Stakeholders

1. **Recognized**
   - Identified key stakeholders: **students, instructors, and administrators**.
   - Defined responsibilities: **students use the platform for coding, instructors manage courses and assignments, and admins oversee operations**.
2. **Represented**
   - Established responsibilities: **instructors create coding problems, students solve them, and admins ensure system functionality**.
   - Authorized **representatives**: instructors guide learning, while admins manage user roles.
   - Ensured **smooth collaboration** between users and system components.
3. **Involved**
   - Provided **timely updates** for assignments, test cases, and submissions.
   - Communicate **assignment updates efficiently** through the notifications.
4. **In Agreement**
   - **Defined minimum expectations**: Course access, assignment completion.
5. **Satisfied for Deployment**
6. **Satisfied in Use**

Opportunity

1. **Identified**
   - Recognized an **opportunity for an online coding lab** tailored for multiple programming languages.
   - Identified **investing stakeholders**: Students looking to practice coding, instructors needing an assignment system, and institutions seeking a structured platform.
2. **Solution Needed**
   - Defined the **core problem**: A need for an interactive, assignment-based coding platform.
   - Identified **root causes**: Lack of structured coding assignments.
   - Confirmed the **necessity of a solution** and proposed a feature-rich **coding dashboard**.
3. **Value Established**
   - The **value of our online lab platform** is quantified:
     - i. Helps students **practice coding** with structured courses.
     - ii. Provides instructors with an **efficient assignment management system**.
   - Clear **success criteria**:
     - i. Increased student engagement.
     - ii. Timely assignment submissions.
     - iii. Improved coding proficiency.
4. **Viable**

5. **Addressed**
6. **Benefit Accrued**

Software System

1. **Architecture Selected**
    - **Technology stack decided**: Frontend (React), Backend (Express.js , Node.js), Database (MongoDB).
    - **System boundaries defined**: Coding environment, assignment tracking.
    - **Decisions made**:
        - Secure authentication for users.
2. **Demonstrable**
    - Key **architectural features**:
        - Course selection and assignment submission work.
    - Will be Integrated **with necessary third-party tools** for code execution.
3. **Usable**
    - **System functionality tested**: Assignment tracking and coding execution.
    - **Bug fixing and performance optimization** will be looked upon.
    - Documentation for instructors and students.
4. **Ready**
5. **Operational**
6. **Retired**

Work

1. **Initiated**
    - Project goals defined (online lab platform for course management and coding practice).
    - Constraints (time, resources, technical feasibility) were identified.
    - Stakeholders (students, instructors) recognized the need for the system.
    - Clear prioritization of features like coding execution and assignment tracking.
2. **Prepared**
    - Risk factors (security issues) understood.
    - Development tasks broken down into coding dashboard, assignment management, and user authentication.
3. **Started**
    - Development of core features began (Frontend , Backend , Database created).
    - Regular progress monitoring to keep the project on track.
4. **Under Control**
    - Tasks being completed in phases (Frontend + Backend +Database).
    - Progress measured and performance optimized for smooth usage.
5. **Concluded**
6. **Closed**

Way of Working

1. **Principles Established**
    - Team agrees on working principles and methodologies (Agile, Scrum).
    - Tools needed are identified (GitHub, Trello, IDEs, databases).
    - The operational context is understood, ensuring clarity on constraints and challenges.
2. **Foundation Established**
    - Key practices and tools are selected based on project requirements.
    - Gaps in existing vs. required workflows are analyzed.
    - An integrated way of working is available for the team to follow.
3. **In Use**
    - The chosen tools and methodologies are actively used.
    - Regular inspections and adjustments ensure efficiency.
    - Feedback loops are in place to refine the process.
4. **In Place**
    - Entire team adopts and follows the established way of working.
    - The system is accessible to all team members.
    - Continuous improvement based on team feedback.
5. **Working Well**
    - Consistent progress is being made following the chosen approach.
    - Tools and practices align naturally with workflow.

Team

1. **Seeded**
    - Mission of the team is defined (developing an online lab platform).
    - Constraints, governance, and leadership structure are determined.
    - Team responsibilities and skill requirements are outlined.
2. **Formed**
    - **Team members are recruited**, and roles are understood.
    - **Communication mechanisms are set up** for smooth coordination.
3. **Collaborating**
    - Works as a unit, fostering strong teamwork.
    - Open and honest communication among members.
    - Members are familiar with each other, reducing friction in coordination.
4. **Performing**
    - Consistently meeting commitments and deadlines.
    - Adapting effectively to project changes.
    - Problem-solving mindset, addressing challenges proactively.

Requirements

1. **Conceived**
    - Stakeholders agree that the system should be developed.
    - Users and funding sources are identified.

    - Clear business or project opportunity defined.

2. **Bounded**
    - Development stakeholders are identified.
    - Purpose, success criteria, and constraints are clear.
    - Requirements management framework is in place.
    - Assumptions are documented and considered.
3. **Coherent**
    - Requirements are shared and their origins understood.
    - Conflicts are resolved, and priorities are set.
    - Team is aligned on what to deliver.