Submission Deadline: 23.04.2025, 23:59

Dear Students,

In Phase III of our project, we will focus on software design and modeling. Below are the requirements for this phase, explained in an easy-to-understand format:

Software Design and Modeling

Group Name: BIBA X COMPANY / BOB NDËRTUESI

Software Architecture

System Architecture:
The "BIBA X" web app follows a **three-tier architecture** based on the **MVC (Model-View-Controller)** pattern:

- **Client Layer (Frontend):**
  Built using HTML, CSS, JavaScript, and Bootstrap. It handles user interaction and sends requests to the backend.

- **Server Layer (Backend):**
  Developed using PHP (optionally Node.js for real-time messaging). It manages business logic, authentication, and task delegation.
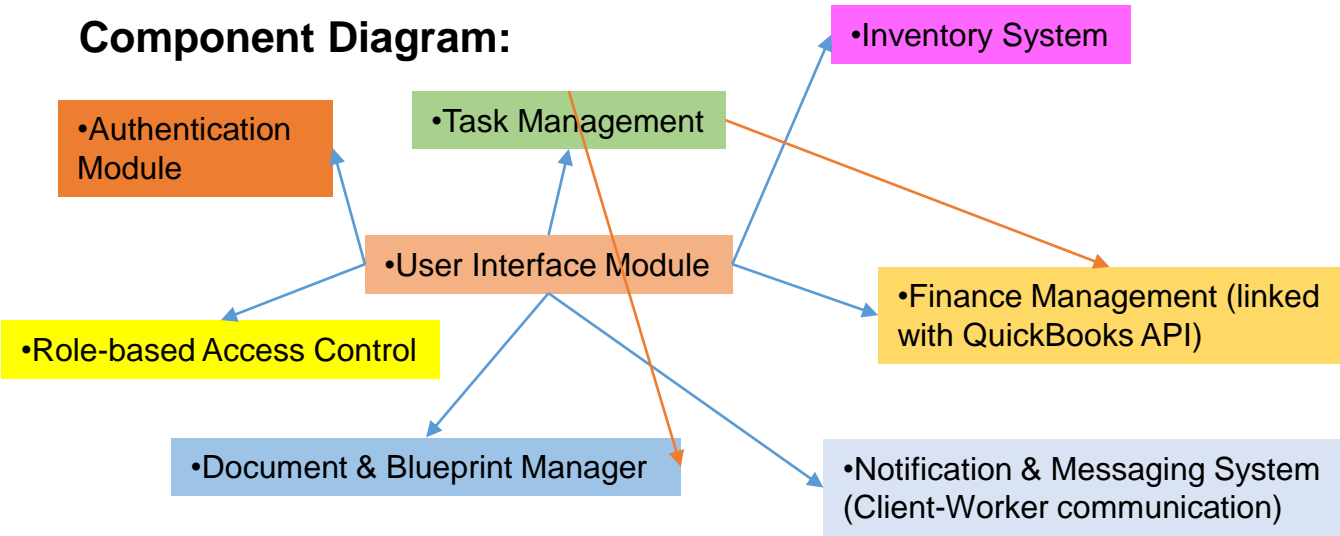
- **Database Layer:**
  MySQL with phpMyAdmin to manage project data, user accounts, tasks, finances, and inventory.
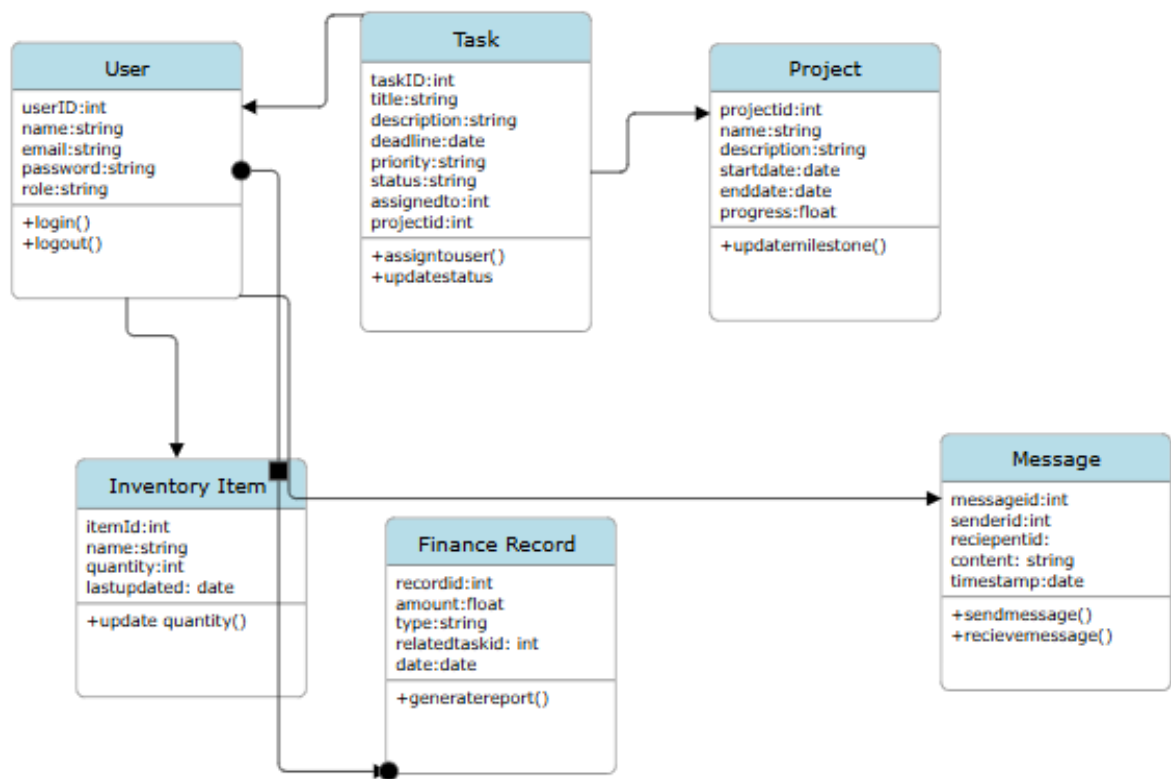
- **Communication & Security:**
  Data exchange via RESTful APIs secured with SSL/TLS. JWT/session tokens(optional) for secure login. Role-based access control ensures data integrity.

Component Diagram:

Authentication Module

Task Management

Inventory System

User Interface Module

Role-based Access Control

Finance Management (linked with QuickBooks API)

Document & Blueprint Manager

Notification & Messaging System (Client-Worker communication)

Detailed Design

Class Diagram:

User
userID:int
name:string
email:string
password:string
role:string

+login()
+logout()

Task
taskID:int
title:string
description:string
deadline:date
priority:string
status:string
assignedto:int
projectid:int

+assigntouser()
+updatestatus

Project
projectid:int
name:string
description:string
startdate:date
enddate:date
progress:float

+updatemilestone()

Inventory Item
itemId:int
name:string
quantity:int
lastupdated: date

+update quantity()

Finance Record
recordid:int
amount:float
type:string
relatedtaskid: int
date:date

+generatereport()

Message
messageid:int
senderid:int
reciepentid:
content: string
timestamp:date

+sendmessage()
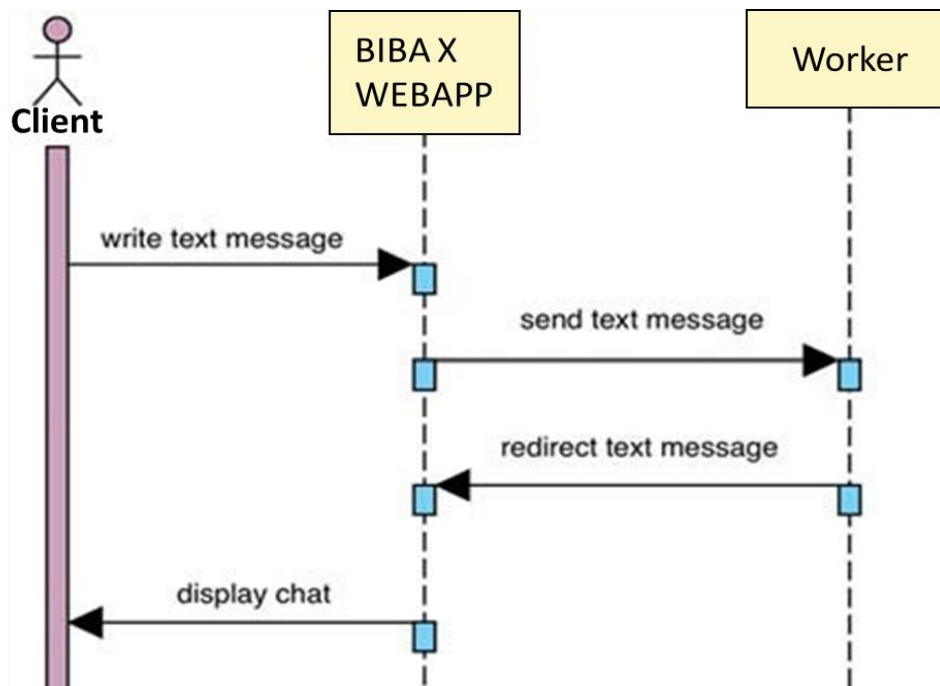+recievemessage()

Sequence Diagrams:

A) TASK ASSIGNMENT FLOW
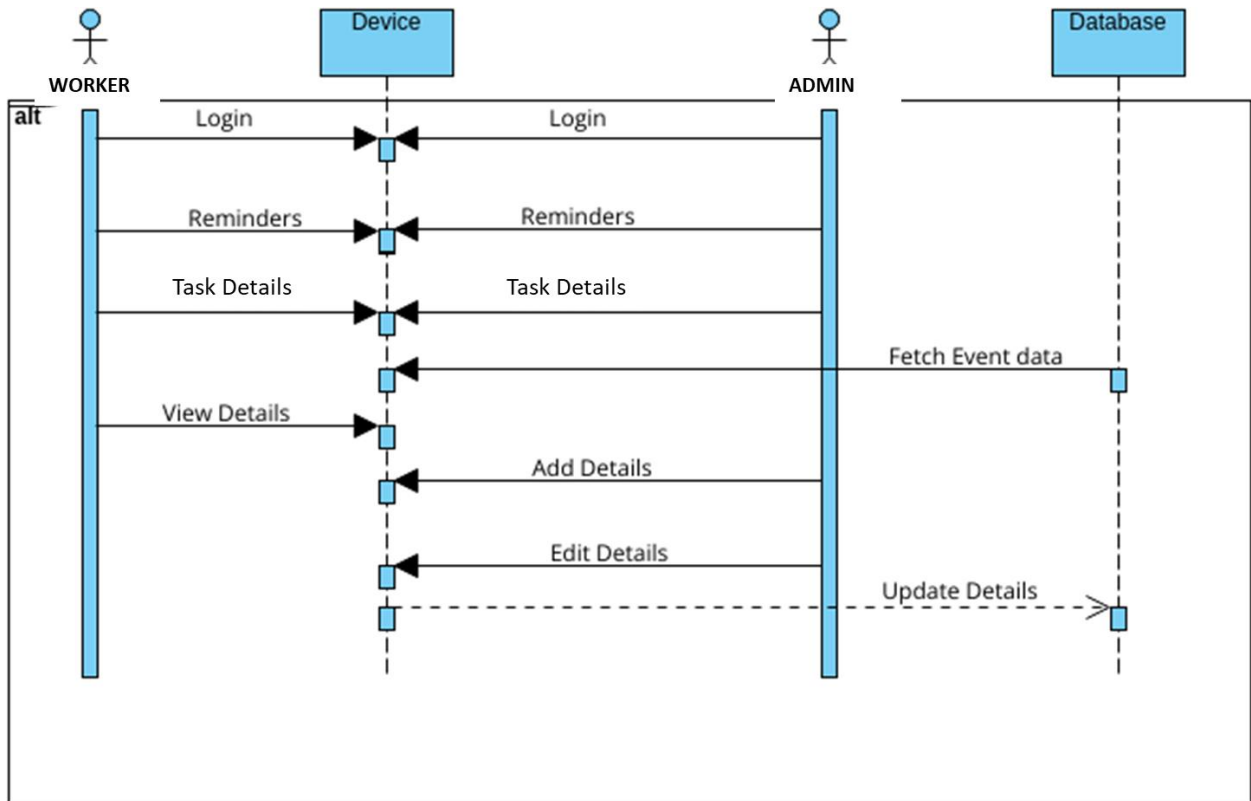
B) CLIENT-WORKER COMMUNICATION

Database Design:
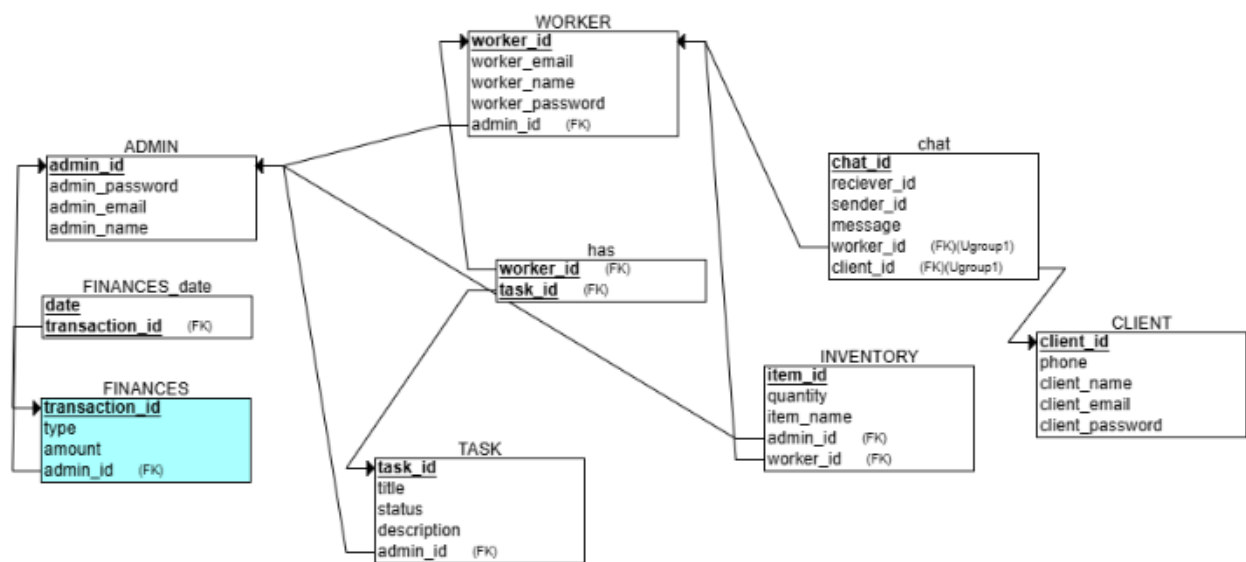
Modeling

Use Case Diagram:
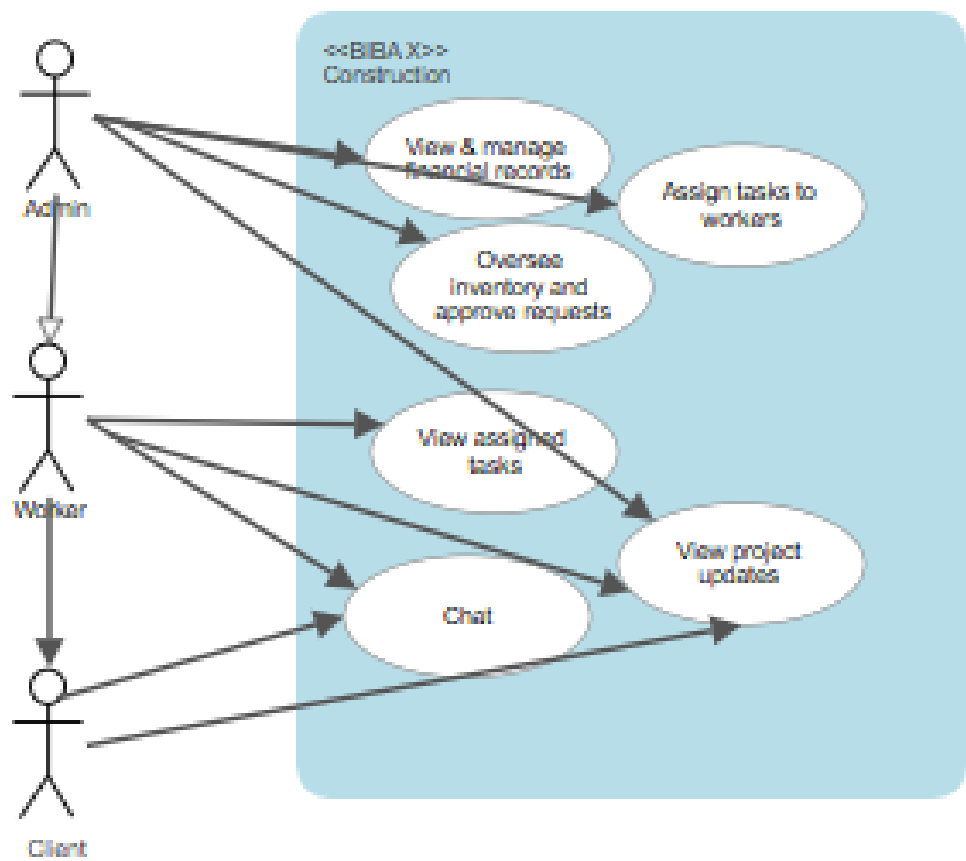
Activity Diagrams:

Task Completion Process Steps:

```
Log in → View Assigned Tasks → Mark progress → Submit → Mark as Completed
                                                                ↓
                                                          Admin checks it
```

```
(start) → Worker logs in → Worker checks assigned task → Worker checks material inventory → Is material available ?
```

Is material available ? → Submit material request to admin → Admin receives request → Admin reviews request → Approve or reject request → Notify worker of approval/rejection → Worker updates task progress → (end)

Worker checks material inventory → Proceed with task → Update progress → (end)

State Diagrams:

A **state diagram** shows the **different states** an object can be in during its lifecycle and how it **transitions** between those states based on events or actions.
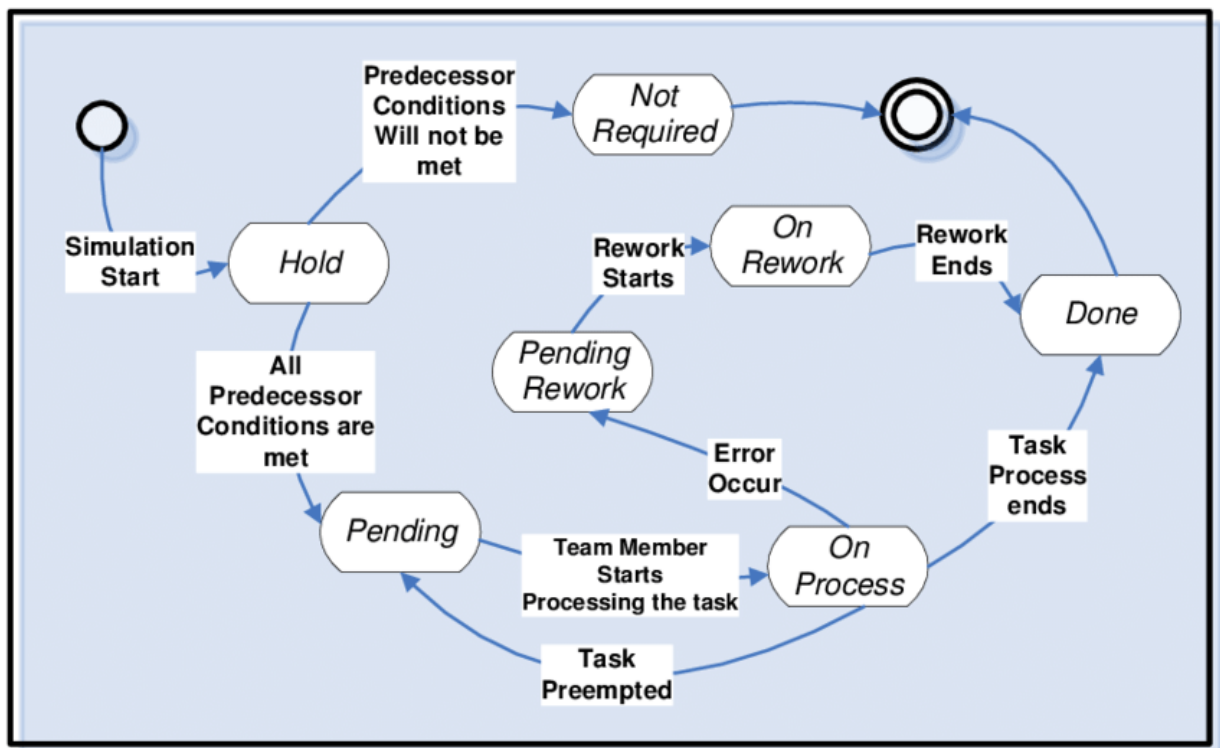
Task Prioritization and Allocation

```
                    ●
                    │
                    ▼
         ┌──────────────────────┐
         │ Define Task Priorities│
         └──────────────────────┘
                    │
          Yes       ▼         No
        ┌────◇ Are priorities defined? ◇────┐
        │                                    │
        ▼                                    ▼
┌──────────────────────┐          ┌──────────────────────┐
│ Allocate Tasks Equitably│       │   Define Priorities   │
└──────────────────────┘          └──────────────────────┘
        │                                    │
 ═══════════════                             ▼
        │                         ┌──────────────────────┐
        ▼                         │ Review Task Assignments│
┌──────────────────────┐         └──────────────────────┘
│  Assign Critical Tasks │                   │
└──────────────────────┘                     ▼
        │                                    ◉
        ▼
┌──────────────────────────┐
│ Assign Non-Critical Tasks │
└──────────────────────────┘
        │
 ═══════════════
        │
        ▼
┌──────────────────────┐
│ Review Task Assignments│
└──────────────────────┘
        │
  Yes   ▼          No
 ┌──◇ Adjustments needed? ◇──┐
 │                            │
 ▼                            ▼
┌──────────────────┐   ┌──────────┐
│ Make Adjustments │   │ Continue │
└──────────────────┘   └──────────┘
        │                   │
        └──────►◇◄──────────┘
                │
                ▼
      ┌──────────────────┐
      │ Implement Tasks  │
      └──────────────────┘
                │
                ▼
                ◉
```