Client Portal Development

1. Introduction

As accounting firms handle sensitive financial data and client interactions, a secure and user-friendly client portal is essential. A well-designed portal improves transparency, strengthens client relationships, ensures compliance, and streamlines workflows. This proposal outlines the key features, integrations, scope, and detailed timeline for building a client portal tailored for an accounting firm.

2. Objective

The goal of this project is to design and implement a centralized client portal that allows:

- Secure exchange of financial documents.
- Task and compliance tracking (like Jira workflows).
- Calendar integration for deadlines and meetings.
- Real-time communication and updates between clients and staff.

3. Scope of the Client Portal

Core Features:

- Secure login with MFA & role-based access.
- File/document sharing with version control.
- Task tracking with progress indicators.
- Dashboard for clients & staff.

Advanced Features:

- Jira-style workflow integration.
- Calendar sync with Google/Outlook.
- Automated reminders and notifications.
- Accounting-specific features: billing, e-signatures, compliance alerts.

4. Expected Benefits

- Security: Safe handling of financial data.

- Efficiency: Less back-and-forth via email.

- Transparency: Clients always see progress.

- Compliance: Alerts and audit trails to meet legal standards.

5. Detailed Project Timeline

Phase 1 – Research & Wireframing

Timeline: 8th Sept – 14th Sept (1 week)

- Activities:

  - Research accounting portal industry standards.

  - Identify key workflows for clients & staff.

  - Design wireframes for dashboard, login, task tracking.

  - Decide tech stack (React, Node.js, Database, Hosting).

- Deliverables:

  - Completed wireframes.

  - Requirement document with clear scope.

Output:

- Research completed
- Wireframes finalized
- Technology stack documented

Phase 2 – Basic Coding & Setup

Timeline: 15th Sept – 28th Sept (2 weeks)

- Activities:

  - Setup frontend (React/Next.js + Tailwind).

  - Setup backend (Node.js/Express + database schema).

  - Create basic UI screens: Login, Register, Dashboard.

- Deliverables:

  - Basic prototype with working login & dashboard placeholder.

  - Backend connected with database.

Output:

  - Frontend & backend setup complete
  - Database schema designed
  - Basic login & dashboard running

Phase 3 – Live Demo Deployment

Timeline: 29th Sept – 5th Oct (1 week)

- Activities:

  - Deploy working demo to Vercel.

  - Ensure login, dashboard, and file upload demo works.

  - Collect team feedback for improvements.

- Deliverables:

  - A live demo link accessible to team members.

  - Documentation of setup for consistency.

Output:

  - Demo portal live on Vercel
  - Team feedback collected
  - Updated codebase for improvements

Phase 4 – Major Coding & Feature Development

Timeline: 6th Oct – 16th Nov (6 weeks)

- Activities:

  - Build authentication with role-based access.

  - Develop file sharing (upload, versioning, audit trails).

  - Implement Jira-style task tracking module.

  - Integrate calendar sync with Google/Outlook.

- Add notifications (email/SMS/push).

- Build accounting-specific features (billing, e-signatures, compliance alerts).

◆ Deliverables:

- Full-featured portal with major modules functional.

- Documentation for all features.

Output:

- Authentication & security ready
- File/document management working
- Task tracking integrated
- Calendar & notification features live
- Accounting modules functional

Phase 5 – Testing & Finalization

Timeline: 17th Nov – 30th Nov (2 weeks)

◆ Activities:

- Internal testing (bugs, UI fixes, performance).

- Conduct User Acceptance Testing (UAT) with trial clients.

- Security compliance checks.

- Final deployment.

- Prepare onboarding material for clients & staff.

◆ Deliverables:

- Bug-free, tested, and secure portal ready for production.

- Training material for end-users.