Business Requirements Document (BRD)

Project Title: Cost Estimation for Home Renovation and Construction

---

1. Project Overview:

This application serves as a bridge between users seeking construction services and constructors providing cost estimates. Users can create requests detailing their construction needs, while constructors can view these requests and generate estimates based on various parameters, including material quality.

The application will include:

- **User Interface**: An intuitive platform for users to submit requests, view estimations, and manage their wishlists.
- **Constructor Dashboard**: A robust interface for constructors to view requests, manage estimates, and generate reports.
- **Backend System**: A reliable database system to store user requests, estimations, reports, and manage interactions between users and constructors.
- **Notification System**: Alerts to keep users informed about new estimations and updates from constructors.

2. Objectives:

- **User Request Creation**: Allow users to create detailed requests for cost estimations for their construction projects.
- **Constructor Dashboard**: Provide constructors with a dashboard to view user requests, manage their estimations, and access features like material quality grades (basic, premium, moderate).
- **Cost Estimation**: Enable constructors to estimate costs based on user requests and submit detailed reports.
- **Report Submission**: Facilitate the process for constructors to send cost estimation reports to users.
- **Multiple Estimations**: Allow users to receive and view multiple cost estimations from different constructors for comparison.
- **Wishlist Feature**: Implement a wishlist functionality for users to save different estimations for future consideration.
- **Area Listings**: Allow constructors to view various user requests based on geographic areas to optimize their estimations and services.
- **User Access to Reports**: Ensure users can easily access and view the reports generated by constructors.

3. Stakeholders:

Stakeholders

1. **Users**:
    - Individuals or businesses seeking construction services.
    - Interested in receiving multiple estimates to make informed decisions.
2. **Constructors**:
    - Construction service providers looking for new projects.

- Require tools to efficiently estimate costs and manage user requests.

3. **Developers**:
   - Responsible for the implementation of the application, including both front-end and back-end development.

3. Functional Requirements:

1. **User Registration and Login**:
   - Users should be able to register an account with email and password.
   - Users should be able to log in to their account securely.
2. **Request Creation**:
   - Users can create requests for cost estimations, including project details like area, type of construction, and preferred materials.
3. **Constructor Dashboard**:
   - Constructors can view a list of user requests in their dashboard.
   - Constructors can filter requests based on location, project type, and other criteria.
4. **Cost Estimation**:
   - Constructors can enter cost estimates based on user requests.
   - Constructors can choose material quality (basic, premium, moderate) to calculate costs.
5. **Report Generation**:
   - After completing the estimation, constructors can generate and submit reports to users.
   - Reports should include detailed cost breakdowns and project timelines.
6. **Multiple Estimates**:
   - Users can receive and compare estimates from multiple constructors for a single request.
7. **Wishlist Feature**:
   - Users can save multiple estimations to a wishlist for later reference.
8. **Report Access**:
   - Users can view and download reports generated by constructors at any time.
9. **Notifications**:
   - Users and constructors should receive notifications for important actions (e.g., new estimates, report submissions).
10. **User Profile Management**:
    - Users can update their profile information and preferences.
    - Constructors can manage their profiles and service offerings.

4. Non-Functional Requirements:

- **Performance**:
  - The application should handle concurrent requests from users and constructors with minimal latency.
  - Page load times should be less than 3 seconds under normal load.
- **Scalability**:
  - The system should be able to scale to accommodate an increasing number of users and constructors without performance degradation.
- **Security**:
  - User data should be protected through encryption and secure authentication methods.

- The application should implement measures to prevent common vulnerabilities (e.g., SQL injection, cross-site scripting).

- **Usability**:

  - The user interface should be intuitive and user-friendly, with clear navigation and accessible design.
  - The application should provide help and documentation for users and constructors.

- **Reliability**:

  - The application should have a high availability rate (99.9%) and minimal downtime.
  - Data backups should be performed regularly to prevent data loss.

5. User Stories

User stories for User

1. **User Registration**:
   - As a **user**, I want to register an account so that I can create requests for cost estimations.
2. **User Login**:
   - As a **user**, I want to log in to my account so that I can access my requests and estimates securely.
3. **Create Request**:
   - As a **user**, I want to create a request for a construction cost estimation so that I can get a quote for my project.
4. **View Estimates**:
   - As a **user**, I want to see multiple cost estimates from different constructors so that I can compare options.
5. **Select Constructor**:
   - As a **user**, I want to choose a constructor from the estimates I receive so that I can start the construction process.
6. **Access Reports**:
   - As a **user**, I want to view and download the reports generated by constructors so that I have a record of the estimates.
7. **Wishlist Feature**:
   - As a **user**, I want to add estimates to a wishlist so that I can save options I'm considering for later.
8. **Update Profile**:
   - As a **user**, I want to update my profile information so that my account details are current.
9. **Receive Notifications**:
   - As a **user**, I want to receive notifications when new estimates are submitted or when a constructor sends a report so that I am always informed.

User stories for Constructor

1. **Constructor Registration**:
   - As a **constructor**, I want to register an account so that I can view user requests and provide estimates.
2. **Constructor Login**:
   - As a **constructor**, I want to log in to my account so that I can access my dashboard and manage estimates securely.

3. **View User Requests**:
    - As a **constructor**, I want to view listings of user requests so that I can find potential projects to work on.
4. **Dashboard Features**:
    - As a **constructor**, I want to have a dashboard that includes features for estimating costs based on different grades of materials (basic, premium, moderate) so that I can provide accurate estimates.
5. **Submit Estimates**:
    - As a **constructor**, I want to submit cost estimates for user requests so that users can review my offers.
6. **Generate Reports**:
    - As a **constructor**, I want to generate detailed reports after completing estimates so that users receive comprehensive information about the costs involved.
7. **View Previous Listings**:
    - As a **constructor**, I want to view previous listings from users in my area so that I can identify trends and opportunities.
8. **Receive Notifications**:
    - As a **constructor**, I want to receive notifications when users create new requests or when my estimates are viewed so that I can stay engaged.
9. **Manage Profile**:
    - As a **constructor**, I want to manage my profile and service offerings so that I can present my business accurately to users.

Admin User story

1. **Admin Management**:
    - As an **admin**, I want to manage user and constructor accounts so that I can ensure proper use of the application.

6. Project Architecture:

The application follows a three-tier architecture with a microservices approach. The primary components are the Presentation Layer (Frontend), Business Logic Layer (Backend), and Data Layer (Database). The system and microservices will be registered via Eureka servers for service discovery and load balancing.

6.1 Presentation Layer (Frontend)

- **Framework**:
    - React with TypeScript
- **State Management**:
    - Redux for managing application state
- **Libraries**:
    - Material-UI or Bootstrap for UI components
- **Communication**:
    - RESTful APIs via HTTP to interact with backend services
        - This layer will handle user interaction for contractors, homeowners, and designers, allowing them to manage projects, access cost estimates, input project data, and view real-time information.

6.2 Business Logic Layer (Backend)

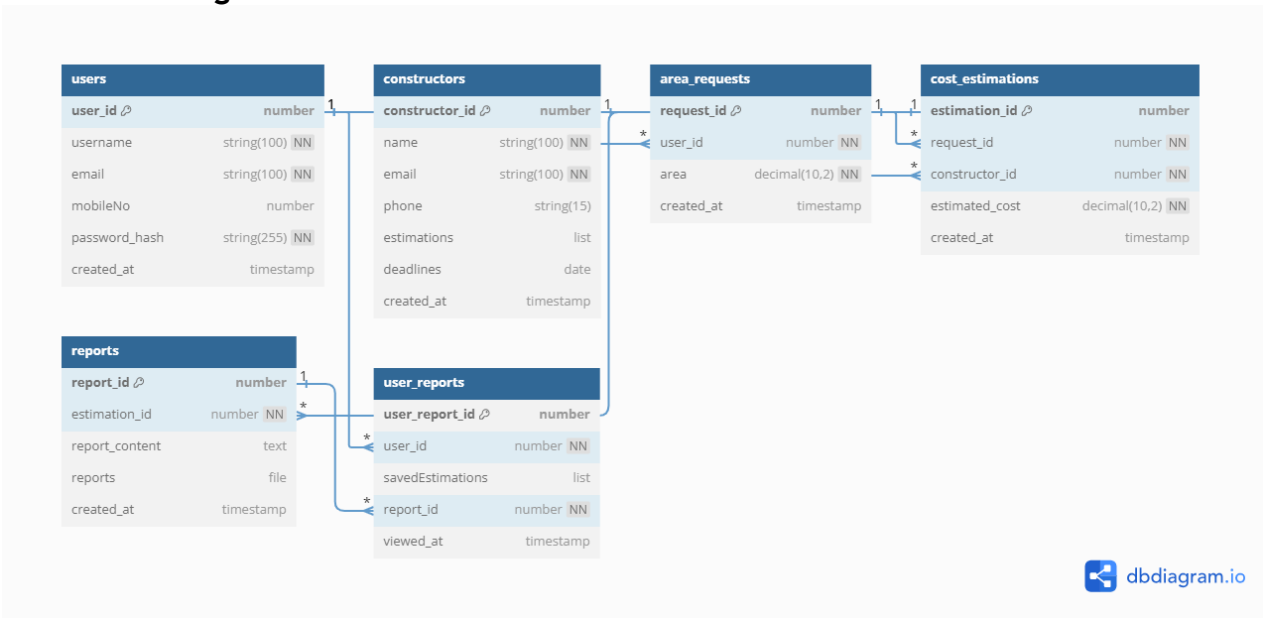
- Node.js with TypeScript (strictly)

- Each microservice handles a distinct business function such as:

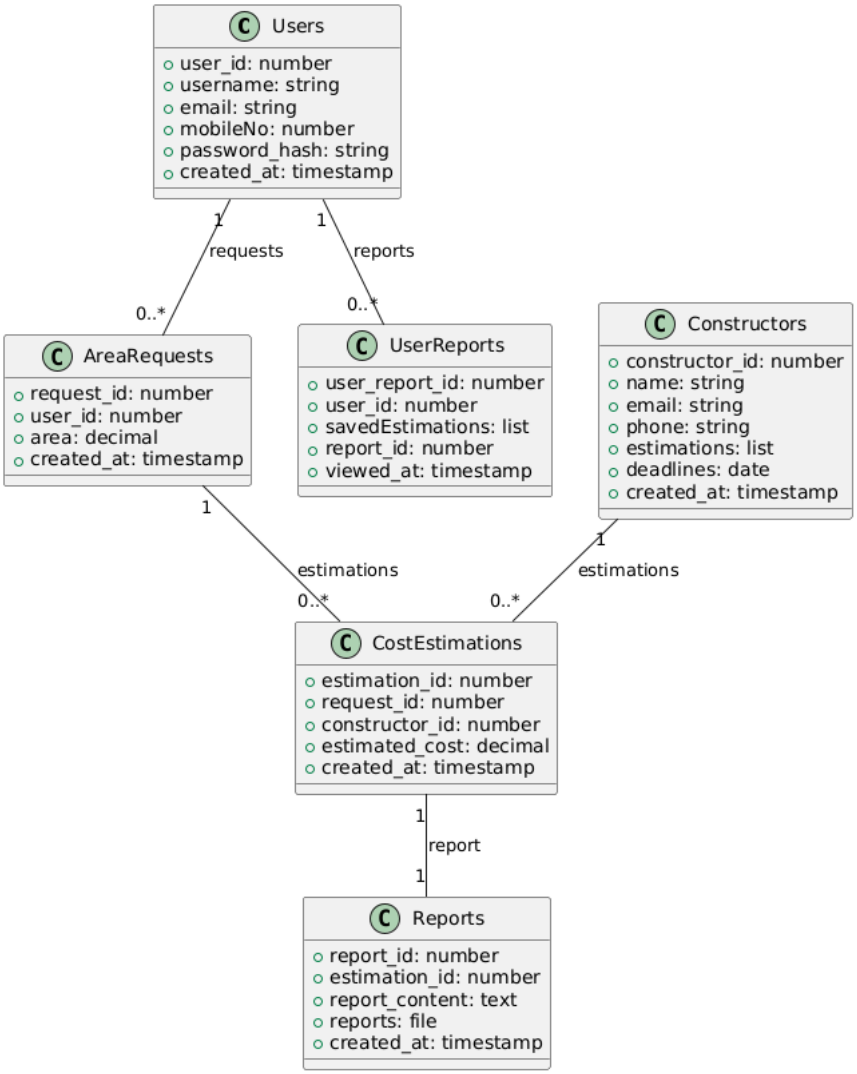
- **Cost Estimation Service**:
  - Calculates project costs based on materials and labor.
- **Project Management Service**:
  - Manages the lifecycle of projects (start, in-progress, completed).
- **Notification Service**:
  - Sends project updates, notifications, posts requirements, and alerts via messaging services.

- **Database**:
  - MongoDB (Non-relational, scalable) to store:
    - **User Information**:
      - Profiles of homeowners, contractors, and designers.
    - **Project Data**:
      - Details of ongoing and completed projects.
    - **Material and Labor Prices**:
      - Data for real-time cost estimation.
    - **Estimation History**:
      - Records of previous project estimations.

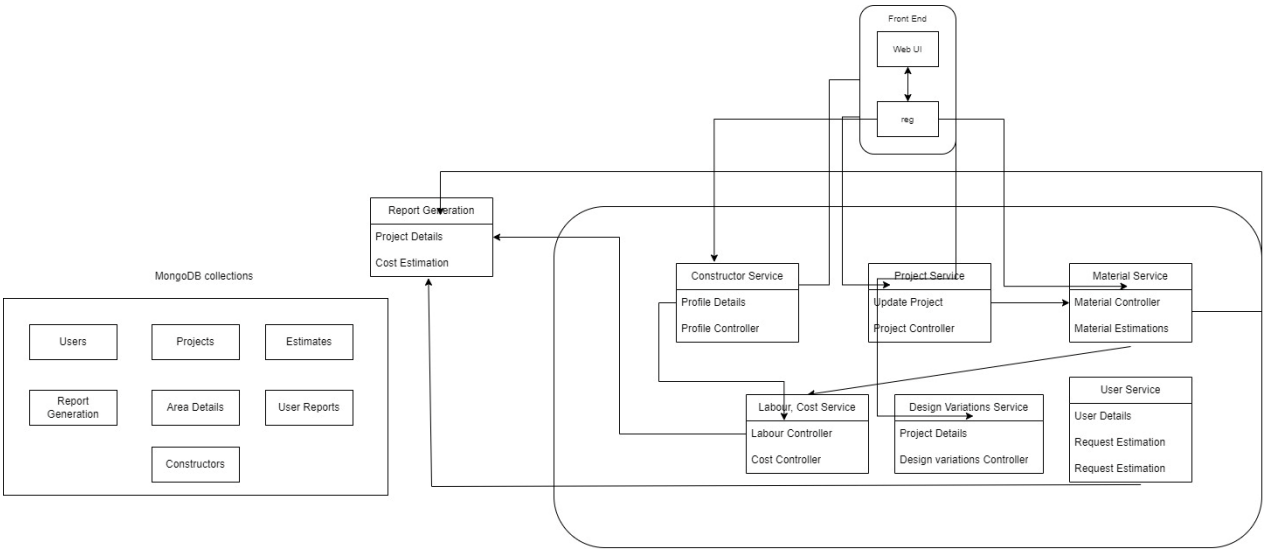
7. Schema Diagram:



UML Diagram


Users
user_id: number
username: string
email: string
mobileNo: number
password_hash: string
created_at: timestamp
1
requests
0..*
1
reports
0..*
AreaRequests
request_id: number
user_id: number
area: decimal
created_at: timestamp
UserReports
user_report_id: number
user_id: number
savedEstimations: list
report_id: number
viewed_at: timestamp
Constructors
constructor_id: number
name: string
email: string
phone: string
estimations: list
deadlines: date
created_at: timestamp
1
estimations
0..*
1
estimations
0..*
CostEstimations
estimation_id: number
request_id: number
constructor_id: number
estimated_cost: decimal
created_at: timestamp
1
report
1
Reports
report_id: number
estimation_id: number
report_content: text
reports: file
created_at: timestamp

:


Front End
Web UI
reg
Report Generation
Project Details
Cost Estimation
MongoDB collections
Users
Projects
Estimates
Report Generation
Area Details
User Reports
Constructors
Constructor Service
Profile Details
Profile Controller
Project Service
Update Project
Project Controller
Material Service
Material Controller
Material Estimations
Labour, Cost Service
Labour Controller
Cost Controller
Design Variations Service
Project Details
Design variations Controller
User Service
User Details
Request Estimation
Request Estimation

8.Authentication Endpoints

User Registration

- **Endpoint:** POST /api/users/register
  Desc: Register a new user.

User Login

- **Endpoint:** POST /api/users/login
  Desc: Authenticate user and return JWT token.

Get Current User Profile

- **Endpoint:** GET /api/users/me
  Desc: Get the current user profile (requires JWT).

Update User Profile

- **Endpoint:** PUT /api/users/me
  Desc: Update user profile (requires JWT).

User Logout

- **Endpoint:** POST /api/users/logout
  Desc: Log out the current user (optional, based on session strategy).

Construction Cost Estimation Endpoints

Add a New Project

- **Endpoint:** POST /api/projects
  Desc: Add a new construction project (requires user authentication).

Get Detailed Project Information

- **Endpoint:** GET /api/projects/:projectId
  Desc: Get detailed project information by its ID.

Update Project Details

- **Endpoint:** PUT /api/projects/:projectId
  Desc: Update project details (requires authentication).

Remove a Project

- **Endpoint:** DELETE /api/projects/:projectId
  Desc: Remove a project from the system (requires authentication).

Cost Estimation Endpoints

Estimate Project Cost

- **Endpoint:** POST /api/cost-estimates
  Desc: Estimate the cost of a project based on parameters (requires user authentication).

Get Cost Analysis Report

- **Endpoint:** GET /api/cost-estimates/report/:projectId
  Desc: Get a detailed cost analysis report for a specific project.

Analytical Reports and Visualizations

Get Analytical Dashboard

- **Endpoint:** GET /api/dashboard
  Desc: Retrieve analytical data and visualizations.

Download Cost Estimate Report

- **Endpoint:** GET /api/reports/download/:projectId
  Desc: Download the cost estimate report for a specific project.

Share Cost Estimate Report

- **Endpoint:** POST /api/reports/share
  Desc: Share the cost estimate report via email.

9.Workflow

1. **User Actions:**
   - Register/Login → Create Project → Input Project Details → View Estimate → Compare Estimates (optional) → Generate Report
2. **Backend Processing:**
   - Input Validation → Data Aggregation (External APIs) → Cost Calculation (Estimation Engine) → Store Estimate in Database → Report Generation
3. **System Notifications:**
   - Send notifications about changes in project estimates due to material or labour price updates.