**Software Development Contract**

**1. Project Overview**

**Project Name:** Modularization of {company}  
**Date:** 7/5/2025  
**Parties Involved:**

* **Frontend Developer:** {frontendDeveloper}
* **Backend Developer:** {backendDeveloper}
* **Quality Control (QC) Team:** {qaName}
  1. **Project Story**

Allow Brite commercial team and company admin to avail modules to specific users from the company's we are contracted with.

**---**

**2. Technical Analysis**

* **Technology Stack:** [List frontend, backend, and database technologies]
* **Architecture Overview:** [Brief explanation of system architecture]
* **Key Features:** [Outline core functionalities]
* **Security Considerations:** [Mention security best practices and requirements]
* **Performance Requirements:** [Specify expected performance benchmarks]

**3. Frontend Requirements**

* **Design System:** [e.g., Tailwind, Material UI]
* **Component Structure:** [Outline major UI components]
* **State Management:** [e.g., Redux, Context API]
* **API Integration:** **/companies/api/createnew/**
* **A screen shot of a computer

  AI-generated content may be incorrect.**

**employees/api/updateemployeeinfo/**

**req:**

**A screen shot of a computer screen

AI-generated content may be incorrect.**

**Response:**

**{**

**attendance: false,**

**leaves: false,**

**tasks: false,**

**payroll: true,**

**goals: true,**

**};**

**Company Profile : companies/api/getcompanyinfo/**

**Response :**

**…same Response**

**attendance: number,**

**leaves: number,**

**tasks: number,**

**payroll: number ,**

**goals: number,**

**that will be the free subscription at every module.**

* **Responsiveness & Accessibility:** [Mention any specific accessibility guidelines]

**4. API Documentation**

**4.1 API Endpoints**

|  |  |  |  |
| --- | --- | --- | --- |
| **Endpoint** | **Method** | **Request** | **Response** |
| /api/login | POST | { email, password } | { token, user } |
| /api/getData | GET | { param } | { data } |
| ... | ... | ... | ... |

**4.2 API Validation**

* **Error Handling Strategy:** [Describe error handling format]
* **Authentication & Authorization:** [Explain how tokens, OAuth, etc., are managed]

**5. Validation Layer**

* **Data Validation:** [Describe validation rules for request parameters]
* **User Input Sanitization:** [Define how user inputs will be sanitized]
* **Security Measures:** [e.g., CSRF protection, CORS settings]

**6. Logic Layer**

* **Business Rules Implementation:** [Describe the business logic applied in the backend]
* **Data Processing Flow:** [Illustrate how data is processed before being sent to the frontend]
* **Caching & Performance Optimization:** [Define caching mechanisms]

**7. UX Screens**

* **Wireframes/Mockups:** [Link to design files]
* **Navigation Flow:** [Explain how screens connect]
* **Loading & Error States:** [Define expected behaviors]
* **User Interaction Guidelines:** [e.g., hover effects, animations]

**8. QC Survey & Test Cases**

**8.1 Frontend Test Cases**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case** | **Description** | **Expected Result** | **Status** |
| Verify user view only his authorized modules |  |  | ✅ / ❌ |
| Verify users has access to his authorized module |  |  | ✅ / ❌ |

**8.2 Backend Test Cases**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Case** | **Description** | **Expected Result** | **Status** |
| Verify login response contains authorized module |  |  | ✅ / ❌ |
| Verify number of subscription limitation | User can’t exceed number of subscription |  | ✅ / ❌ |

**9. Time Estimation & Deadline**

* **Project Start Date:** [Insert Date]
* **Estimated Completion Date:** [Insert Date]
* **Milestones & Deadlines:**
  + **Frontend Completion:** [Date]
  + **Backend Completion:** [Date]
  + **QC Testing Completion:** [Date]
  + **Final Review & Deployment:** [Date]

**10. Approval & Signatures**

**Frontend Developer:** [Signature]  
**Backend Developer:** [Signature]  
**QC Team:** [Signature]

This document serves as a contractual agreement between the frontend, backend, and QC teams, ensuring a clear understanding of requirements, responsibilities, and deliverables.