

# **Application Development Group Project Deliverables (With Submission Deadlines)**

## **Overview**

This document outlines the detailed list of deliverables for students working in a group project on a software application development project. The students (in consultation with course instructor and TAs) are expected to decide the application type, determine its scope, collaboratively develop it, and deploy it on a public platform (preferably free, otherwise paid). **It is assumed that project groups have been formed and project ideas for respective groups have been approved.**

### **1. Project Proposal Document (Due: 26-Aug-2025)**

- Project title and short description
- Application type (web (preferred for admin side), mobile (preferred for client side), hybrid, etc.)
- Motivation/problem statement
- Target users and use cases
- Scope and high-level features
- Technology stack
- Roles and responsibilities
- Timeline with milestones

**Deliverable:** PDF document (3–4 pages)(in Git repo and Google classroom)

### **2. Requirements Specification Document (Due: 07-Sep-2025)**

- Functional and non-functional requirements
- Detailed User stories / use case diagrams
- Interface requirements
- Wireframes or UI/UX mockups (Initial Draft)
- Acceptance criteria

**Deliverable:** Business Reqs. Doc. (BRD) (in Git repo and Google classroom).

### **3. System Design Document (Due: 14-Sep-2025)**

- Architecture diagram
- Database schema (ER diagram)
- Class/component diagrams (Add other diagrams for a more detailed design)
- REST API contracts (Swagger/OpenAPI)
- Wireframes or UI/UX mockups (Majorly Final)

**Deliverable:** Design Doc. + diagrams + mockups (in Git repo and Google classroom)

### **4. Project Plan / Gantt Chart (Due: 14-Sep-2025)**

- Task timeline and milestones
- Team task assignments
- Tools: Trello, Jira, GanttProject

**Deliverable:** Timeline chart + task board screenshot (in Git repo and Google classroom)

### **5. Weekly Progress Reports (Weeks 15-Sep to 25-Nov-2025)**

- Summary of tasks
- Roadblocks
- Upcoming tasks

**Deliverable:** Weekly 1-page PDF or shared doc (in Git repo and Google classroom)

### **6. Mid-Semester Demonstration (Due: 14-Oct-2025)**

- Partial working demo
- Code walkthrough
- Team Q&A

**Deliverable:** Live demo + slides (in Git repo and Google classroom)

### **7. Test Plan and Test Cases (Due: 04-Nov-2025)**

- Unit and integration tests
- Populating DB with meaningful records.
- UI testing (if applicable)
- Bug log (if applicable)

**Deliverable:** Test documents + results (in Git repo and Google classroom)

### **8. Final Application (Due: 15-Nov-2025)**

- Complete feature set (fully functional app)
- Frontend + backend + database integration
- API integration (on need basis)

**Deliverable:** Source code + README (in Git repo and Google classroom (zip version))

## **9. Deployment & Hosting (Due: 15-Nov-2025)**

- Hosting on Heroku, Firebase, Vercel, etc.
- Public URL with working app
- Optional CI/CD and basic security configs

**Deliverable:** Live URL + deployment notes

## **10. Final Project Report (Due: 18-Nov-2025)**

- Summary, system overview, tech stack
- Challenges and solutions
- Screenshots, testing, deployment notes

**Deliverable:** 15–25 page PDF report (in Git repo and Google classroom)

## **11. Final Presentation and Demo (Due: 18-Nov-2025)**

- Slide deck (10–15 slides)
- Project Poster
- Application demo
- Lessons learned + future scope

**Deliverable:** Slides (in Git repo and Google classroom) + live presentation

## **12. Peer Review and Contribution Report (Due: 19-Nov-2025)**

- Individual peer reviews
- Self-assessment
- Instructor/TA-only access

**Deliverable:** Peer review forms (in Google classroom)

## **Other Important Deliverables**

- Project poster (36" x 48") (softcopy and print version) (in Git repo and Google classroom)
- Project video walkthrough
- User manual/help guide (in Git repo and Google classroom)
- API documentation (Swagger/Postman) (in Git repo and Google classroom)
- Dockerized deployment / Helm charts (Preferred)

## Suggested Tools and Platforms

- **Version Control:** GitHub, GitLab, or similar.
- **UI Design:** Figma, Adobe XD, or similar.
- **Project Management:** Trello, Jira, or similar.
- **Frontend:** React, Angular, Vue, or similar.
- **Backend:** Node.js, Spring Boot, Django, or similar.
- **Database:** PostgreSQL, MongoDB, MySQL, or similar.
- **Deployment:** Heroku, Vercel, Firebase, AWS Free Tier, or similar.
- **CI/CD:** GitHub Actions, GitLab CI or similar.