

Level 1	Level 2	Level 3	Level 4
Internseek (Internship Application System)	1.0 INITIATION	1.1 Define Project Objectives	
		1.2 Identify Project Scope and Boundaries	1.2.1 Determine core functionalities
			1.2.2 Establish project constraints (time, cost, resources)
			1.2.3 Define out-of-scope functionalities
			1.2.4 Define project deliverables
		1.3 Conduct Feasibility Study	1.3.1 Analyze and assess the technical feasibility by evaluating the available tools and platforms
			1.3.2 Assess resource feasibility, including budget and time
			1.3.3 Assess market feasibility by analyzing the target market demand and competition
			1.4.1 Identify people involved in the system, directly affected individual, indirectly affected individual
		1.4 Identify Key Stakeholders	1.4.2 Define the roles and responsibilities for every stakeholders
		1.5 Perform Requirement Gathering	1.5.1 Collect requirements from students and employers
			1.5.2 Define initial success criteria
		1.6 Plan User Engagement and Feedback Methods	1.6.1 Define key milestones for user feedback
			1.6.2 Determine the method for collecting feedback at the relevant milestones
	2.0 PLANNING	2.1 Create Project Schedule and Timeline	2.1.1 Break down major milestones and deadlines
			2.1.2 Establish task dependencies and key deliverables
		2.2 Allocate Resources and Budget	2.2.1 Define human resources and assign roles
			2.2.2 Set and approve budget
		2.3 Conduct Risk Assessment	2.3.1 Identify potential project risks
			2.3.2 Develop mitigation plans for high-priority risks
		2.4 Arrange Priority Features and Requirements	2.4.1 Rank features based on importance and feasibility
			2.4.2 Align features with project objectives and stakeholder needs
			2.4.3 Set clear acceptance criteria for each feature
		2.5 Plan Development Tools and Frameworks	2.5.1 Finalise backend and frontend frameworks
			2.5.2 Choose necessary databases and tools
		2.6 Design System Architecture, Relevant Diagrams	2.6.1 Create Entity-Relationship diagrams for database structure
			Diagram
			2.6.3 Create Use Case Description, Data Dictionary
		2.7 Create prototype	2.6.4 Design the high-level architecture of system
			2.7.1 Design user interfaces using Figma
			2.7.2 Link the relationship between each interface
	3.0 EXECUTION	3.1 Set Up Development Environment and Version Control	3.1.1 Establish development environment and workspace configuration
			3.1.2 Set up version control and repositories
		3.2 Install Required Dependencies	3.2.1 Configure environmental variables
			3.2.2 Set up modules used in development
		3.3 Implement Database	3.3.1 Design and create database schemas
			3.3.2 Connect relational database connections
		3.4 Develop User Authentication and Access Control	3.4.1 Implement login and registration functionality
			3.4.2 Add session management and access level permissions
			3.4.3 Implement password reset and account recovery
		3.5 Design Frontend Layout and Navigation	3.5.1 Develop UI Components Based on Prototype Design
			3.5.2 Develop the User Interface Using Previously Developed Components
			3.5.3 Define Navigation Flow and Structure, and Implement Navigation Logic
		3.6 Develop Backend Functionalities and Key Features	3.5.4 Design the Interface to Be Responsive
			3.6.1 Set Up Database Models and Relationships
			3.6.2 Implement the CRUD functionality in the controller
			3.6.3 Define the corresponding routes for each CRUD operation
			3.6.4 Define the task schedule to automatically update the internship posting status based on the posting's start and end date
			3.6.5 Implement a messaging feature to enable communication between students and employers.
	4.0 EXECUTION & MONITORING	4.1 Conduct Unit Testing for Individual Components	4.1.1 Write and run test cases for core functionalities
		4.2 Perform Integration Testing for Feature Interactions	4.1.2 Document test results and fix identified issues
			4.2.1 Test interdependencies among modules
		4.3 Conduct System Testing on Whole Application	4.2.2 Record and troubleshoot any integration issues
			4.3.1 Run complete system tests to ensure full functionality
		4.4 Conduct User Acceptance Testing (UAT)	4.3.2 Conduct end-to-end testing across major user scenarios
			4.4.1 Create Google Form for UAT Testing
		4.5 Execute Load and Performance Testing	4.4.2 Execute UAT with End-Users and Collect Detailed Feedback
			4.4.3 Make improvements based on collected feedback
	5.0 CLOSING	5.1 Conduct Usability Testing	4.5.1 Test application response time under high load
			4.5.2 Optimize code or queries to improve performance
			5.1.1 Conduct usability testing sessions with users
		5.2 Deploy Final Version to Production	5.1.2 Collect qualitative and quantitative feedback from users
			5.1.3 Implement improvements based on usability feedback
		5.3 Prepare Handover Documentation	5.2.1 Migrate system to live server and set up domain
			5.2.2 Verify deployment setup and accessibility
		5.4 Create User Manual and Support Documents	5.3.1 Compile project documentation
			5.3.2 Archive code, assets, and resources in repository
		5.5 Gather Post-Implementation Feedback	
			5.5.1 Monitor user experience and address issues
		5.6 Perform Final Review and Close-Out Report	5.5.2 Document user suggestions for future maintenance
			5.6.1 Review project goals, outcomes, and any discrepancies
		5.7 Collect Final Sign-Off from Stakeholders	5.6.2 Generate project close-out report with lessons learned
		5.8 Formally Close Project	