

---

## Design Document for WorkSync

---

Group 3\_Rasel\_2

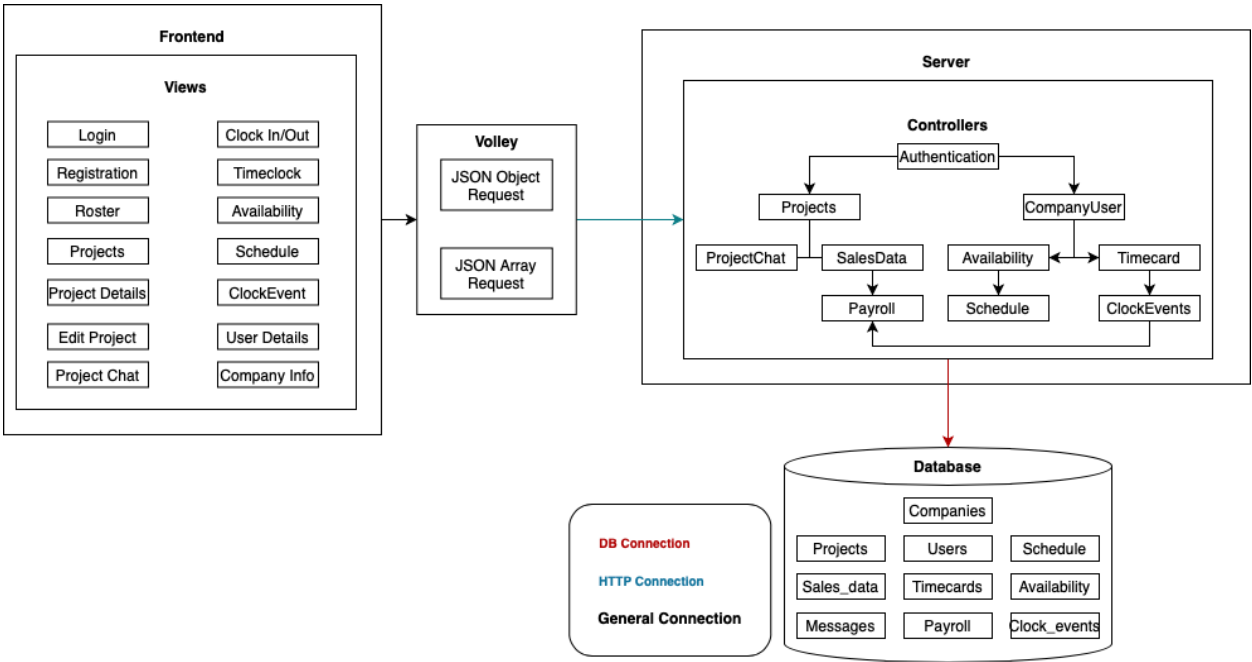
Nathan Willimack: 25% contribution

Miray Hirabayashi: 25% contribution

Srdan Kopunovic: 25% contribution

Cai Chen: 25% contribution

# Block Diagram



## Design Description

### Frontend

#### *TimeCard* (Manager, Employee)

- Time Card page generates a page with following elements:
  - TextView: time\_card\_text
  - TextView: check\_in\_label
  - Button: btnClockIn
  - Button: btnClockOut
  - TextView: latestTimeCard
- "time\_card\_text" TextView indicates that the user is currently on the Time Card page.
- The "check\_in\_label" TextView displays whether the user is currently checked in or not.
- Upon clicking the button “btnClockIn” the value of userID from the previous intent is sent as a POST request to the server.
- Upon clicking the button “btnClockOut” the value of userID from the previous intent is sent as a POST request to the server. Additionally, a GET request is made to the server to update the latestTimeCard TextView.

#### *CompanyRoster* (Owner, Executive)

- Company Roster page generates a page with following elements:
  - Button: fetch\_employee\_button
  - Button: add\_employee\_button
  - Button: edit\_employee\_button
  - Button: delete\_employee\_button
  - EditText: employee\_id\_input
  - ListView: employee\_list
  - FrameLayout: fragment\_container
- Upon clicking the button “fetch\_employee\_button” the value of the CompanyID from the previous intent is sent as a GET request to the server and fills the “employee\_list” ListView accordingly.

- Upon clicking the button “add\_employee\_button”, the values from fragment\_container; FirstName, LastName, Email, role, password are sent as a POST request to the server.
- Upon clicking the button “edit\_employee\_button”, the value of Employee “employee\_id\_input” is sent as a PUT request to the server and opens the fragment\_container.
- Upon clicking the button “delete\_employee\_button”, the value of Employee “employee\_id\_input” is sent as a DELETE request to the server.

#### *Payroll (Owner, Executive)*

- Payroll page generates a page with following elements:
  - TextView: time\_card\_text
  - TextView: weekStatus
  - Button: btnRefresh
  - Button: btnEditWeek
  - TableLayout: payroll\_table
- “time\_card\_text” TextView indicates that the user is currently on the Payroll/Time Card page.
- “weekStatus” TextView indicates which week’s payroll is displayed.
- Upon clicking the button “btnRefresh”, the value of userID from the previous intent and data from date variable are sent as a GET request to the server and updates the payroll\_table.
- Upon clicking the button “btnEditWeek”, a calendar is displayed for the user to select a date.

## Database Schema

Table Name	Fields
Users	id, firstName, lastName, email, role, password
Projects	id, name, description, dueDate, ownerId
TimeCards	id, userId, clockIn, clockOut, totalHours
Schedules	id, userId, dayOfWeek, shiftStart, shiftEnd
Messages	id, projectId, senderId, messageContent, timestamp

## Backend

### *Overview*

Our backend, built primarily with Java and the Spring Boot framework, it manages both communication with the frontend and interactions with our MySQL database. We also use Maven for build automation and dependency management, Swagger Docs for generating API documentation, and MySQL as our relational database.

### *Communication*

To allow communication between the frontend and the MySQL database, we've implemented RESTful APIs. These endpoints support CRUD operations through these methods: Post - sends information (create), Get - receives information (read), Put - edits information (update), and Delete - deletes information (delete). To enable these operations, the following three core components are created for each table in the database:

1. A model that describes the columns of the table, along with getters and setters.
2. A repository that manages and accesses the database.
3. A controller that acts as an intermediary between the database and the frontend.

## Database Table and Fields

