# Custor-Portal-System project charter

# 1 General Project Information

|  |  |
| --- | --- |
| **Project Name** | Custor portal system |
| **Executive Sponsors** | Custor computing plc |
| **Departement Sponsor** |  |
| **Project Impact** | The portal will streamline internal team collaboration at Custor Computing by providing centralized tools for project tracking, document sharing, communication, and performance visibility. This will enhance workflow efficiency, reduce dependency on scattered tools, and improve internal transparency across departments. |

## 2 Project Team

|  |  |  |  |
| --- | --- | --- | --- |
| **Project Manager Name** | **Departement** | **Phone** | **Email** |
| Betel Yemanebirhan | SE | +251 906644386 | [yemanebetel@gmail.com](mailto:yemanebetel@gmail.com) |
| **Team members Name** | **Departement** | **Phone** | **Email** |
| Betel Yemanebirhan | SE | +251 906644386 | [yemanebetel@gmail.com](mailto:yemanebetel@gmail.com) |
| Eden Getachew | SE | +251960181601 | [edengetachew49@gmail.com](mailto:edengetachew49@gmail.com) |
| Henok | SE | +2519 |  |
| yabsra |  |  |  |

## 3 stakeholders

* Software Development Department
* HR & Project Management Team,
* Company Executives,
* Internship Program Coordinators
* Intern Mentorship Leadrs

## 4 project Scope

|  |  |
| --- | --- |
| **Project Purpose/Business Justification** | The current workflow for internal projects at Custor Computing lacks centralized tools for interns and junior staff to collaborate, manage tasks, and share progress. The portal will be an internal solution that simplifies team coordination, tracks project progress, and facilitates communication within and across teams. |
| **Objectives** | **• Provide a centralized system for project-based collaboration**  The portal will serve as a **single platform** where all employees, including interns and full-time staff, can collaboratively manage their work. the system will **consolidate communication, planning, documentation, and progress tracking** in one place. This centralization enhances **transparency**, **efficiency**, and **consistency** across departments and teams.  **• Allow interns/employees to form project teams and assign tasks**  The system will allow users to **create or join project teams** based on their roles or departments. Within each team, tasks can be created, prioritized, and assigned to team members. These tasks will be visually managed using a **Kanban-style board** (e.g., "To Do", "In Progress", "Review", "Done") to ensure everyone knows their responsibilities and task status at a glance. The goal is to foster **structured collaboration** and **accountability** in day-to-day work.  **• Include document uploads with version tracking**  To support work transparency and reduce file confusion, the portal will allow team members to **upload and manage documents** relevant to their projects (e.g., proposals, reports, code, designs). Each file upload will support **version control**, making it easy to track changes over time, revert to previous versions if needed, and ensure that teams are always working with the **latest approved documents**.  **• Enable mentors and leads to monitor project status and give feedback**  Supervisors, team leads, and mentors will have access to a **dedicated dashboard** where they can monitor the real-time progress of all ongoing projects. They’ll be able to view task boards, review uploaded files, leave feedback on specific deliverables, and even **approve or reject** project milestones or documents. This feature ensures timely guidance, faster issue resolution, and a **stronger connection between leadership and project teams**. |
| **Deliverables** | **• Web-based portal with login system (JWT + role-based access)**  This is the foundation of the platform. The portal will be built as a **responsive web application** accessible from any device through a browser. Users must log in with their **username and password**, and upon successful authentication, a **JWT (JSON Web Token)** is issued to authorize access securely.  The system will also support **role-based access control (RBAC)**:  **Admin** – full access to users, teams, and configuration  **Mentor/Supervisor** – can view team progress, give feedback  **Employee/Intern** – can create or join teams, manage tasks and files  Each role will have restricted access to only relevant pages and data.  **• Project & task management modules (Kanban-style)**  This module provides teams with a **visual way to organize and track their work**. Each project will have a Kanban board where tasks are represented as movable cards that flow through columns:  **To Do**  **In Progress**  **In Review**  **Done**  Features include:  Task creation with descriptions, deadlines, priorities  Assigning tasks to team members  Commenting on tasks  Drag-and-drop updates to task status  Filtering by assignee or status  This increases **visibility**, promotes **accountability**, and helps with **project tracking**.  **• File upload and document tracking**  Users will be able to **upload project-related files** such as:  Reports  Design documents  Code archives  Presentations  Files will be **tagged by project and file type**, and a **versioning system** will allow:  Storing historical versions  Viewing previous uploads  Preventing accidental overwrites  This ensures that teams always have access to the **latest approved documents** and enables **traceability**.  **• Communication feature (comment threads or chat)**  A lightweight communication system will allow:  **Threaded comments** on tasks and documents  Mentions (e.g., @username) to notify specific people  Notifications for updates or changes  This eliminates the need for external tools like WhatsApp or email for daily collaboration, ensuring that **all discussions stay contextually linked to the work being done**.  **• Admin/mentor dashboard**  A centralized view for users with elevated permissions:  **Mentors**: can view all assigned teams, see project dashboards, review file submissions, and provide approval or feedback  **Admins**: can manage system settings, assign mentors to teams, manage user roles and access, and view system activity logs  This ensures better **oversight**, **support**, and **decision-making** without micromanagement.  **• Progress overview and reporting**  A **progress tracking system** will be included to:  Show task completion percentages per project  Highlight upcoming or overdue deadlines  Display a **timeline/milestone view** of project phases  Generate **summary reports** on team activity and submissions  This helps mentors and leads quickly understand how each project is progressing and where attention may be needed. |
| **Within Scope** | • Angular frontend and ASP.NET Core backend • Role-based authentication (Intern, Mentor, Admin) |
| **Outside of Scope** | • Full-feature chat (use comment threads instead) • Mobile app version • Third-party integrations (email/calendar) |
| **Project Milestones** | **Form Project Team / Preliminary Review / Scope**  **Day 1–3**   * Assign roles: Betel (Project Charter), Eden (BRD), Henock (SRS) * Outline high-level scope and tool setup * Initial research and understanding of business goals * Share approved Charter sections (Objectives, Stakeholders) with team * prepares preliminary architecture ideas based on tech stack(Henok)   **Finalize Project Plan / Charter / Kick Off**  **Day 3–4**   * Project Charter finalized and signed off (Betel) * BRD (Eden) begins using Charter inputs * SRS Architecture assumptions (Henock) initialized * Official team kickoff with mentor   **Define Phase**  **Day 5–11**   * **Eden starts BRD Functional Requirements (Day 5)** * Henock works on **Security & API plan (Day 7)** using BRD’s business goals * **Betel** sets up backend structure, database design, API endpoints (Day 5–10) * **Eden & Henock** collaborate on Angular layout, modules, and routing setup * **Henock finalizes SRS Data Model (Day 11)** after BRD requirements are completed   **Day 12–16**   * implement and integrate:   Kanban-style Task Module  File Upload + Versioning  Mentor/Admin Dashboard   * Backend testing: endpoint security, database queries * UI testing: task transitions, file preview, dashboard data * Progress indicators for each project/team displayed   **Analysis Phase**  **Day 17–20**   * Internal demo to mentor Collect feedback on architecture, UX, and security * Review all system components for performance, gaps * Refactor or redesign areas based on feedback   **Improvement Phase**  **Day 21–28**   * Implement all feedback * Complete role-based access logic * Refine UI styling, responsiveness, and UX flows * Finalize BRD, SRS, and Project Charter docs * Begin writing user documentation and API specs   **Control Phase**  **Day 29–34**   * Final deployment to staging or live server * System monitoring and manual testing * Mentor signs off features * Prepare final user manual, training/demo slides * Lock project codebase   **Project Summary Report and Close Out**  **Day 35–38**   * Final project presentation to company/mentor |
| **Risks** | • Unfamiliarity with full-stack development could slow progress • Conflicting task distribution or scope creep |
| **Constraints** | • Must be completed within internship timeline (~3–4 weeks) • Team consists of interns with learning curve • Use technologies approved by company mentor |
| **Assumptions** | • Mentor will be available for regular feedback • we must have access to company systems/resources |
| **External Dependencies** | • Internet access for dev tools ( GitHub) |

## 5 Communication Strategy

Method and frequency

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Frequency | Purpose | Owner/participants |
| In person Team Meeting | 3 days per week | Discuss progress,assign tasks,resolve blockers,and plan next steps | Betel,Eden,Henok |
| Mentor Sync-up | Once a week | Share updates, get feedback, validate technical and functional direction | Yabssra |
| Telegram Group chat | As needed(daily) | Quick coordination, reminders, updates, and informal discussion | All team members |
| GitHub Repository | Ongoing | Version control, code collaboration, pull request reviews, issue tracking | Betel, Eden, Henock |
| Document Sharing | Weekly updates | Store and share Project Charter, BRD, SRS, API documentation, etc. | all contributors |
| Internal Demo Sessions | Day 17 & Day 33 | Present project progress to mentor and gather formal feedback | Team and Yabssra |

## 6 Notes

**PROCESS OWNER**

**KEY STAKEHOLDERS**

Software Development Department

**FINAL CUSTOMER**

Custor Computing – Software Development Department

**EXPECTED BENEFITS**

Improved team collaboration, clear task tracking and progress visibility, enhanced intern engagement and accountability

## 7 sign-off

|  |  |  |  |
| --- | --- | --- | --- |
| Role | Name | Signature | Date |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |