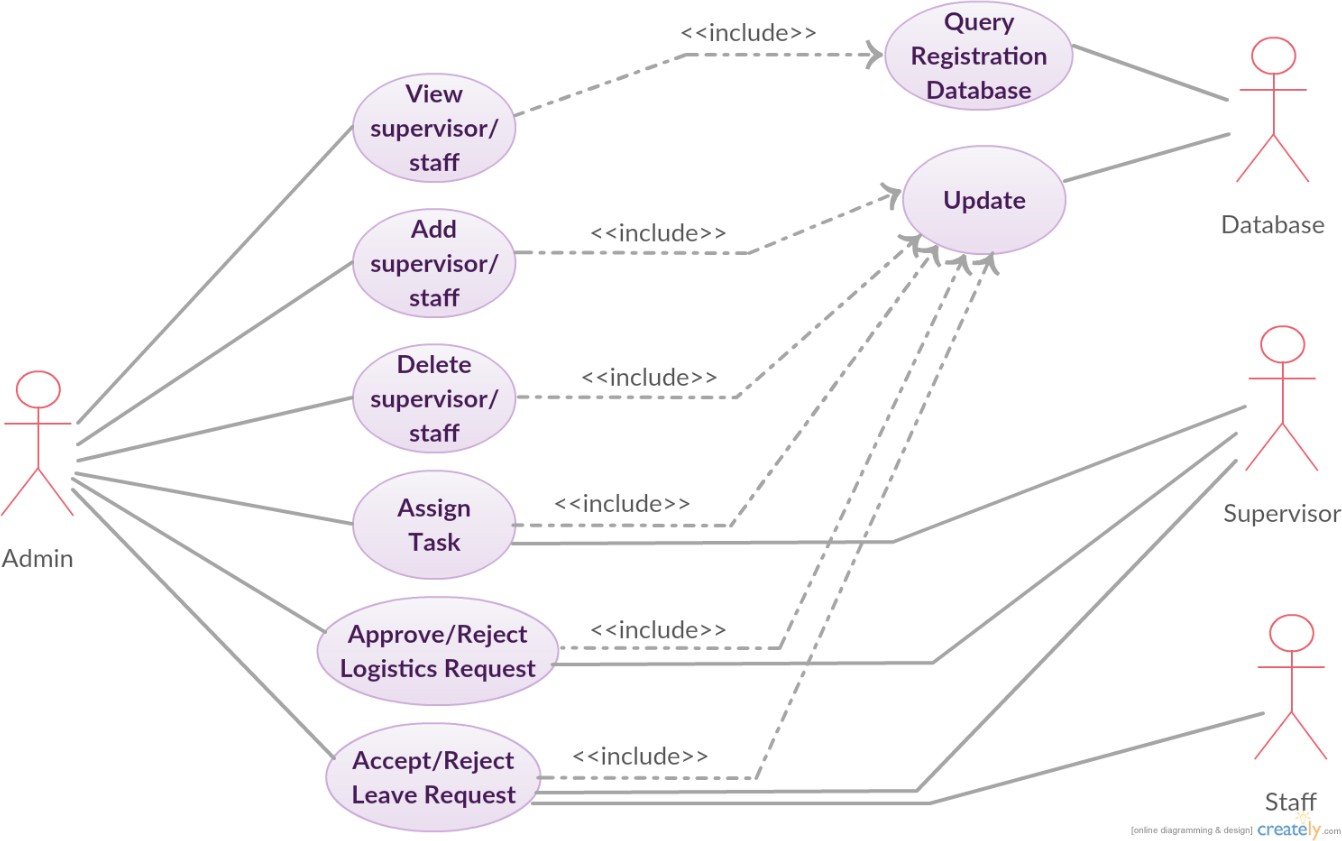
FACILITY MANAGEMENT SERVICES SYSTEM

Design Analysis

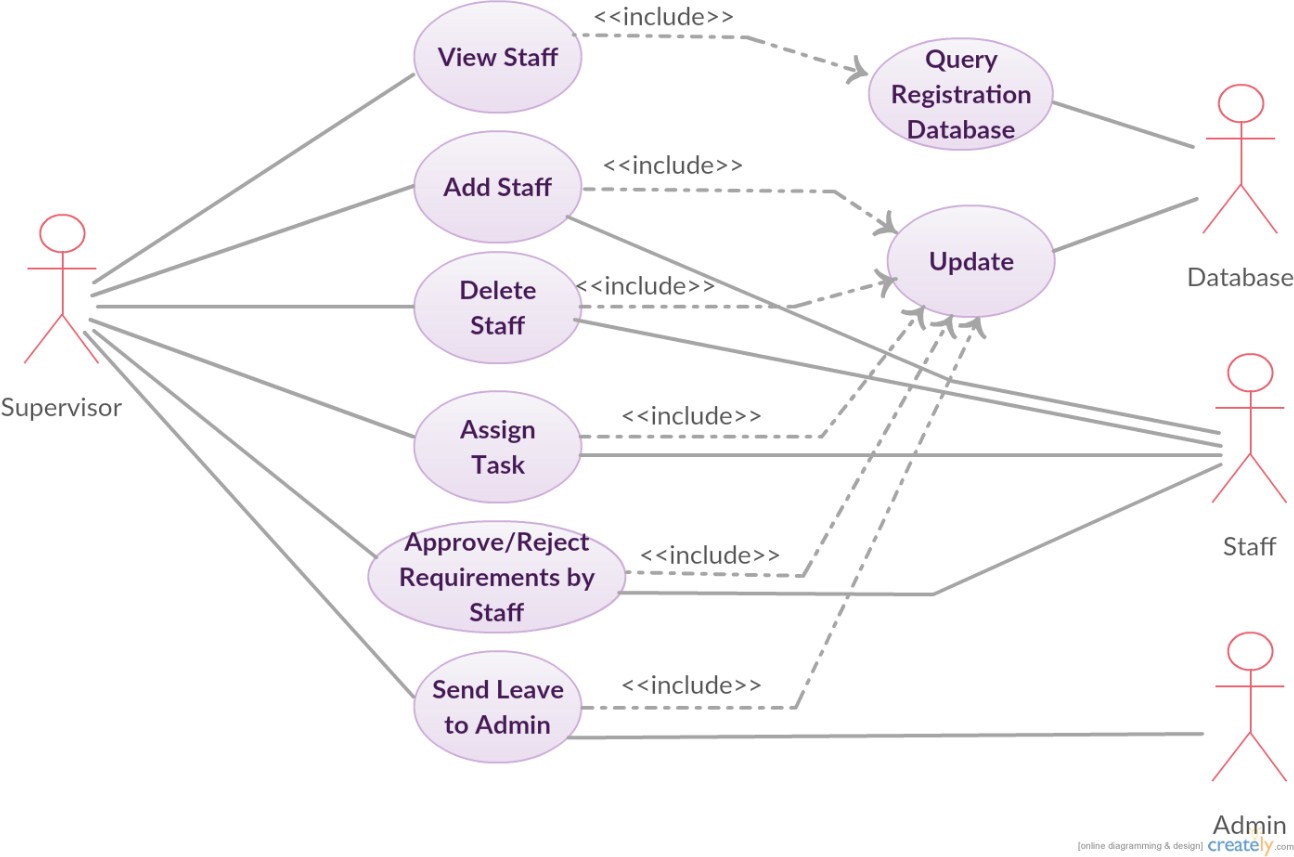
# Use Case Diagrams:

1. Functionalities of the Employee/Admin

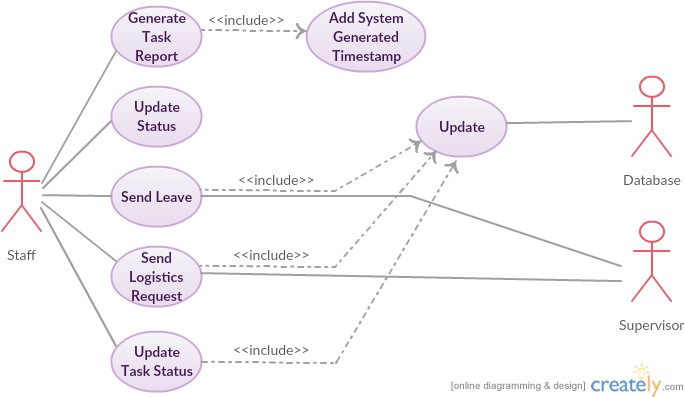


This diagram illustrates the functionalities of the administrator, which include viewing, adding, and deleting staff and supervisors (since everything visible to the staff and supervisors is also visible to the admin). Also, it can assign tasks to the supervisor, approve/reject logistics, and accept/reject leave requests of staff and supervisors. All these include updating of database.

1. Functionalities of the Supervisors

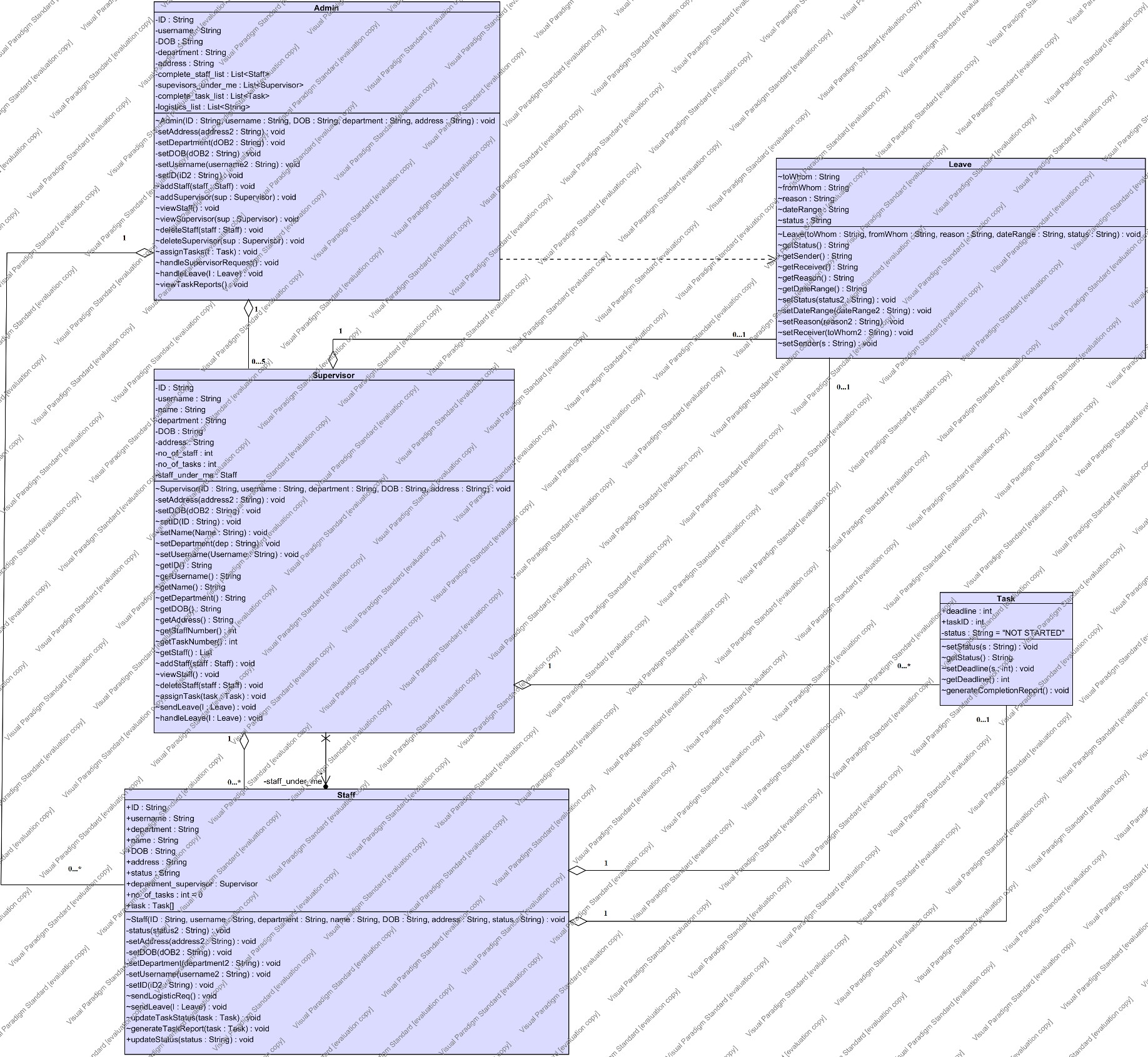


This diagram has the use cases for supervisor- viewing, adding and deleting staff, assigning tasks to staff and handling logistics requests and leave requests of staff. Database is also updated in these processes. It can also send leave to the admin.

1. Functionalities of Staff

The diagram shows the functionalities of the staff, such as generating task reports, updating status (available/busy), sending leave requests to supervisor, sending logistics request to supervisor and updating task status.

# Class Diagram:



## Admin Class:

**private** String ID, username,DOB,department,address;

**private** List<Staff> complete\_staff\_list; **private** List<Supervisor> supevisors\_under\_me; **private** List<Task> complete\_task\_list; **private** List<String> logistics\_list;

* Setters for admin---initialises all the fields of the admin
* void addStaff(Staff staff)---add a new member in the staff
* void addSupervisor(Supervisor sup)---adds the supervisor in any of the five departments
* void viewStaff()---views the complete staff with information of each of them
* void viewSupervisor()---views 5 supervisors and their respective departments
* void deleteStaff(Staff staff)---can delete any staff member
* void deleteSupervisor(Supervisor sup)---can delete any supervisor
* void assignTasks(Task t)---Assign task to supervisor and staff
* void handleSupervisorRequest()---Accepts or rejects requests for inventory
* void viewTaskReports()---views tasks assigned to each person

## Supervisor Class:

**private** String ID,username,name,department,DOB,address;

**private** List<Staff> staff\_under\_me;

**private int** no\_of\_staff,no\_of\_tasks;

* Getters and setters for supervisor---initialises all the fields and returns field data when required
* void addStaff()---add a new member in the staff
* void viewStaff()---views the complete staff with information of each of them
* void deleteStaff()---can delete any staff member
* void assignTasks()---Assign task to staff
* void sendleave()---can send leave to supervisor
* void handleleave()---can accept and reject leaves sent by staff
* void sendrequest()---sends requests to admin for inventory.

## Staff Class:

**public** String ID, username,department,name,DOB,address,status;

**public** Supervisor department\_supervisor;

**public int** no\_of\_tasks=0;

**public** Task task[];

* Getters and setters for all the fields of the staff---sets the value for all the fields and returns the data of the required field when needed
* void updatestatus(String status2)---updates the status of the staff member
* void sendLogisticReq()---send logistic requirement to supervisor/admin
* void sendLeave(Leave l)---requests for a leave to supervisor/admin
* void generateTaskReport(Task task)---generates a task report for each staff member with respect to their department

## Leave Class:

String toWhom,fromWhom,reason,dateRange,status;

* Getters and setters for all the fields

## Task Class:

**public int** deadline,taskID;

**private** String status="NOT STARTED";

* Getters and setters for different variables.
* void GenerateCompletionReport()---generates completion report when task status is complete