that runs the background operations of web applications. It contains every piece of code required to develop the database, server, and application. To create this application, we will mainly use C-Sharp as our backend language. There will be more features like manager-employee group chat communication and time estimated for employees for certain work and so on that will need the help of backend coding.

Time Efficiency: Reduce the time it takes for a manager and employee communication and everyone will be able to work from home.

Testing: Application testing is the process of checking applications for bugs utilizing test automation frameworks, tools, and scripts. It will enable our team to publish strong, bug-free software applications into the actual world. Additionally, it will allow our teams to spot faults quickly and cut down on development time. The testing phase of our application is a bigger cycle that also includes research, process flow design, technical documentation, thorough testing, debugging, and iteratively pushing the program to live.

5.b.v. **Software Estimation**

- **FP Estimation**

Factor	Value
1. Backup and Recovery	4
2. Data Communication	4
3. Distributed Processing	5
4. Performance Critical	2
5. Existing Operating Environment	0
6. Online Data Entry	0
7. Input Transaction over multiple screens	4
8. Master Files updated online	3
9. Information domain values complex	4
10. Internal processing complex	2
11. Code designed for reuse	4
12. Conversion/installation in design	3
13. Multiple installations	2
14. Application designed for change	4
Value Adjustment Factor (0.65+ 0.01 X $\sum F_i$)	1.06

Measurement Parameter	Count	Weight	Weighting Factor (Count * weight)
Number of external inputs (EI)	35	4	140
Number of external outputs (EO)	25	4	100
Number of external inquiries (EQ)	20	6	120
Number of internal Logical files (ILF)	1	10	10
Number of external interfaces (EIF)	0	5	0
Count Total			**370**

So, Sum of all F_i (1 to 14) = (4+4+5+2+0+0+4+3+4+2+4+3+2+4) = 41

FP = Count-Total * [0.65 + 0.01 * sum of (F_i)]

= 370 * [0.65 + 0.01 * 41]

= 370 * 1.06

= 392

Average Productivity = 8 FP / month

Average Salary = 10,000 BDT / month

Cost per FP = (10,000 / 8) = 1250 BDT

Estimated Cost = (1250 * 392) = 490,000 BDT

- **Process Based Estimation**

Activity	**CC**	**Planning**	**Risk Analysis**	**Engineering**		**Construction Release**		**CE**	**Totals**
Functions				**Analysis**	**Design**	**Code**	**Test**		
UICF				0.50	4.50	0.40	5.00	n/a	10.40
2DGA				0.25	4.00	0.60	2.00	n/a	6.85
3DGA				0.25	4.00	1.00	3.00	n/a	8.25
CGDF				0.50	4.00	1.00	1.50	n/a	7.00
DBM				1.00	5.00	1.00	5.00	n/a	12.00
PCF				0.25	2.00	0.50	1.50	n/a	4.25
DAM				0.50	3.00	1.00	1.00	n/a	5.50
Totals	**0.25**	**0.25**	**0.25**	**3.25**	**26.50**	**5.50**	**19**		**54.25**
%Effort	**1%**	**1%**	**1%**	**6%**	**48%**	**10%**	**35%**		

Average Salary = 10,000 BDT

Estimated Cost = (10,000 * 54.25) = 542,500 BDT

Estimated Effort = 55 person-month

5.b.vi. **Project Estimation**

The estimated cost of our project is around 650,000 BDT.

5.b.vii. **Cost benefit Analysis**

Present Value, PV = 650,000 BDT
Time, n = 3 (Years)
Interest Rate, I = 15%
Future Value, FV = PV * (1+i)^n
= 650,000* (1+0.15) ^ 3
= **988,568 BDT**

(This is a rough estimation. There is a possibility for loss)

5.b.viii. **Training**

Training always proves to be handy to get a jump on the latest technology and trends, especially in the software industry. With the technology changing every now and then, it is very much required for a developer or a tester to get comfortable with the new technologies. One might get inclined towards the training programs aiming to become an expert; however, there are endless reasons for opting for a training program. Today, the perception of training is to stay on top of the ceaseless technologies, which could give some professional advantage. Software Engineering Training customized as Software Requirements, Definition, Development & Management Training. This is intended for those wishing to learn about the best software engineering practices. It means the practical introduction to software development and specifically methods to elicit, analyze, define and manage requirements. Software Engineering Training teaches One about the technical best practices, and how, in a changing environment, to communicate and manage requirements.

Who should do this work:

1. Software Developers	2. Software Quality Assurance Engineers
3. Software Engineers	4. System Engineers
5. Test Engineers	6. Project Managers

5.b.ix. **Risk Analysis**

Software testing risk analysis is a technique for software testing that analyzes and measures software risk. Usually, software testing has focused on very simple function testing. A software risk analysis looks for code flaws that could jeopardize the code's stability, security, or performance. Complex applications that use numerous frameworks and languages can have highly difficult-to-find defects that cause the most software failures.

5.b.x. **Resource Requirements**

i) Human Resource (HR)
   1) One Programmer
   2) One Designer
   3) One Tester
   4) One Database Administrator

ii) Others
   1) Computer Equipment
   2) Domain Server
   3) Internet

5.c. **Modeling**

5.c.i. **Project Features**

a) Task Assignment
b) Task Tracking
c) Chat
d) To Do list
e) Calendar
f) Profit
g) Report

5.c.ii. **Function Definitions**

A. Task Assignment:

Manager , who will distribute, co-ordinate and assign tasks to the team members . where employees can get task from the manager and upload the work so that the manager can verify within deadline

B. Task Tracking:

With several tasks scheduled simultaneously and that too in teams, it is a struggle to keep a track on their completion. When a number of tasks are scheduled, especially in teams, keeping track of all the tasks,, it is wise to adopt a tool that can help to trace exactly which activities are started, in progress and done with.

C. Chat:

An intelligent task organizer application should provide you with an efficient collaboration channel. By using this option, the teams should be at the liberty to exchange needed information related to task sharing and task updates.

D. To-do List:

Everything in Onset starts with a task, which is an actionable to-do that can stand alone or be a building block for bigger projects. Connect the tasks on your to-do lists back to larger projects, loop in teammates, and share real-time updates online.

E. Calendar:

A good task management tool will be integrated with a simplified calendar that is easy to manage and access. Besides, Spot holes and overlaps in your workflow, and quickly make adjustments by viewing projects and their due dates on a calendar.

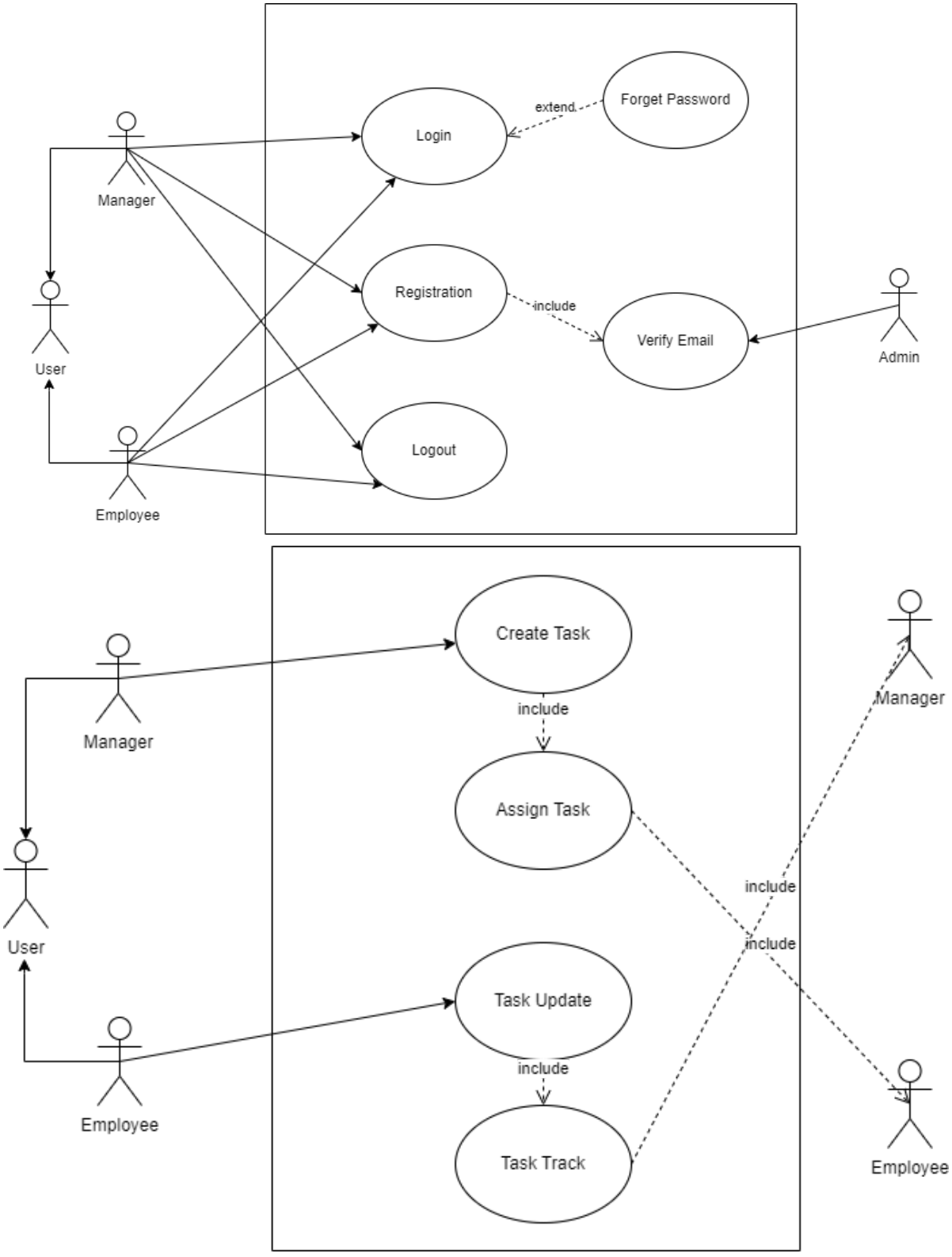
F. Profile:

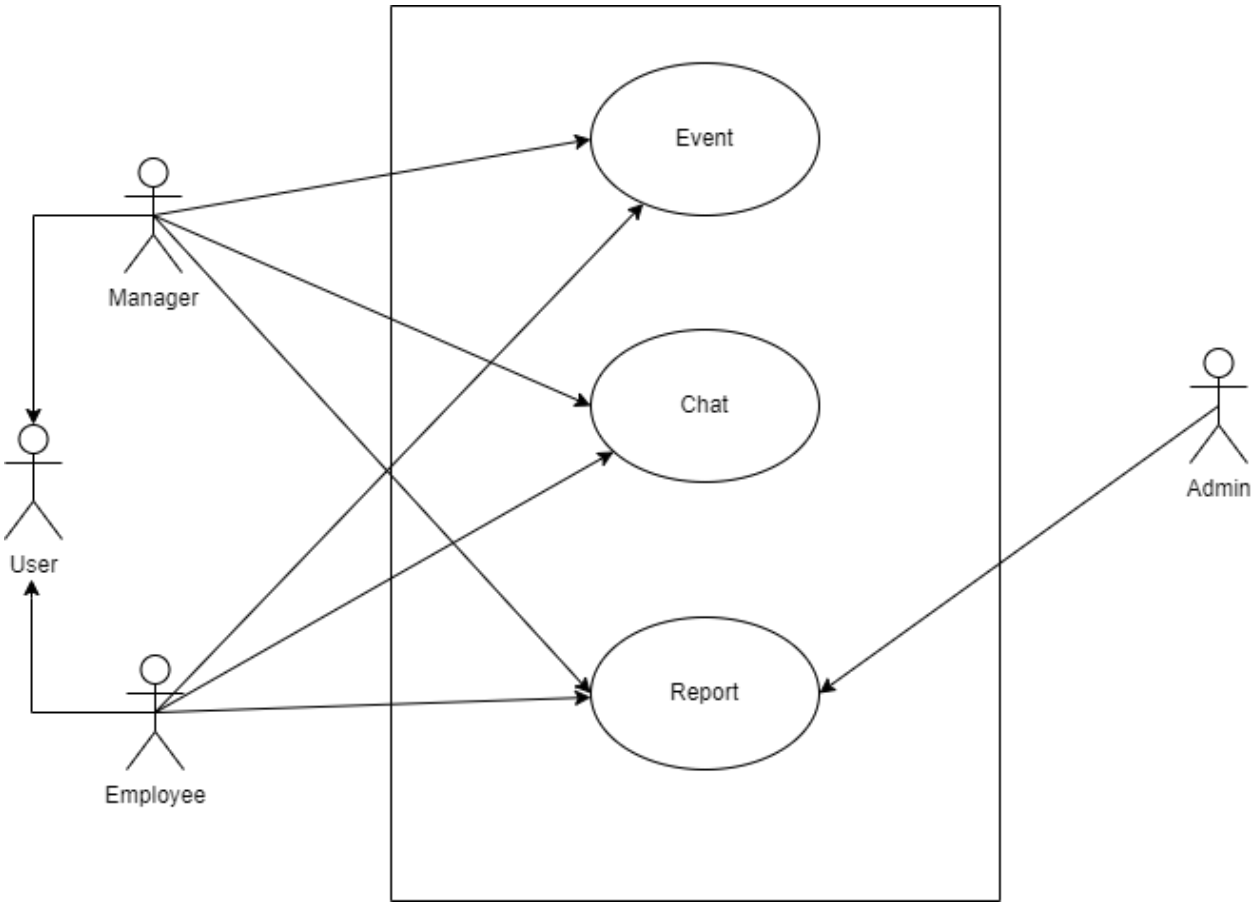
It contains critical information that is used to identify an individual, such as their name, age, portrait photograph and individual characteristics such as knowledge or expertise.

G. Report / Feedback:

Reports are a convenient way to check the progress of the project and help in identifying the best performing teams.Similarly, you can also take action against the team or individuals that are underperforming.

5.c.iii. **Use Case Diagram**




Event
Manager
Chat
User
Admin
Report
Employee

- **Use Case Narrative**

Registration:

Use Case ID:	1001		
Use Case Name:	Registration		
Process Owner:	Siddiqua Mumu	**Last Updated By:**	Siddiqua Mumu
Date Created:	02/02/2023	**Date Last Updated:**	16/02/2023
Business Actor:	User		
Description:	To Register to the system where it will either registers successfully or cancels registration		
Preconditions:	User must have a valid email account		
Postconditions:	The System shows relevant homepage of our application, and after registration users can have all facilities		
Performance Goal:	After the registration, users can enjoy all facilities and then the main is displayed		
Basic Workflow:	User must enter valid information regarding registration Upon entering, the system will verify the information If valid, then entries will be saved Else error message will be thrown		
Alternative Workflow:	N/A		
Category:	N/A		
Risks:	N/A		
Possibilities:	N/A		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

Login:

Use Case ID:	1002		
Use Case Name:	Login		
Process Owner:	Ayon Raihan	**Last Updated By:**	Ayon Raihan
Date Created:	02/02/2023	**Date Last Updated:**	16/02/2023
Business Actor:	Admin/User		
Description:	To Logged into the system when the registration will be completed and web application interface will shows up.		
Preconditions:	User must registered into the system.		
Postconditions:	The System shows relevant homepage of our application, and after login users can have all facilities		
Performance Goal:	Users can enjoy all the benefits of a very short and easy process.		
Basic Workflow:	User must enter valid information regarding log in Upon entering, the system will verify the information If valid, user can access the site Else error message will be thrown		
Alternative Workflow:	N/A		
Category:	N/A		
Risks:	N/A		
Possibilities:	N/A		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

Create Task:

Use Case ID:	1003		
Use Case Name:	Create Task		
Process Owner:	Saiful Islam	**Last Updated By:**	Saiful Islam
Date Created:	02/02/2023	**Date Last Updated:**	16/02/2023
Business Actor:	User		
Description:	User will choose the manager part where he/she can create /assign task towards the employees.		
Preconditions:	User must Logged in into the system.		
Postconditions:	Users created task will be added to assigned task.		
Performance Goal:	Users created task will directly notify the employee to complete the assigned task.		
Basic Workflow:	Manager will select a employee then he/she will give information about the task		
Alternative Workflow:	N/A		
Category:	N/A		
Risks:	N/A		
Possibilities:	N/A		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

Assign Task:

Use Case ID:	1004		
Use Case Name:	Assign Task		
Process Owner:	Saiful Islam	**Last Updated By:**	Saiful Islam
Date Created:	02/02/2023	**Date Last Updated:**	16/02/2023
Business Actor:	User		
Description:	User Employee will be assigned by a certain task by manager user to complete it accordingly.		
Preconditions:	Create Task by Manager		
Postconditions:	Employee will receive the task		
Performance Goal:	Employee user gets tasks from manager user to complete it accordingly.		
Basic Workflow:	Upon the creation of the task by manager The system will show the task to the employee		
Alternative Workflow:	N/A		
Category:	N/A		
Risks:	N/A		
Possibilities:	N/A		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

Task Update:

Use Case ID:	1005		
Use Case Name:	Task Update		
Process Owner:	Siddiqua Mumu	**Last Updated By:**	Siddiqua Mumu
Date Created:	02/02/2023	**Date Last Updated:**	16/02/2023
Business Actor:	User		
Description:	User Employee will be assigned by a certain task by manager user where employee can update the task.		
Preconditions:	User must Logged in into the system and complete the steps from Assign Task.		
Postconditions:	Task Update will be uploaded by users.		
Performance Goal:	Employees can update about the task and its progress.		
Basic Workflow:	After receiving the task The employee can upload a picture and a description about the task update		
Alternative Workflow:	N/A		
Category:	N/A		
Risks:	N/A		
Possibilities:	N/A		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

Task Upload:

Use Case ID:	1006		
Use Case Name:	Task Upload		
Process Owner:	Ayon Raihan	**Last Updated By:**	Ayon Raihan
Date Created:	02/02/2023	**Date Last Updated:**	16/02/2023
Business Actor:	User		
Description:	User Employees updated task will be uploaded.		
Preconditions:	User must Logged in into the system and complete the steps from Assign Task update.		
Postconditions:	File for task will be uploaded for the manager		
Performance Goal:	Employees can finish and upload the task.		
Basic Workflow:	After receiving the task The employee can upload a file about the task update		
Alternative Workflow:	N/A		
Category:	N/A		
Risks:	N/A		
Possibilities:	N/A		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

Chat:

Use Case ID:	1007		
Use Case Name:	Chat		
Process Owner:	Saiful Islam	**Last Updated By:**	Saiful Islam
Date Created:	02/02/23	**Date Last Updated:**	16/02/23
Business Actor:	USER/ ADMIN		
Description:	User Manager, employees and Admin can communicate with each other		
Preconditions:	User must Logged in into the system		
Postconditions:	The message will be delivered		
Performance Goal:	Employees, Manager and Admin can chat with each other..		
Basic Workflow:	Upon creating a room, user can send message to the room		
Alternative Workflow:	N/A		
Category:	N/A		
Risks:	N/A		
Possibilities:	N/A		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

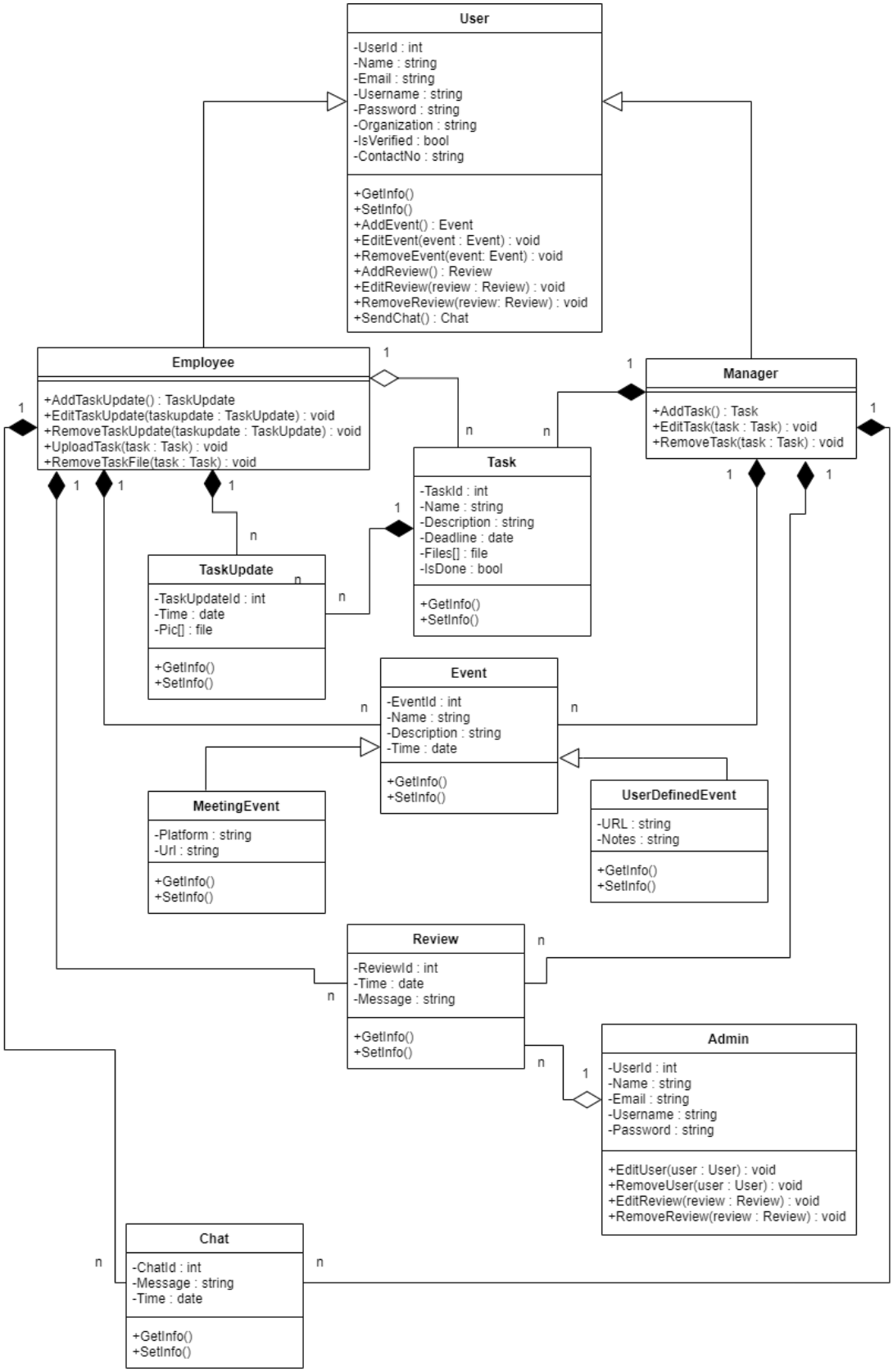
Event;

Use Case ID:	1008		
Use Case Name:	Event		
Process Owner:	Siddiqua Mumu	**Last Updated By:**	Siddiqua Mumu
Date Created:	02/02/23	**Date Last Updated:**	16/02/23
Business Actor:	USER/ ADMIN		
Description:	User Manager and employees can communicate with each other and fix reminders about planning		
Preconditions:	User must Logged in into the system		
Postconditions:	Everyone must follow the reminders in time		
Performance Goal:	Employee/ Manager and Admin can add events and reminders.		
Basic Workflow:	The system will create a event from the information provided		
Alternative Workflow:	N/A		
Category:	N/A		
Risks:	N/A		
Possibilities:	N/A		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

Review:

Use Case ID:	1009		
Use Case Name:	Review		
Process Owner:	Saiful Islam	**Last Updated By:**	Saiful Islam
Date Created:	02/02/23	**Date Last Updated:**	16/02/23
Business Actor:	USER/ ADMIN		
Description:	User Manager and employees can Review their task progress and Admin will monitor the review		
Preconditions:	User must Logged in into the system		
Postconditions:	NULL		
Performance Goal:	Employees and Managers can review each other. Admin will monitor the review.		
Basic Workflow:	A report will be send to the admin		
Alternative Workflow:	N/A		
Category:	N/A		
Risks:	N/A		
Possibilities:	N/A		
Special Requirements:	N/A		
Assumptions:	N/A		
Notes and Issues:	N/A		

5.c.iv. **Class Diagram**



- **Class Responsibility Collaborator**

Class: User	
Class Type: Role	
Responsibilities:	**Collaborators:**
Can verify email Can edit events Can check reviews Can Send Chats	Event Review Chat

Class: Manager	
Class Type: Role	
Responsibilities:	**Collaborators:**
Can assign task Can assign events Can delete events Can update reviews	Task Employee Review

Class: Employee	
Class Type: Role	
Responsibilities:	**Collaborators:**
Can update task Can assign event Can update reviews	Task Task update Event Review

Class: Admin	
Class Type: Role	
Responsibilities:	**Collaborators:**
Can check reviews Can edit reviews Can edit/delete accounts	Review Manager Employee

Class: Event	
Class Type: Event	
Responsibilities:	**Collaborators:**
Reminder for meetings Can be user defined	Manager Employee

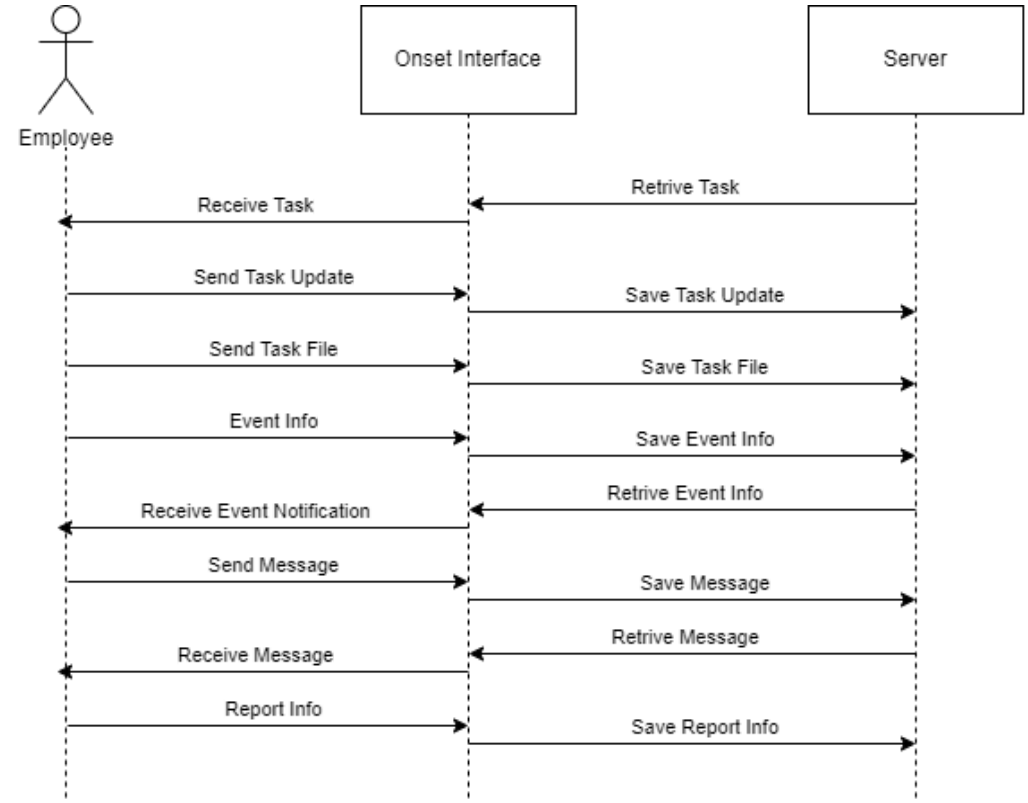
Class: Review	
Class Type: Event	
Responsibilities:	**Collaborators:**
Can be checked and edit by admin Can be updated by managers and employees	Admin Manager Employee

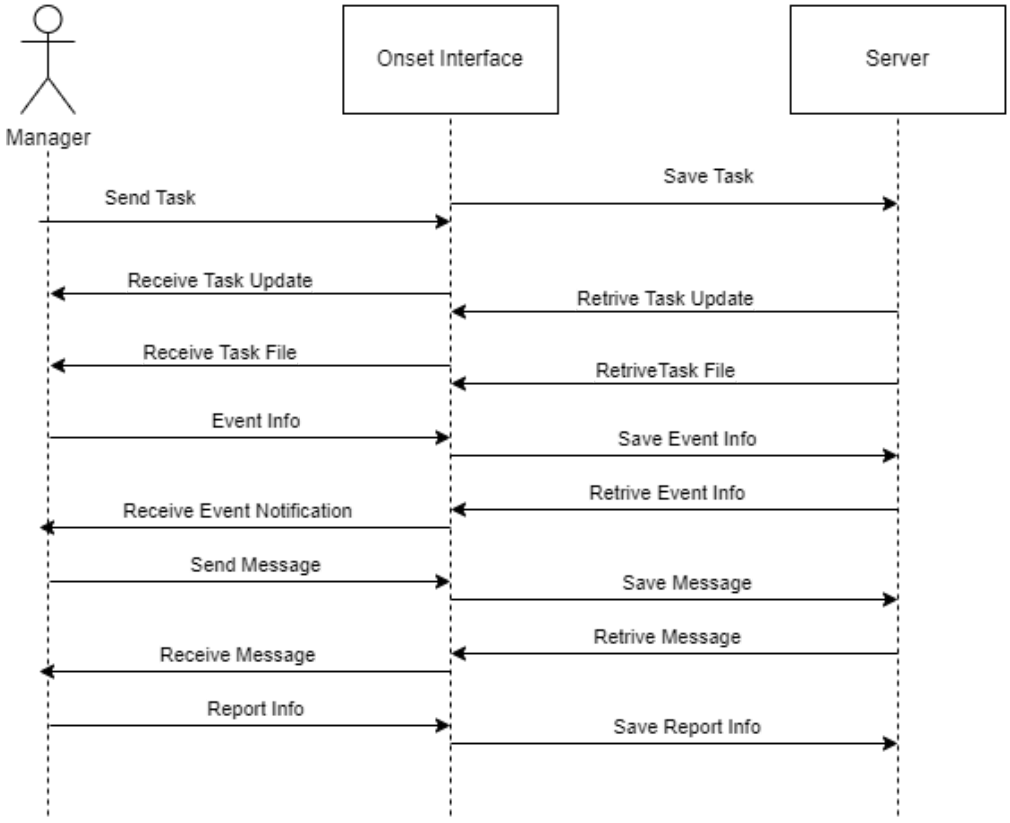
Class: Task	
Class Type: Event	
Responsibilities:	**Collaborators:**
Can be assigned by managers Can be done by employees	Manager Employee

Class: Task Update	
Class Type: Event	
Responsibilities:	**Collaborators:**
Can be modified by employees	Employee Manager

Class: Chat	
Class Type: Event	
Responsibilities:	**Collaborators:**
Used to send message between users	Employee Manager

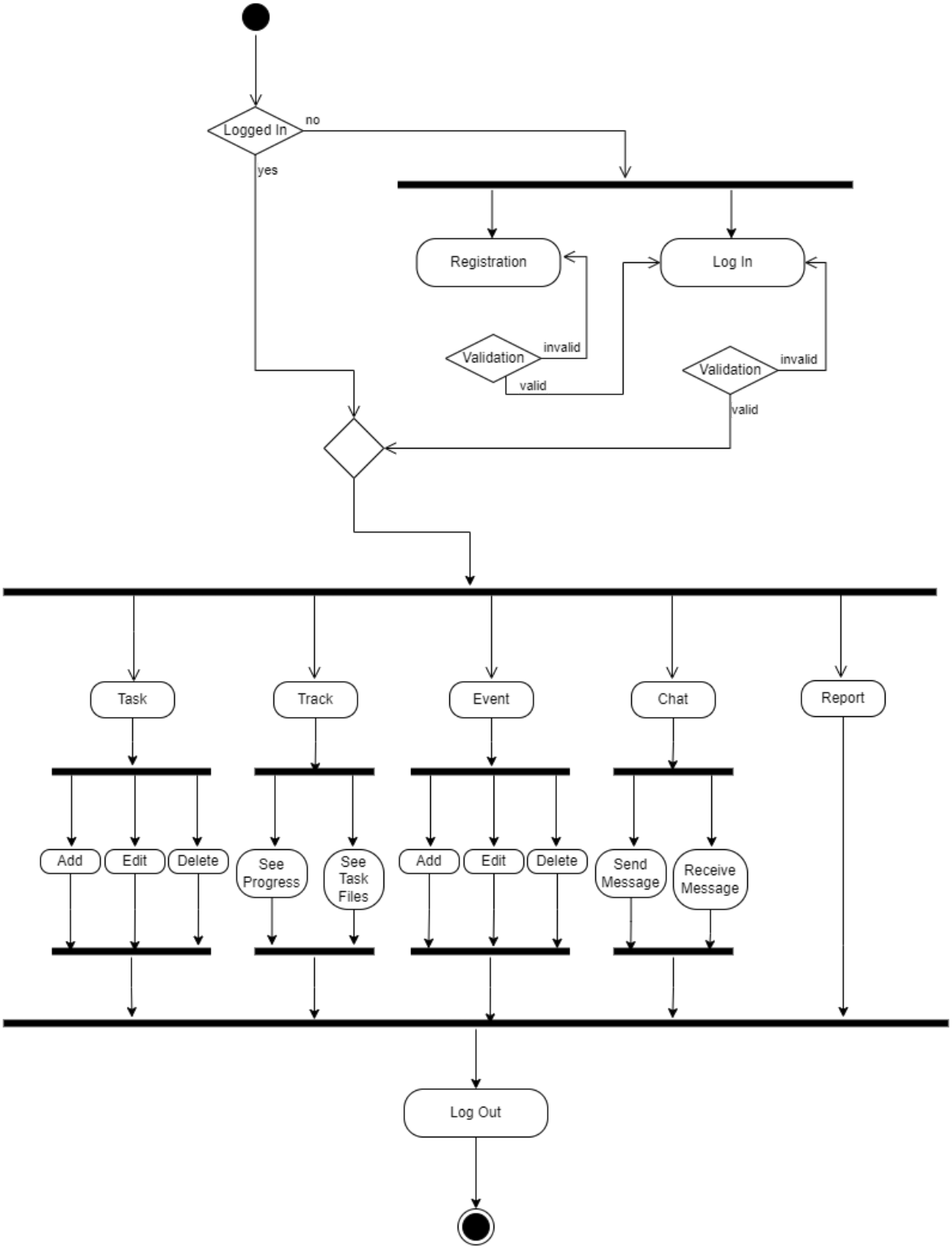
5.c.v. **Sequence Diagram**



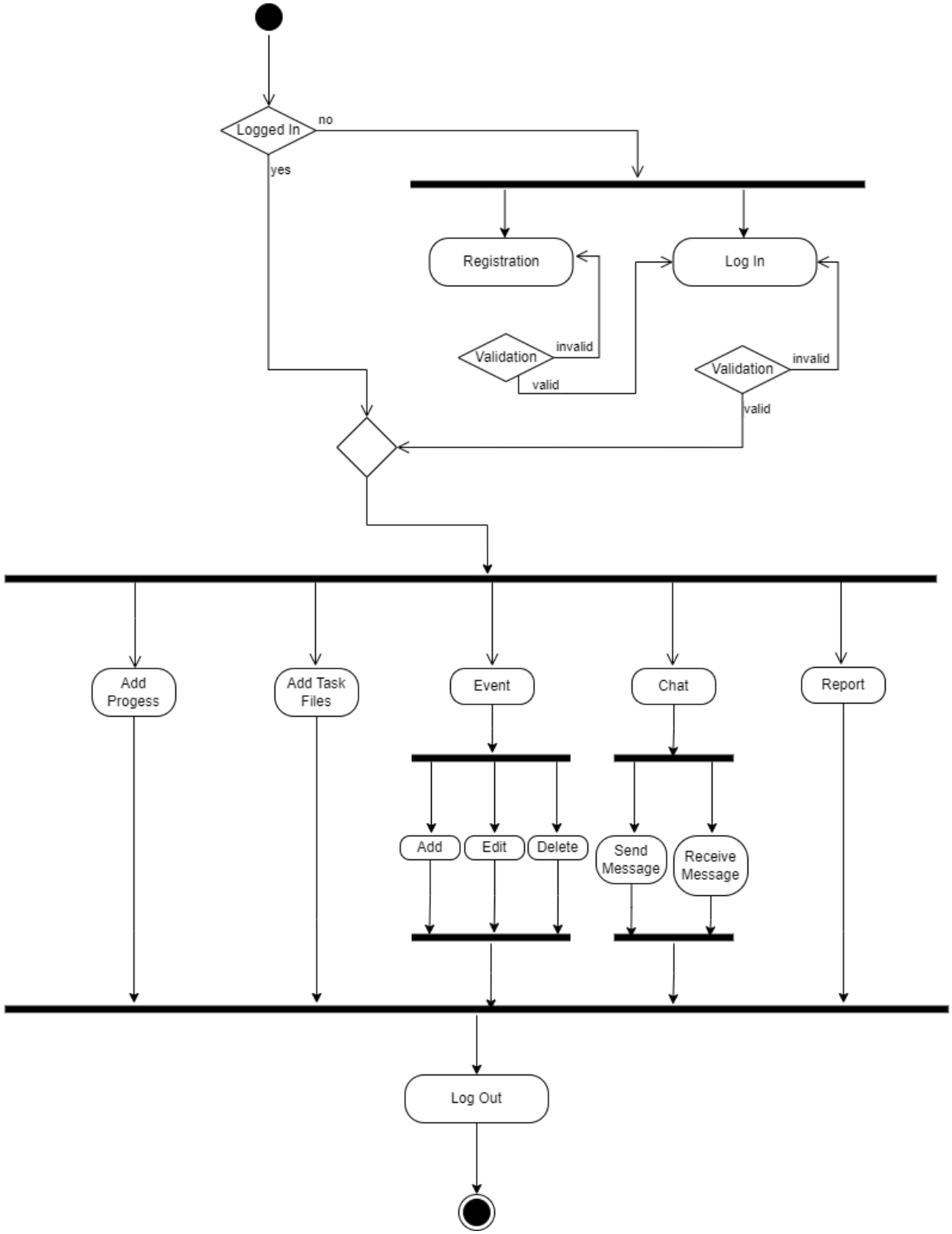


5.c.vi. **Activity Diagram**

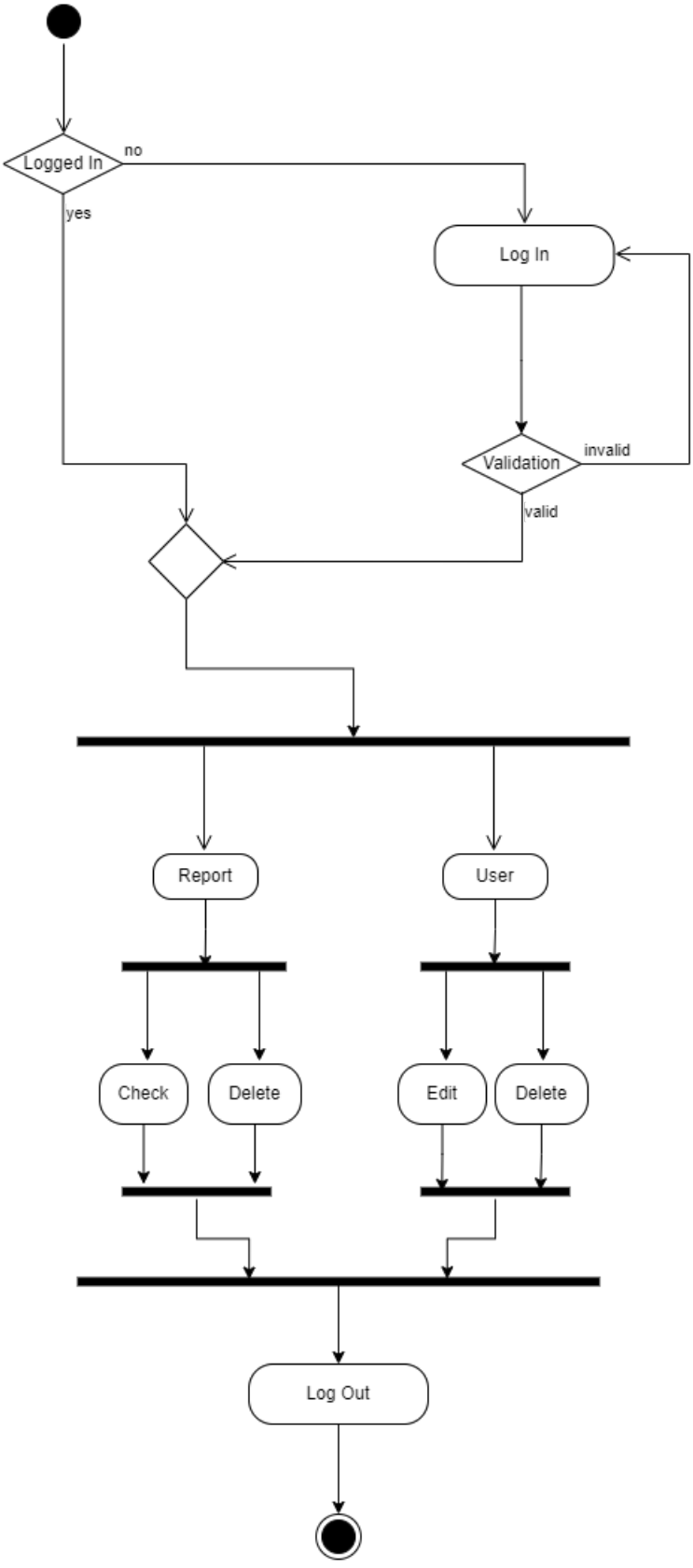
Manager Activity



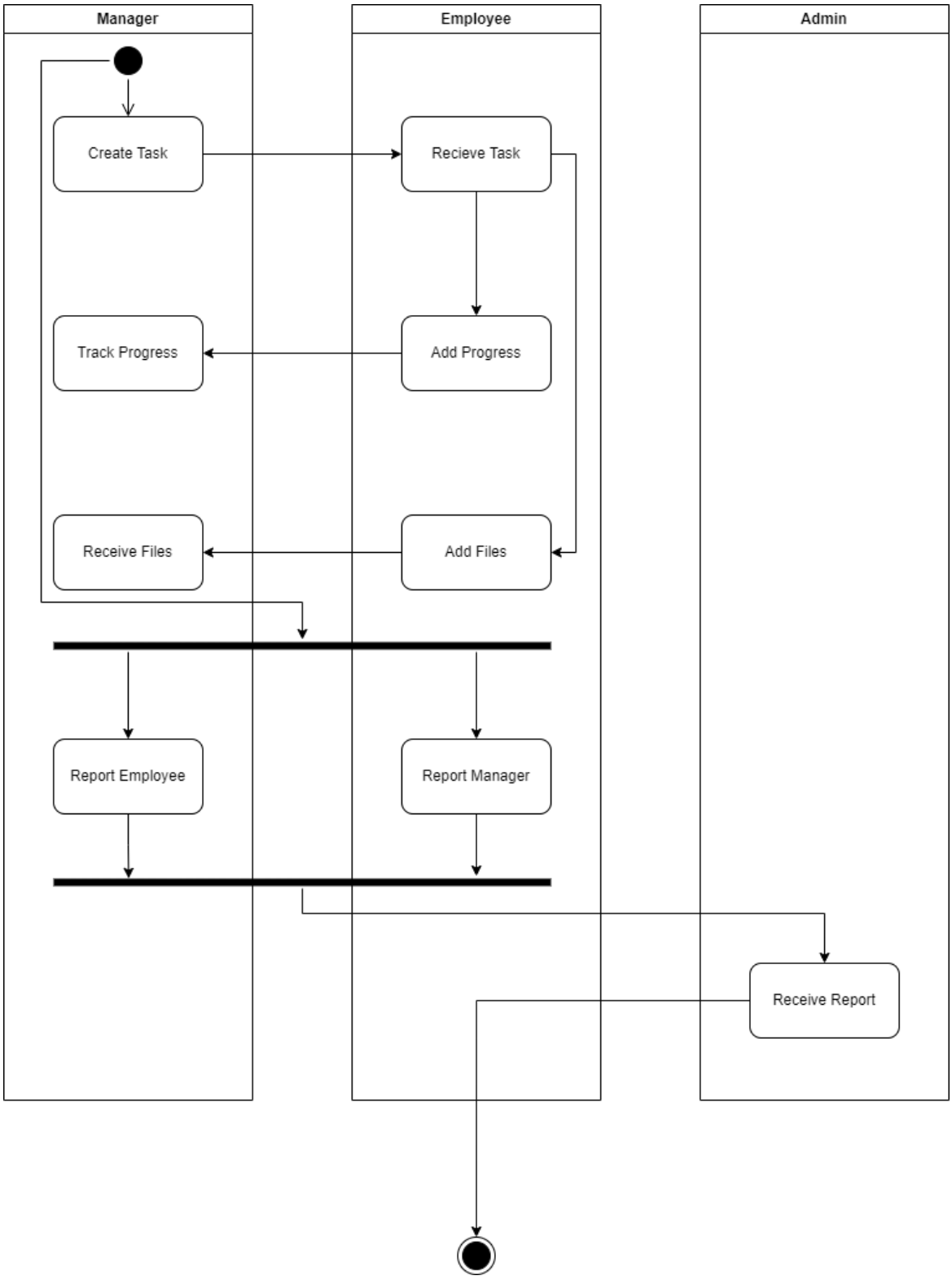
Employee Activity



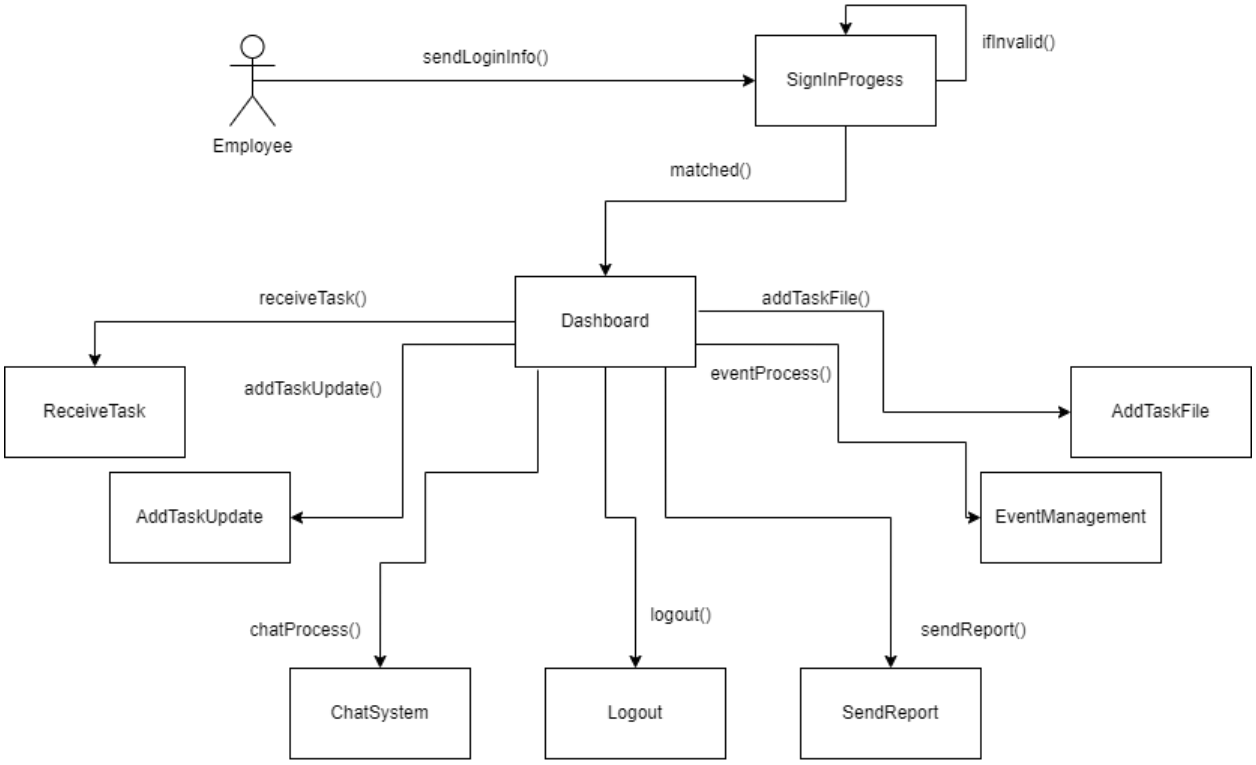
Admin Activity

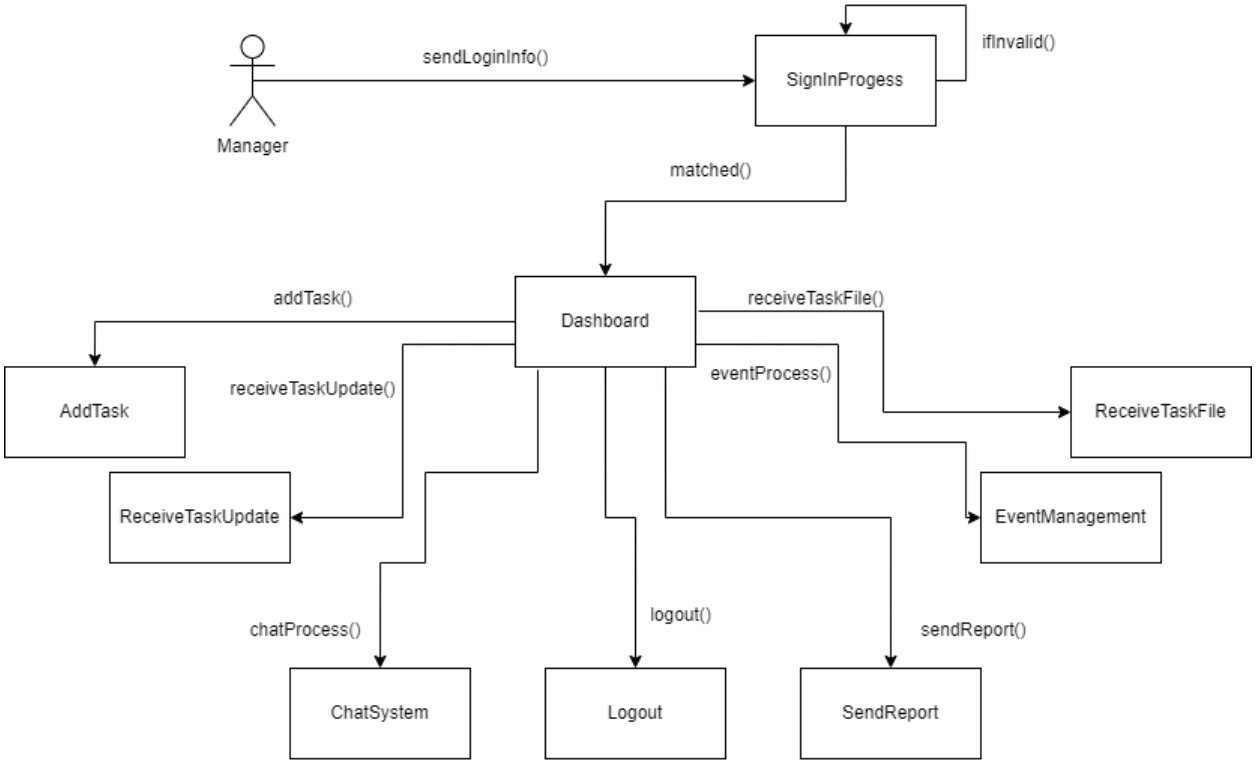


5.c.vii. **Swim Lane Diagram**



5.c.viii. **Collaboration Diagram**





5.c.ix. **Data Flow Diagram**

- **Process Specification**

Main Processes

1. Authentication
2. Task Management
3. Event Management
4. Report System
5. Database System
6. Chat System

Sub Processes

1.1. Registration
1.2. Login

2.1. Create Task
2.2. Assign Task
2.3. Task Update
2.4. Display Progress

3.1. Create Event
3.2. Edit Event
3.3. Calendar
3.4. Generate Notification

4.1. Create Report
4.2. Display Report

5.1. Edit Info
5.2. Check Report

- **Control Specification**
- **Context Level Diagram**

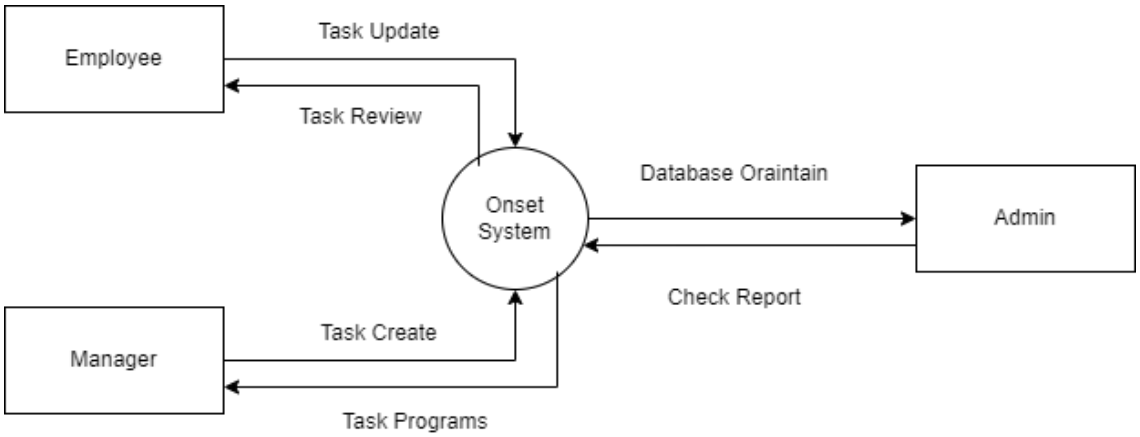


Fig.: Context Level Diagram

- **Level 1 Diagram**

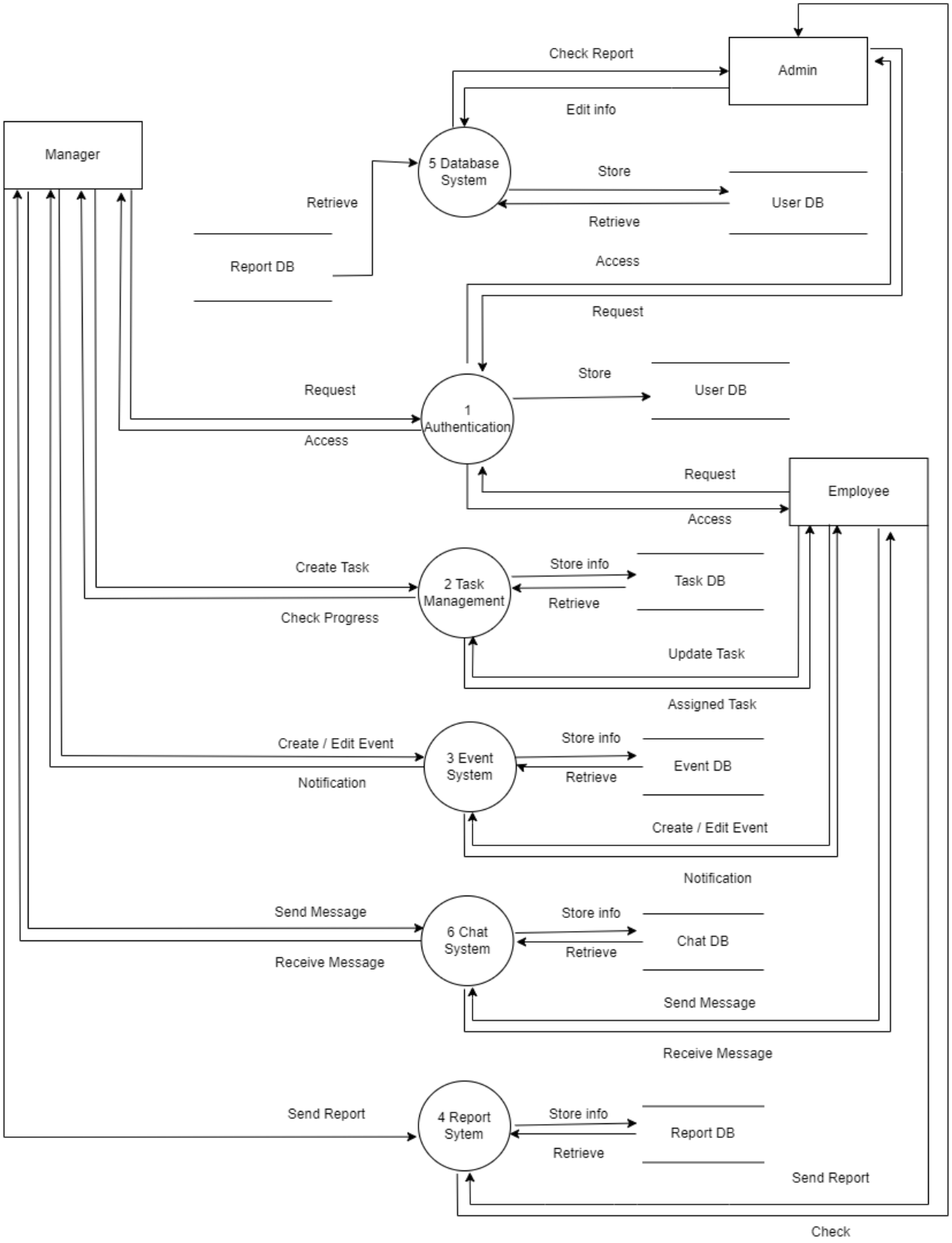


Fig.: Level 1 Diagram

- **Level 2 Diagram**

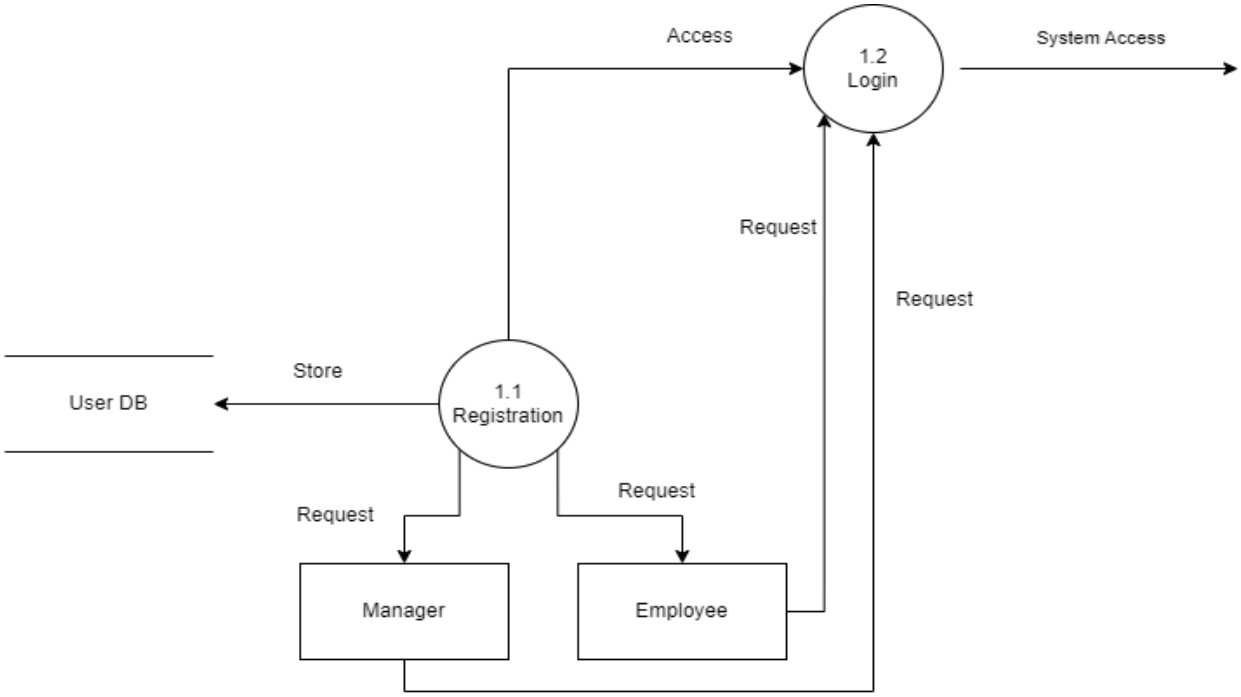


Fig. Level 2 Authentication

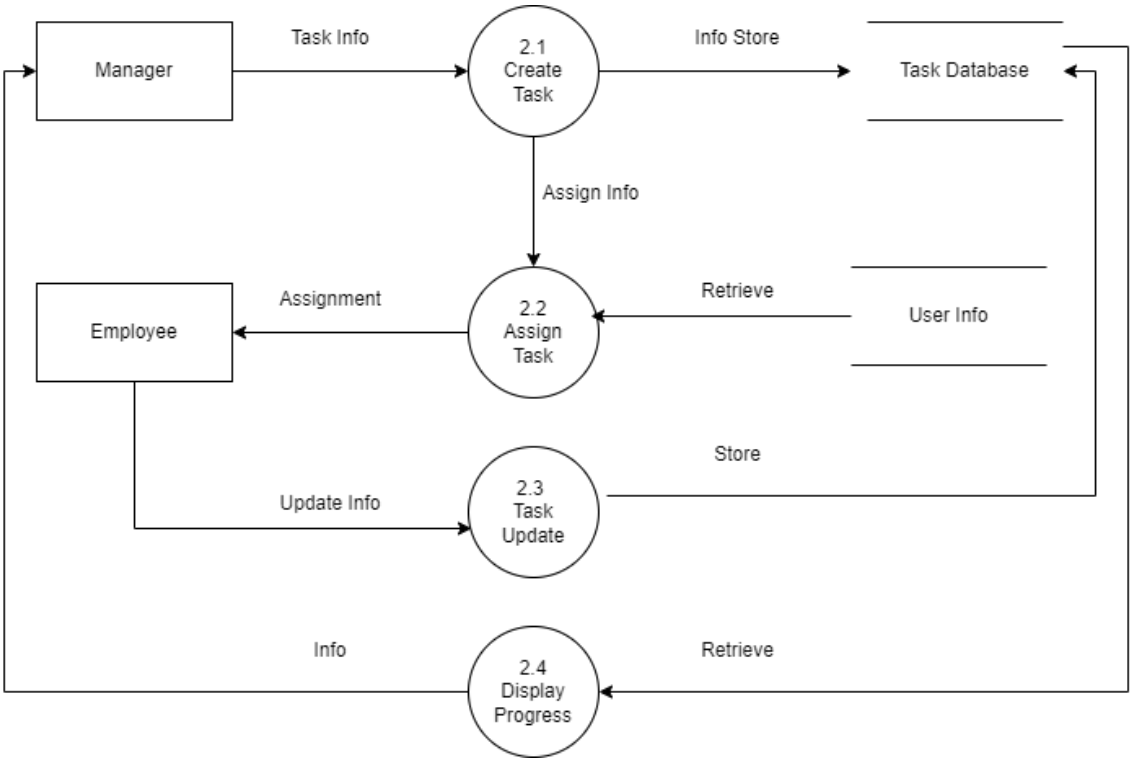


Fig. Level 2 Task Management

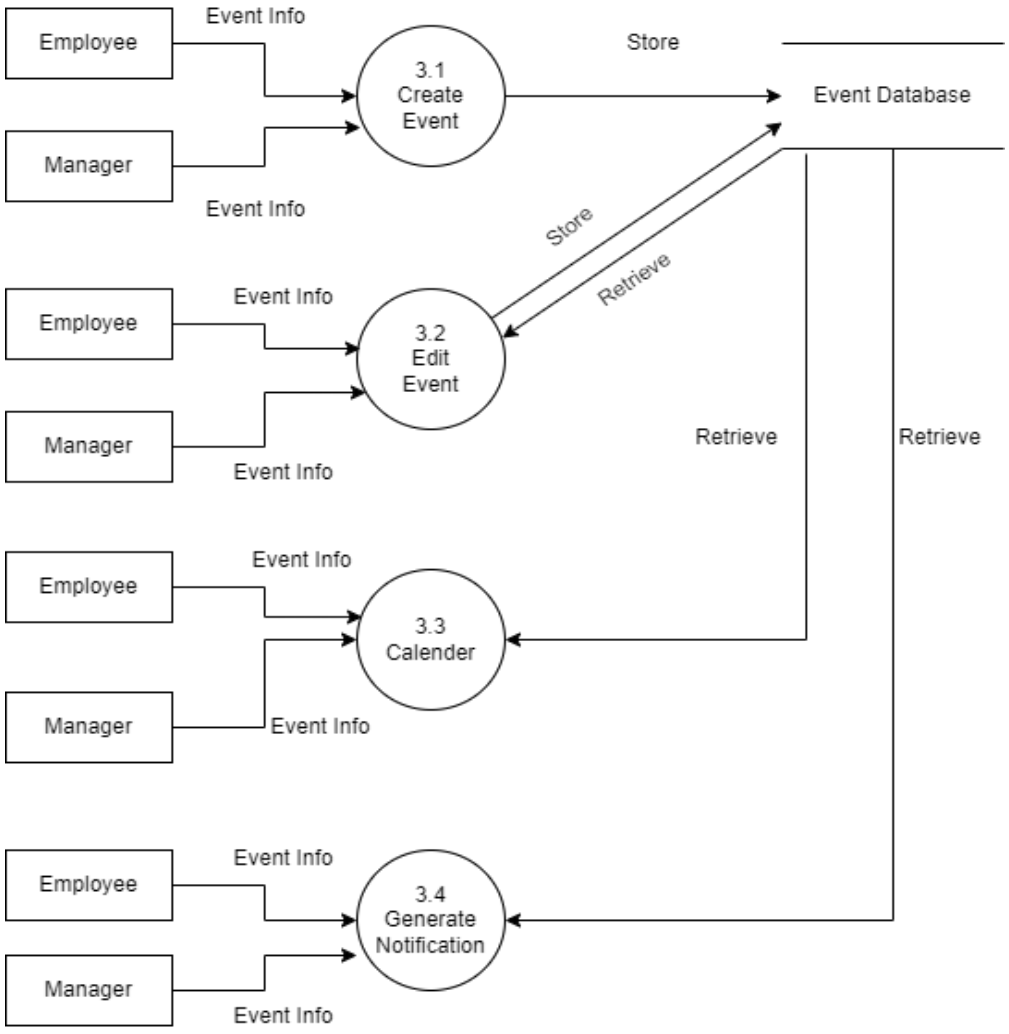


Fig. Level 2 Event System

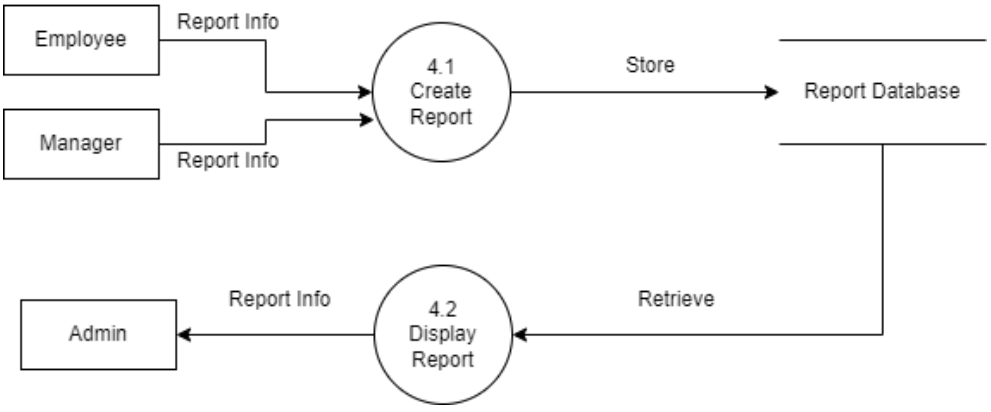


Fig. Level 2 Report System

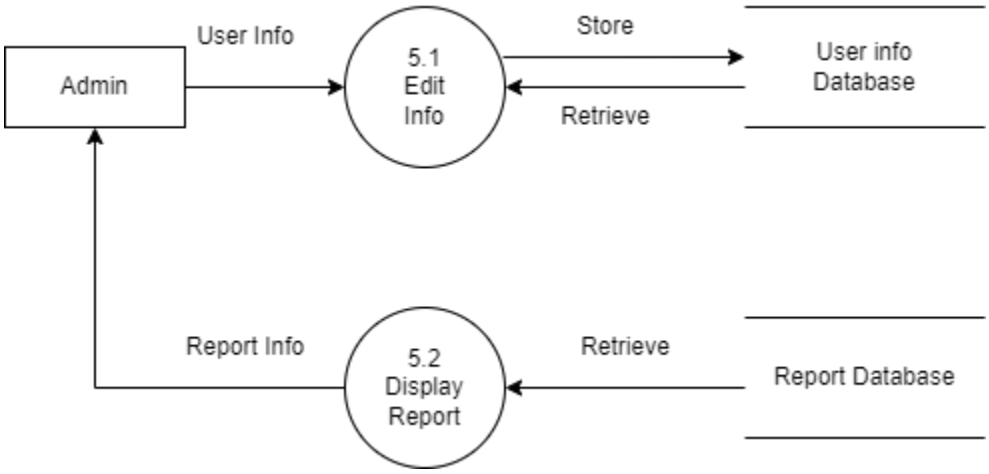


Fig. Level 2 Database Management

5.c.x. **Architecture Flow Diagram**

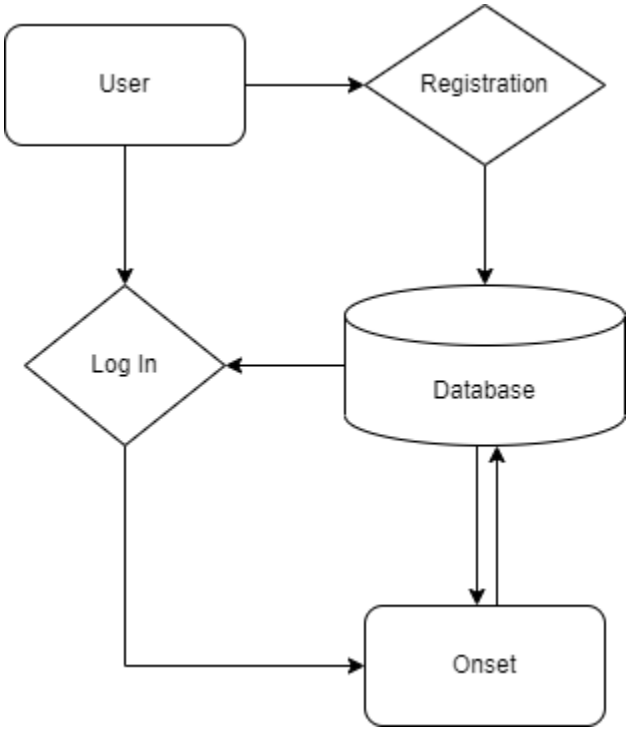
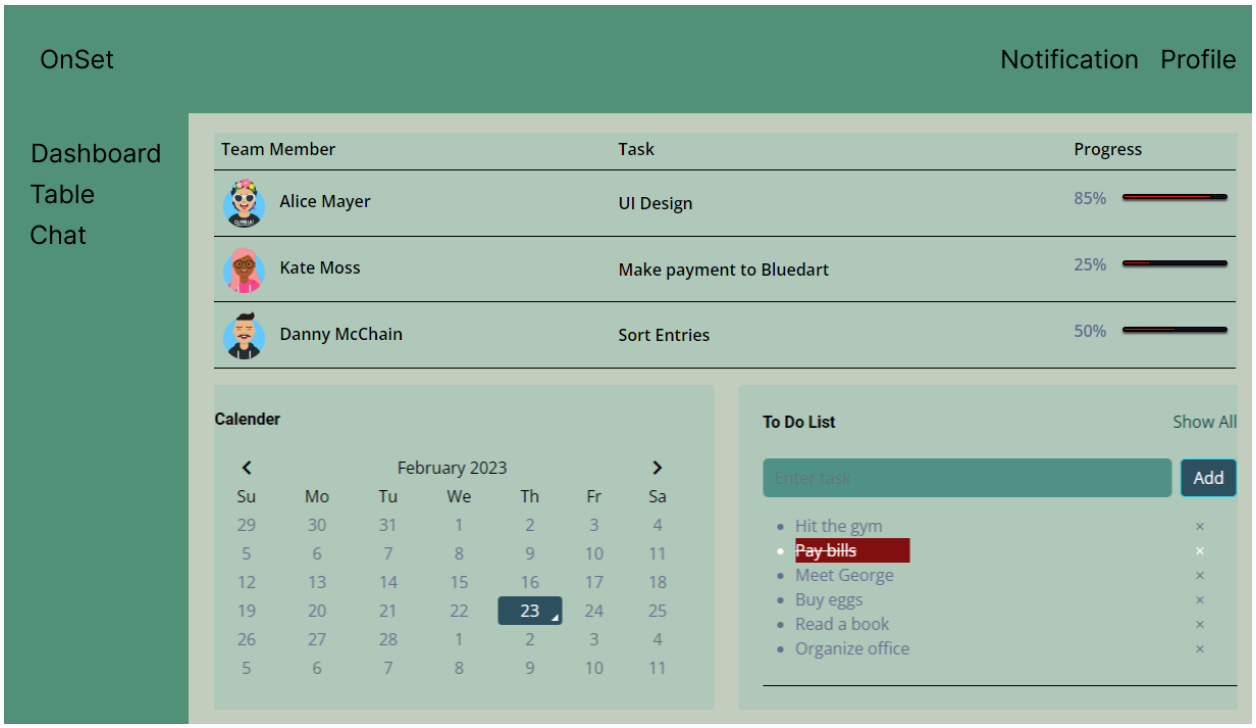


Fig: Architecture Flow Diagram

5.c.xi. **UI/UX Design**



5.c.xii. **Entities and Attributes**

Entity Names

1. Employee
2. Manager
3. Admin
4. Task
5. Progress
6. Event
7. Room
8. Report

Entity Set Names and Their Attributes Specified with Data Type

Employee

Attribute Name	Data Type	Specification
employeeID	int	Primary key
employeeFirstName	varchar	Not null
employeeLastName	varchar	Not null

Attribute Name	Data Type	Specification
employeeID	int	Primary key
employeeEmail	varchar	Candidate key,Not null
employeePassword	varchar	Not null
designation	varchar	Not null
employeePhone	varchar	Not null
organization	varchar	Not null
employeeGender	varchar	Not null
registration_date	datetime	Not null
dateOfbirth	datetime	Not null
employeeAddress	varchar	Not null

Manager

Attribute Name	Data Type	Specification
managerID	int	Primary key
managerFirstName	varchar	Not null
managerLastName	varchar	Not null
managerEmail	varchar	Candidate key,Not null
managerPassword	varchar	Not null
designation	varchar	Not null
managerPhone	varchar	Not null
organization	varchar	Not null
managerGender	varchar	Not null
registration_date	datetime	Not null
dateOfbirth	datetime	Not null
managerAddress	varchar	Not null

Admin

Attribute Name	Data Type	Specification
adminID	int	Primary key
adminName	varchar	Not null
adminUsername	varchar	Not null
adminEmail	varchar	Candidate key, Not null
adminPassword	varchar	Not null

Task

Attribute Name	Data Type	Specification
taskID	int	Primary key
taskName	varchar	Not null
description	varchar	Not null
deadline	datetime	Not null

Progress

Attribute Name	Data Type	Specification
progressID	int	Primary key
taskID	int	Foreign key
employeeID	int	Foreign key
description	varchar	Not null
file	varchar	Not null

Event

Attribute Name	Data Type	Specification
eventID	int	Primary key
eventName	varchar	Not null
description	varchar	Not null
deadline	datetime	Not null

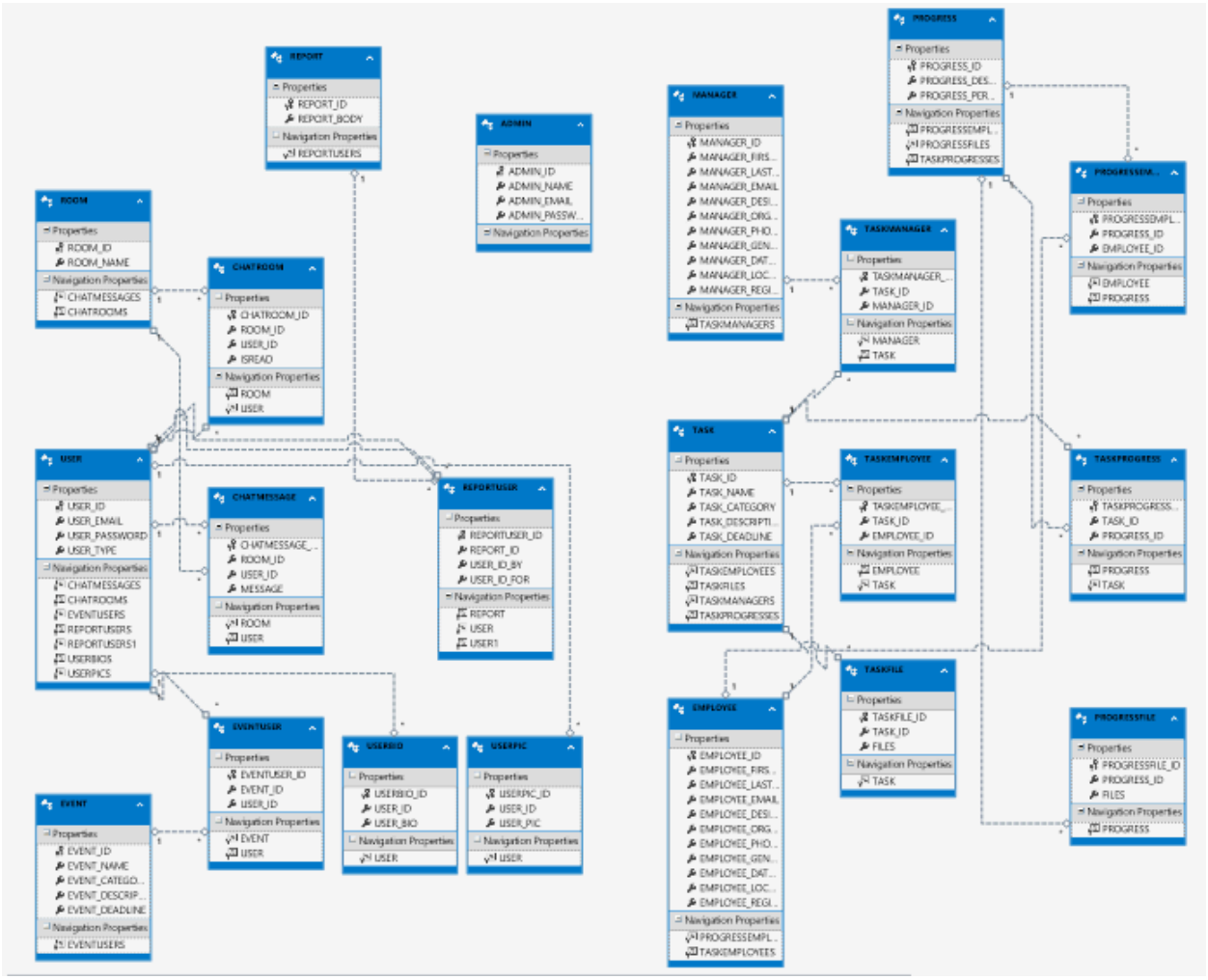
Room

Attribute Name	Data Type	Specification
roomID	int	Primary key
roomName	varchar	Not null

Report

Attribute Name	Data Type	Specification
reportID	int	Primary key
reportMessage	varchar	Not null

5.c.xiii. **Schema Diagram**



5.c.xiv. **Normalization**

Normalization is the process of organizing data in a database. This includes creating tables and establishing relationships between them based on rules designed to protect data and make databases more resilient by eliminating redundancies and inconsistent dependencies.

We have implemented our database in 5NF.

Task

TaskID	TaskName	Description	Deadline
…	…	…	…

TaskEmployee

TaskID	EmployeeID
…	…

TaskManager

TaskID	ManagerID
…	…

Here, All tables have primary keys and all the information is conveyed via columns. It follows the 1nf form.

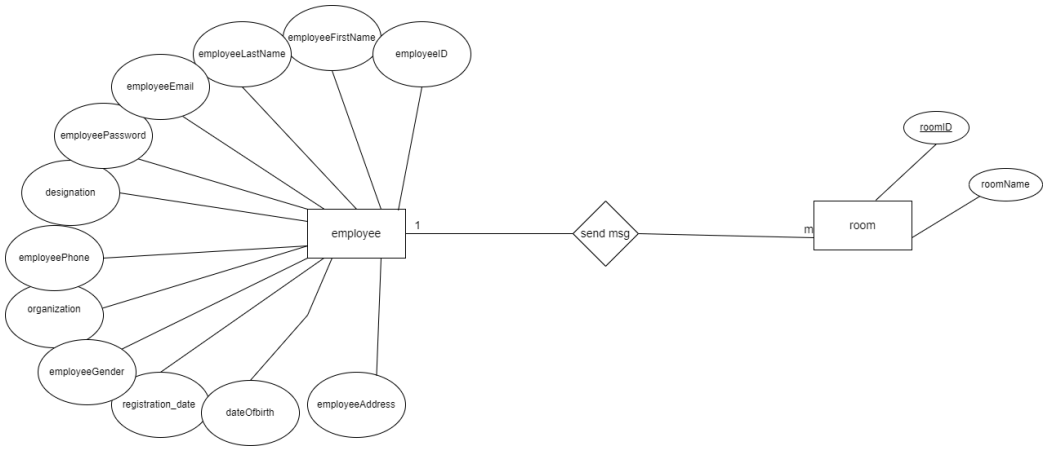
All non-key attributes are dependent on the primary keys on each table. It follows the 2nf form.

As there is no transitive dependency and dependencies are part of super key, it follows 3nf and BCNF form

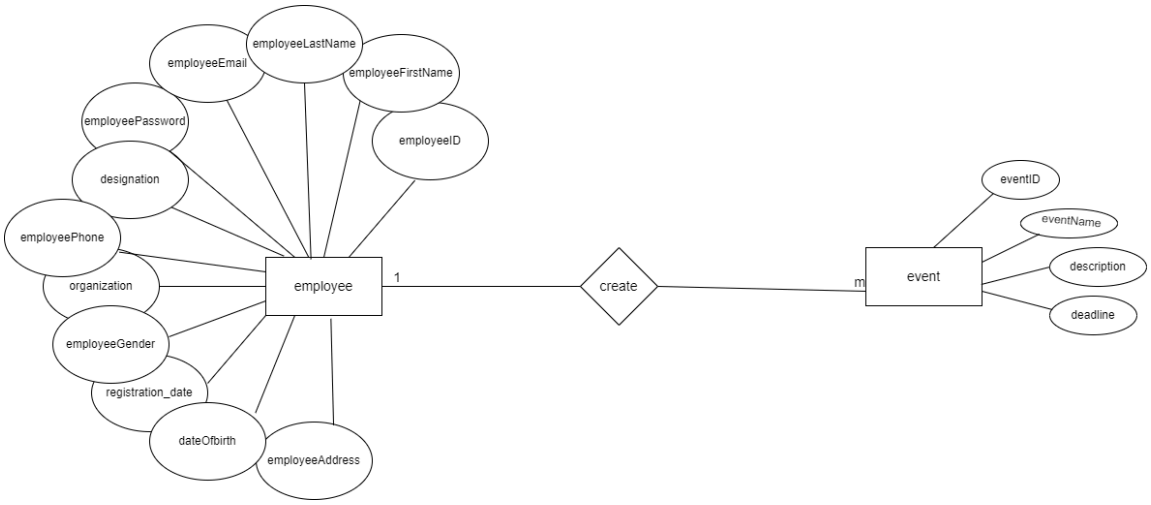
There is also no multivalued dependency and after decomposition, during natural join, no data will be lost. It follows the 4nf and 5nf form.

5.c.xv. **Entity Mapping**

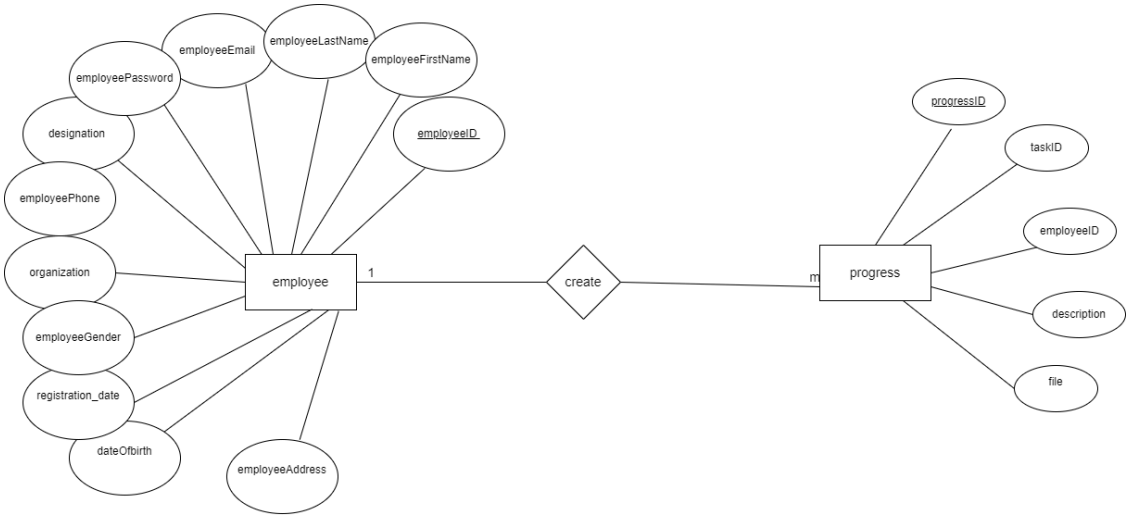
1. Employee Room Relationship (Many to Many)
   One employee can send messages to multiple rooms and One room can have multiple employees.



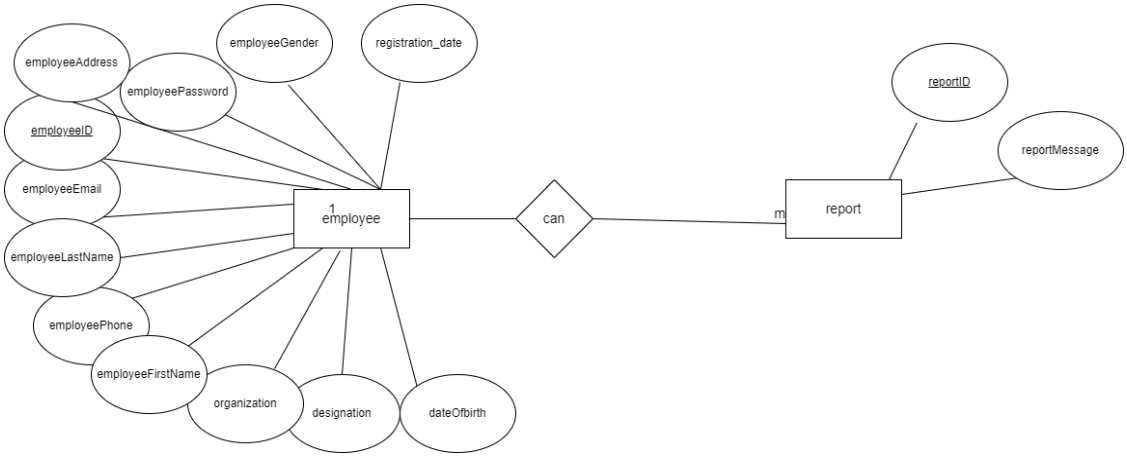
2. Employee Event Relationship (One to Many)
   One employee can create multiple events.



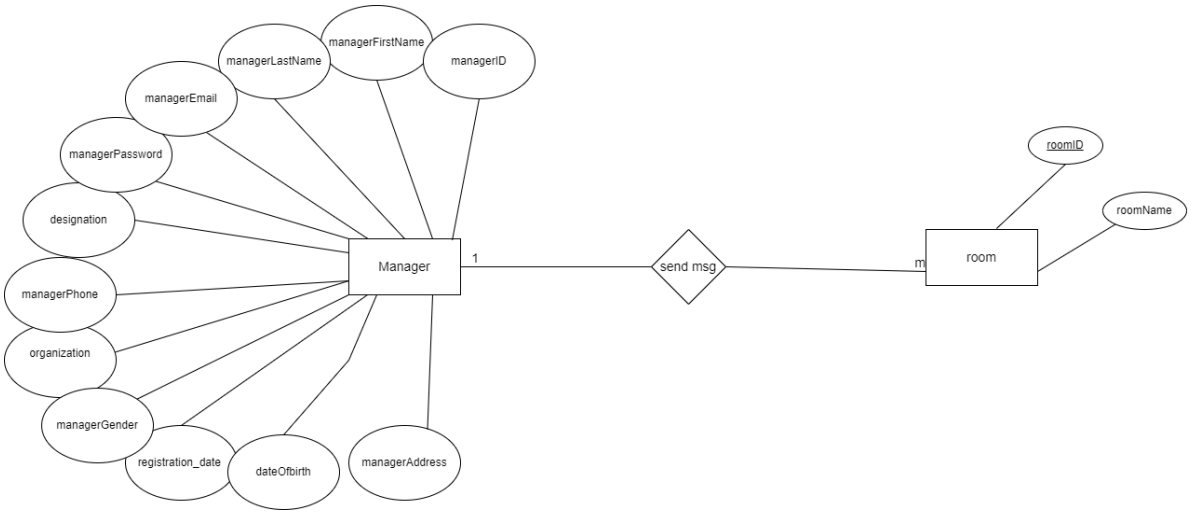
3. Employee Progress Relationship (One to Many)
   One employee can create multiple progresses for a singular task.



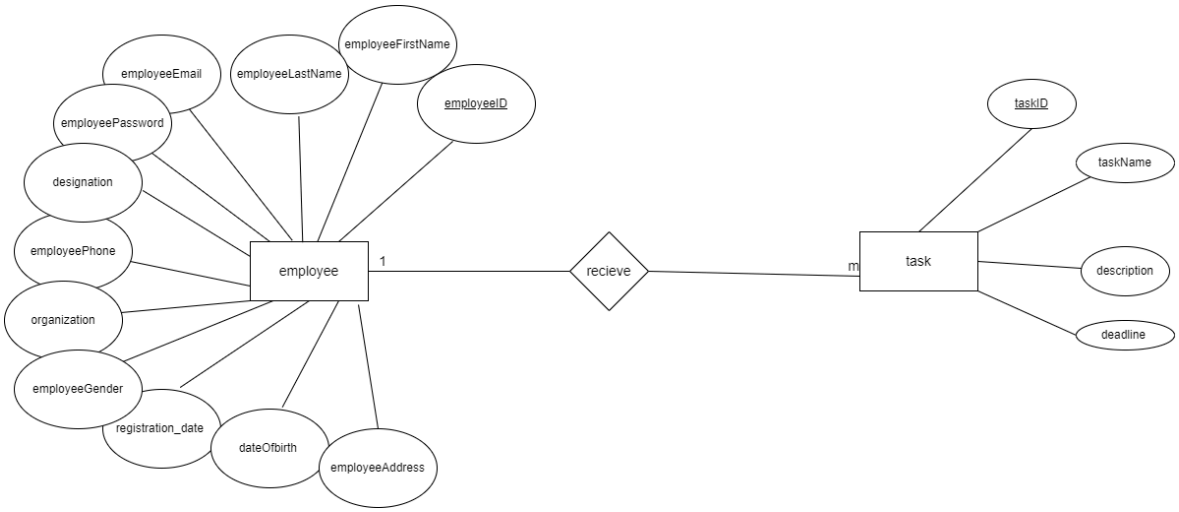
4. Employee Report Relationship (One to Many)
   One employee can report about managers multiple times.



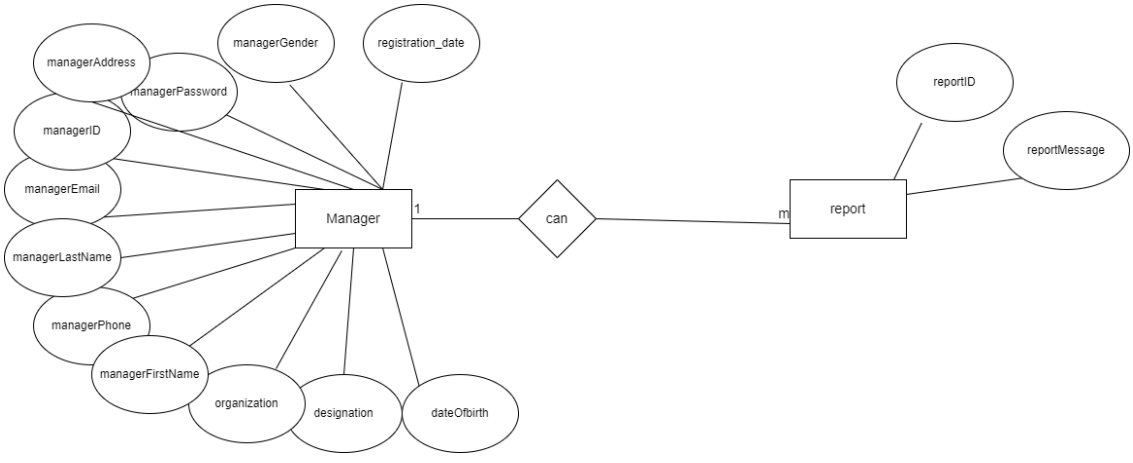
5. Manager Room Relationship (Many to Many)
   One manager can send messages to multiple rooms and One room can have multiple managers.



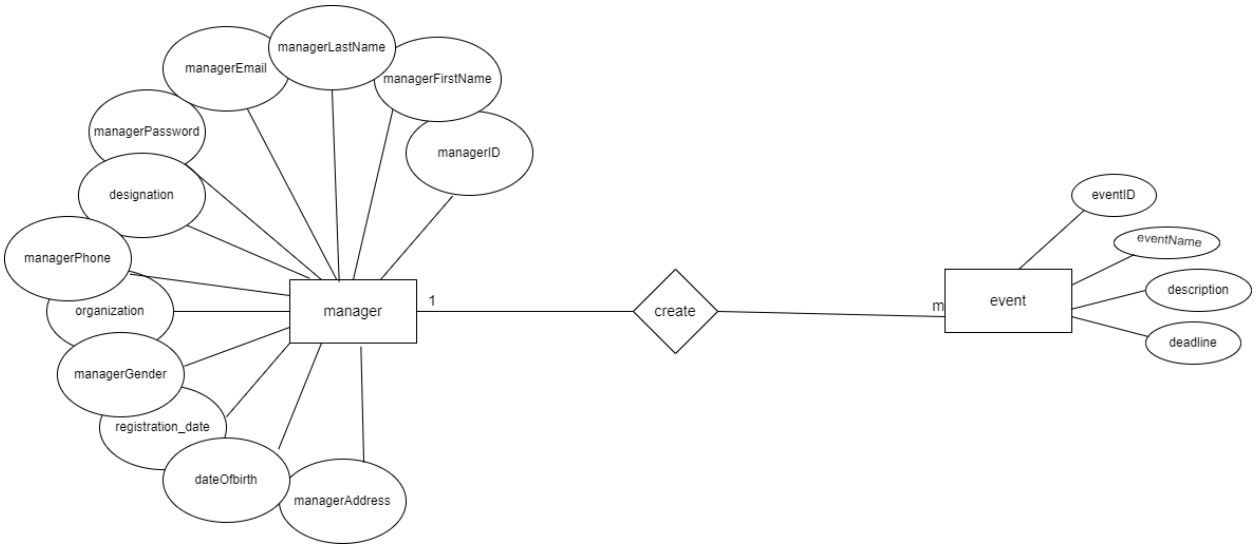
6. Employee Task Relationship (One to Many)
   One employee can receive multiple tasks from managers.



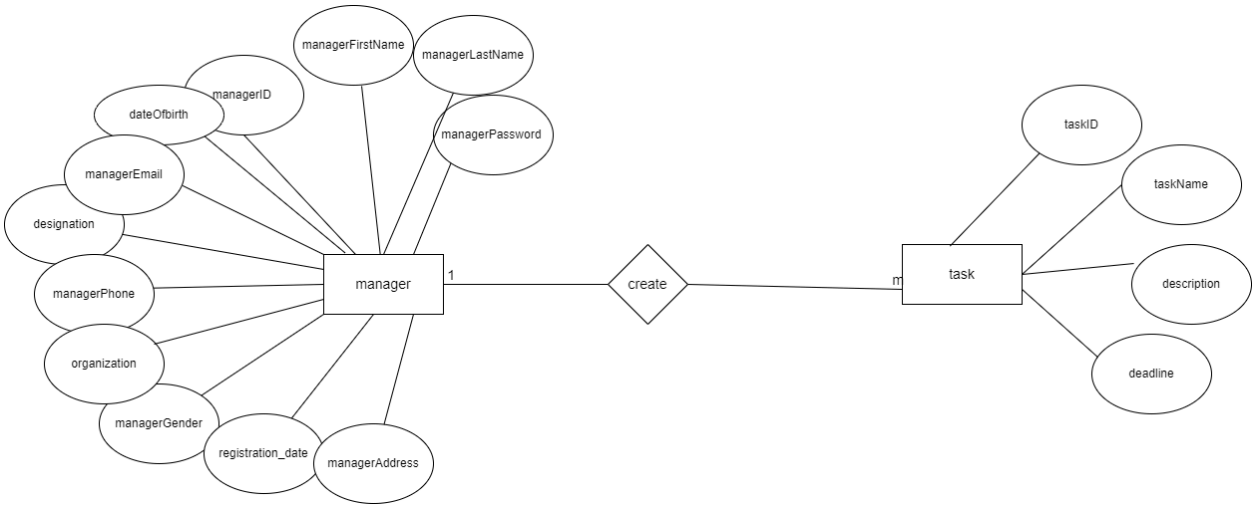
7. Manager Report Relationship (One to Many)
   One manager can report about employees multiple times.



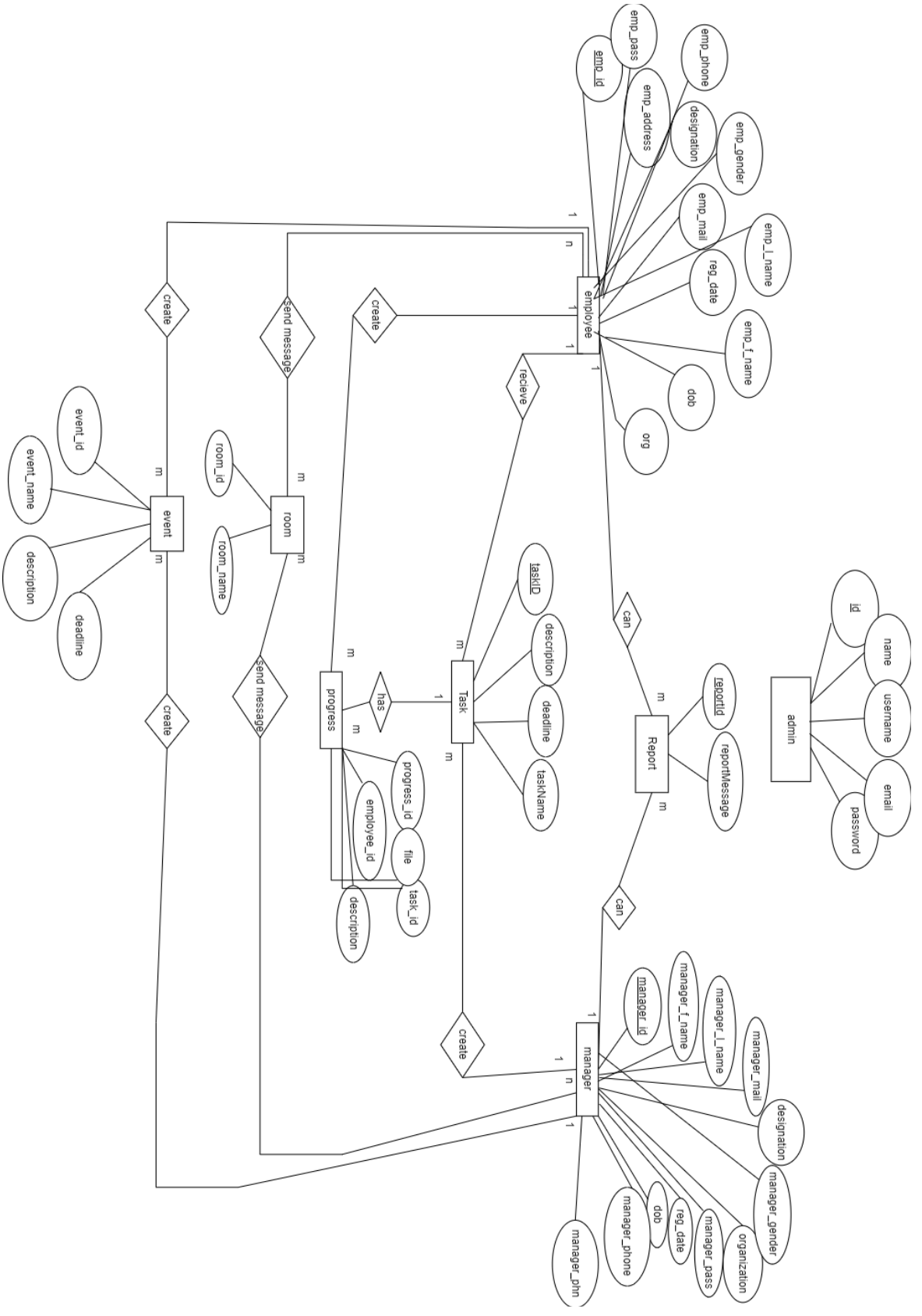
8. Manager Event Relationship (One to Many)
   One manager can create multiple events.



9. Manager Task Relationship (One to Many)
   One manager can create multiple tasks for employees.



5.c.xvi. **ERD**


emp_pass
emp_id
emp_phone
emp_address
designation
emp_gender
emp_mail
reg_date
emp_l_name
emp_f_name
dob
org
employee
create
send message
create
recieve
id
name
username
email
password
admin
can
reportId
reportMessage
Report
can
manager_id
manager_f_name
manager_l_name
manager_mail
designation
manager_gender
organization
manager_pass
reg_date
dob
manager_phone
manager_phn
manager
create
taskID
description
deadline
taskName
Task
has
progress
progress_id
employee_id
file
task_id
description
send message
create
room_id
room_name
room
event_id
event_name
description
deadline
event
create
1
n
m

6. **Construction**

6.a. **Development Environment**

a) **Framework**: ASP.NET
b) **Software**: Visual Studio
c) **Language**: C#, Html, Css
d) **Database**: MS SQL

6.b. **Testing Strategy**

In order to test the software, we supplied our beta version to certain people. They have tested it and reviewed the system. While testing, some bugs had arrived which we fixed after testing. While testing our system

We followed below testing strategies:

i) Static Testing Strategy
ii) Structural Testing Strategy
iii) Behavioral Testing Strategy

6.c. **Testing techniques**

Testing is a critical component of the development of any software. The goal of testing is to ensure that the software functions as intended, meets customer requirements, and is of high quality. The following are some of the key testing techniques used in the development:

1. Manual Testing: Manual testing involves verifying the software manually, using a test plan and test cases. This testing technique is used

to verify the functionality and performance of the software, and to validate that it meets the requirements and specifications.

2. Automated Testing: Automated testing involves using software tools to automate the testing process. This testing technique is used to verify

the functionality, performance, and reliability of the software, and to increase the efficiency and accuracy of the testing process.

3. Functional Testing: Functional testing focuses on verifying that the software functions as intended, and meets the customer requirements. This testing involves verifying the input and output of the software, and testing the software in various conditions and scenarios.

4. Performance Testing: Performance testing focuses on verifying that the software meets the performance requirements, including response time, scalability, and capacity. This testing involves simulating real world workloads and conditions, and verifying that the software meets the performance targets.

5. Security Testing: Security testing focuses on verifying that the software is secure and meets the security requirements, including the protection of sensitive data and customer information. This testing involves using various techniques, including penetration testing, vulnerability scanning, and threat modeling, to identify and mitigate security risks.

6. Usability Testing: Usability testing focuses on verifying that the software is usable, intuitive, and meets the needs of the users. This testing involves testing the software with representative samples of users, and gathering feedback on the user experience and interface.

7. **Deployment**

7.a. **Deployment**

This application is a task management system for now. But in the future, we surely have better plans for it. So, for now, for the deployment part, users will log in for their use. We will maintain everything from our server.

7.b. **AMC (Annual Maintenance Contract)**

AMC is a maintenance contract or an insurance policy for your technological advancement. It is a software update service for which your company has to pay annually, to the software provider. In our project we will always check how it is performing in other devices of different organizations.

7.c. **Support and Maintenance**

Software maintenance is part of the software development life cycle. Here we will modify and continuously update software applications to eliminate all possible errors, malfunctions, to improve work efficiency and better system performance.

8. **Learning Experiences**

While we were doing the project we acquired a lot of skills and knowledge individually and we gathered a lot of experience as a team. We learnt to follow a strong schedule, and strictly maintain a working model and procedures, which helped in improving our communication and teamwork skills. We were familiar with testing, debugging and improving our code and the work of our team members to develop the software on schedule which will be very helpful for ourselves.

9. **Conclusion**

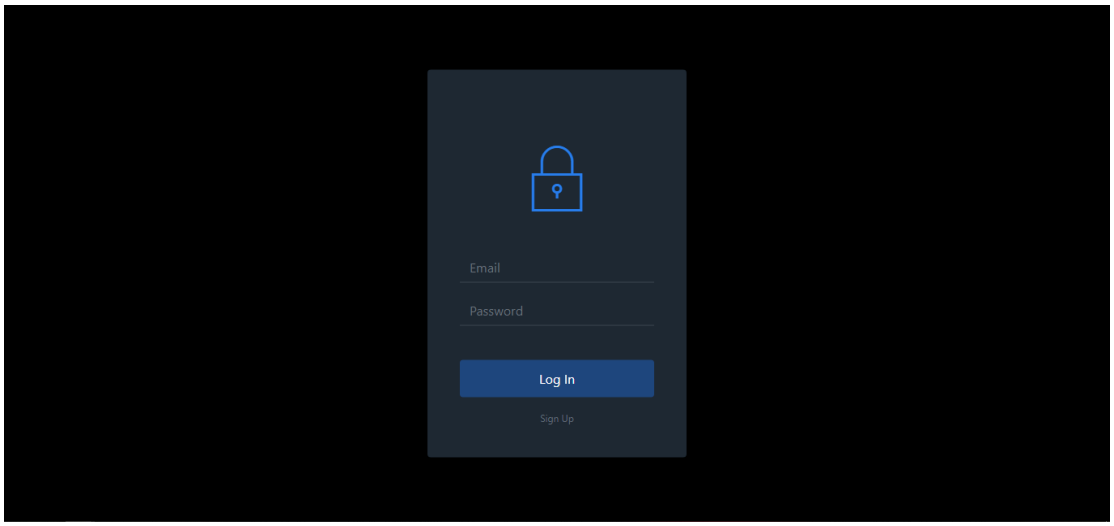
Our platform aims to target those people who want their work organized. Our goal is to provide managers and employees a hassle-free work experience. In conclusion, our task management platform is an innovative and practical solution for the workers who want to make their work easy. Our team has developed a user-friendly and comprehensive platform that aligns with current industry standards, and we are confident that it will be well-received by our target market.

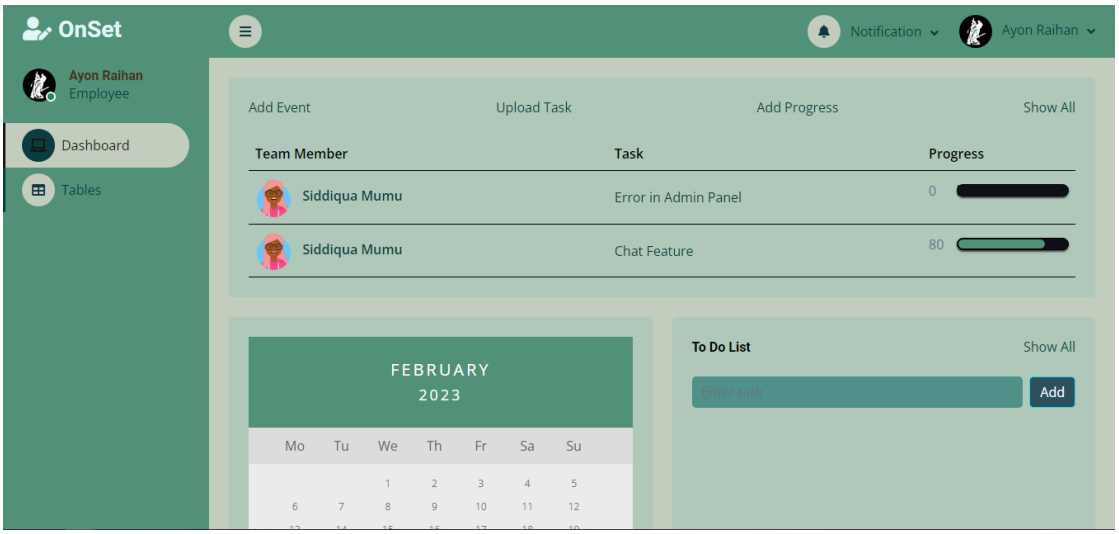
Bibliography/References:

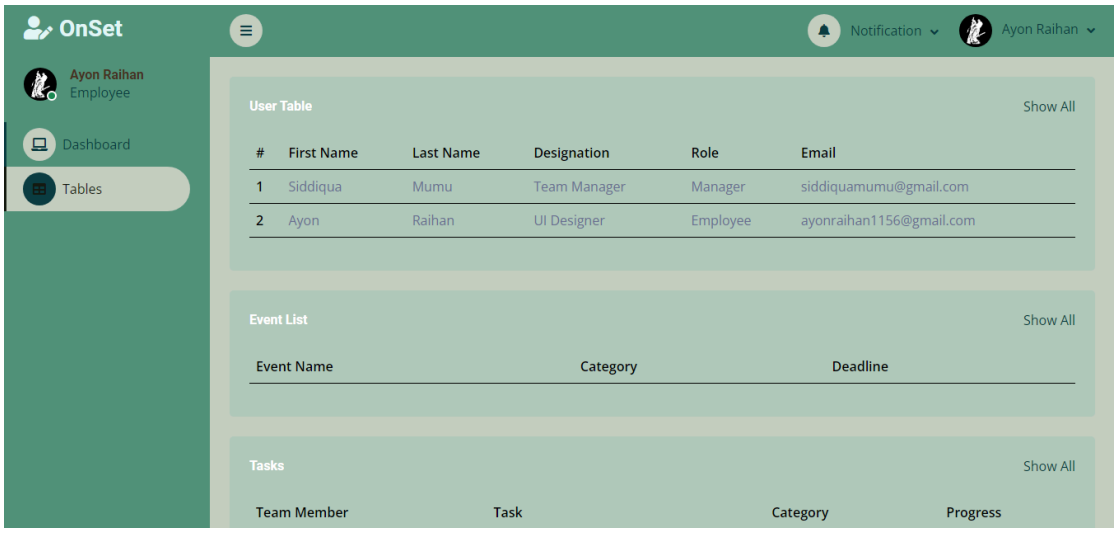
Lutkevich, B. and Lewis, S. (2022) What is the waterfall model? - definition and guide, Software Quality. TechTarget. Available at: https://www.techtarget.com/searchsoftwarequality/definition/waterfall-model (Accessed: February 20, 2023).

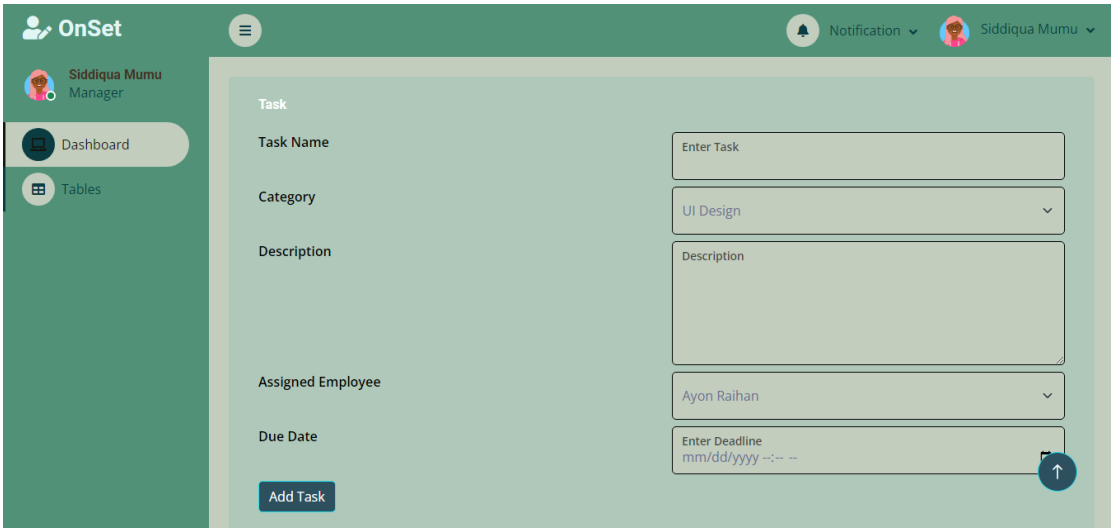
Pressman, R.S. (2009) Software engineering: A practitioner's approach. 8th edition. Dubuque, IA: McGraw-Hill.

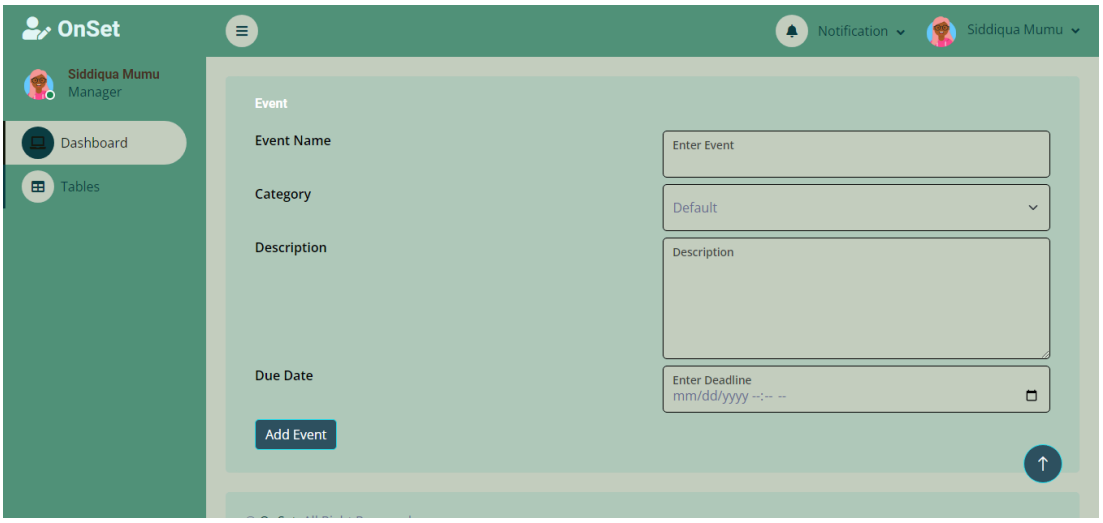
Appendix:

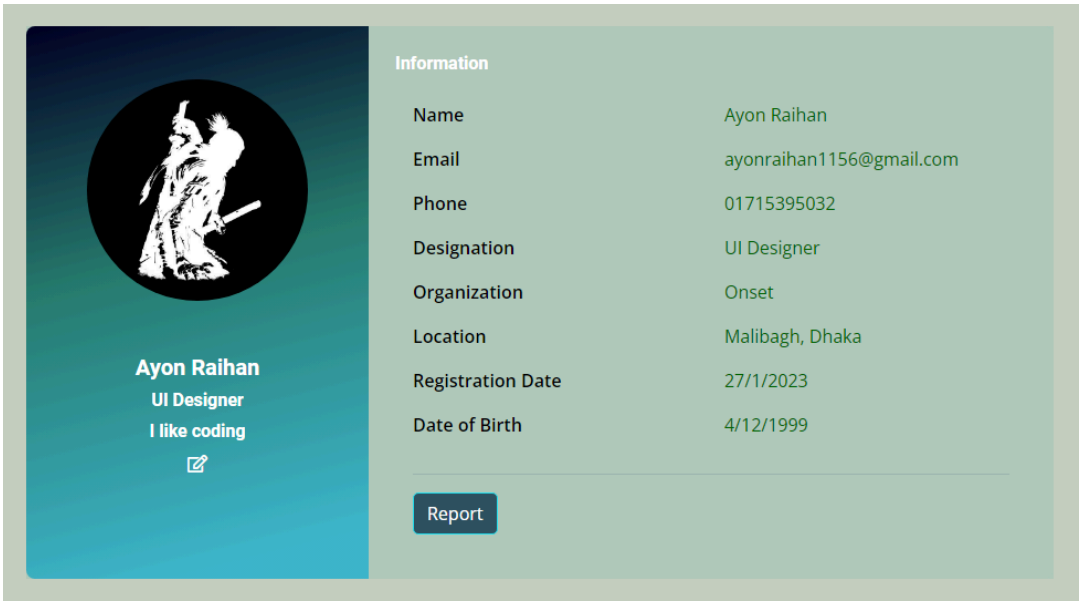

Email
Password
Log In
Sign Up


OnSet
Notification
Ayon Raihan
Ayon Raihan
Employee
Dashboard
Tables
Add Event
Upload Task
Add Progress
Show All
Team Member
Task
Progress
Siddiqua Mumu
Error in Admin Panel
0
Siddiqua Mumu
Chat Feature
80
FEBRUARY
2023
Mo Tu We Th Fr Sa Su
To Do List
Show All
Enter task
Add


OnSet
Notification
Ayon Raihan
Ayon Raihan
Employee
Dashboard
Tables
User Table
Show All
First Name Last Name Designation Role Email
1 Siddiqua Mumu Team Manager Manager siddiquamumu@gmail.com
2 Ayon Raihan UI Designer Employee ayonraihan1156@gmail.com
Event List
Show All
Event Name
Category
Deadline
Tasks
Show All
Team Member
Task
Category
Progress


OnSet
Notification
Siddiqua Mumu
Siddiqua Mumu
Manager
Dashboard
Tables
Task
Task Name
Enter Task
Category
UI Design
Description
Description
Assigned Employee
Ayon Raihan
Due Date
Enter Deadline
mm/dd/yyyy --:-- --
Add Task


OnSet
Notification
Siddiqua Mumu
Siddiqua Mumu
Manager
Dashboard
Tables
Event
Event Name
Enter Event
Category
Default
Description
Description
Due Date
Enter Deadline
mm/dd/yyyy --:-- --
Add Event


Ayon Raihan
UI Designer
I like coding
Information
Name
Ayon Raihan
Email
ayonraihan1156@gmail.com
Phone
01715395032
Designation
UI Designer
Organization
Onset
Location
Malibagh, Dhaka
Registration Date
27/1/2023
Date of Birth
4/12/1999
Report