جامعة الإمام عبدالرحمن بن فيصل
IMAM ABDULRAHMAN BIN FAISAL UNIVERSITY
كلية علوم الحاسب وتقنية المعلومات
College of Computer Science and Information Technology

CSC 305: Software Engineering
Term: 1
Academic Year: 2025/2026
Course Coordinator: Dr. Rahma Ahmed

Software Project Management Plan (SPMP)

Based on IEEE 1058 - Adapted for Student Projects

Project Title: Integrated Supply Chain Management System with QR-Based Employee Attendance Tracking

Team Name & Members:

- Jawad Ali Alnatah (2240002923) - Team Leader & Backend Developer
- Mustafa AbdulKarim AbdRabAlameer (2240002959) - Backend Developer & Database Designer
- Abdullah Jaffer Masiri (2240004545) - UI/UX Designer
- Ahmed Hussain Alghazwe (2240002359) - Frontend Developer
- Abdullah Abdulaziz Alhamadi (2240003012) - Frontend Developer & GUI Designer
- Mohammad Khalid Alqallaf (2240005145) - Quality Assurance & Documentation

Advisor: [Saeed Matar Alshahrani]

Version: [1.0]

Date: [07-11-2025]

Table of Contents

1. Project Overview

1.1 Purpose, Scope, and Objectives

Purpose

Create a JavaFX desktop application that combines supply chain management with QR-code employee attendance tracking using MySQL database. This application helps businesses manage their inventory and track employee attendance in one system, making work easier and faster while reducing mistakes and saving time.

Scope

Included:

- Inventory management with stock tracking and alerts
- Supplier and purchase order management
- Employee management with role-based access
- QR code generation and web-based scanning (HTML5)
- Attendance tracking with timestamps and validation
- Reporting dashboards for supply and attendance metrics
- Admin panel for system configuration

Excluded:

- Native mobile apps (using web for QR scanning)
- Biometric authentication
- External ERP integration
- AI/ML analytics
- Payroll processing
- IoT sensor integration
- Multi-currency support

Objectives

1. Build functional supply chain management system
2. Implement secure QR-based attendance via smartphone browsers
3. Integrate supply and attendance modules
4. Design intuitive JavaFX desktop interface and responsive web QR scanner
5. Create role-based access control (employees, managers, admins)
6. Develop comprehensive reporting and analytics
7. Apply software engineering best practices
8. Deliver production-ready, scalable system

1.2 Assumptions, Constraints, and Risks

Assumptions

- All team members have laptops with 8GB+ RAM
- Reliable internet access available
- MySQL can use free-tier cloud services
- Smartphones support HTML5 Camera API
- Team willing to learn JavaFX

Constraints

- **Time:** 16-week semester, fixed deliverable deadlines
- **Resources:** 6 team members, no budget, free tools only
- **Technology:** Java 11+, JavaFX, MySQL 8.0, HTML/CSS/JS

Top Risks

- Limited JavaFX and QR code integration experience may slow development
- Database design errors could cause data integrity and performance issues
- Key team member unavailability could severely impact progress
- Unclear or changing requirements leading to rework

Scope: Desktop app only, no native mobile apps

1.3 Project Deliverables

Deliverable	Due Date	Format
Project Idea Form	Week 3	PDF
Project Proposal	Week 4	PDF
SPMP	Week 8	PDF/DOCX
Status Report 1	Week 9	PDF
SRS	Week 11	PDF/DOCX
SDD	Week 13	PDF/DOCX
Status Report 2	Week 13	PDF
STS	Week 15	PDF/DOCX
Final Code + Documentation	Week 16	ZIP + GitHub
Presentation & Demo	Week 16/17	Live Demo

GitHub Repository: https://github.com/JawadAlnatah/Integrated-Supply-Chain-Management-System-with-QR-Based-Employee-Attendance-Tracking-.git

1.4 Schedule Summary

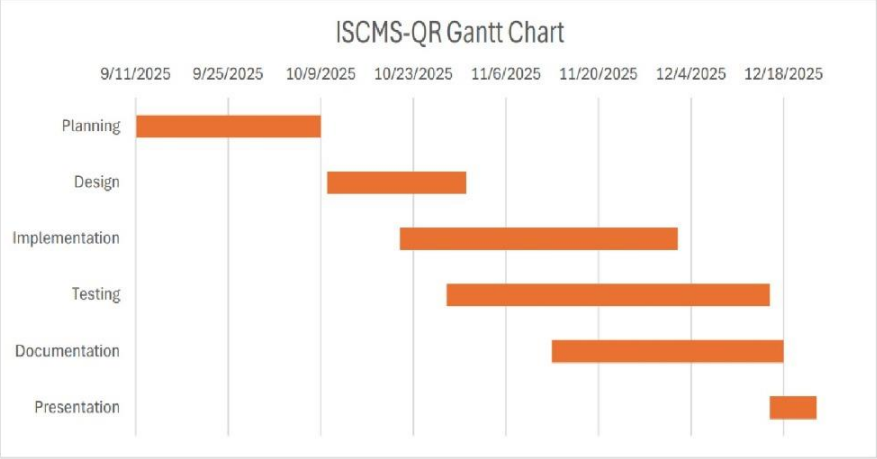
Schedule Summary

- **Planning:** Weeks 1-4
- **Design:** Weeks 5-8
- **Implementation:** Weeks 7-13
- **Testing:** Weeks 8-15
- **Documentation:** Weeks 11-16
- **Presentation:** Week 16/17

Budget Summary

Total Budget: $0 - $60

- Development tools: $0 (open source)
- Cloud database hosting: $0-60 (free tier or DigitalOcean)
- Web hosting: $0 (GitHub Pages/Netlify)



1.5 References

Scheduled Updates:

- Week 9 (after Status Report 1)
- Week 11 (after SRS)
- Week 13 (after SDD)
- Week 15 (after STS)

Version Control:

- Stored in GitHub /documentation/management/
- Version increments for updates
- Change history maintained

1. IEEE Std 1058-1998 - Software Project Management Plans
2. IEEE Std 730-1998 - Software Quality Assurance Plans
3. IEEE Std 830-1998 - Software Requirements Specifications
4. CSC 305 Course Syllabus
5. Project Idea Form - Week 3
6. Project Proposal Form - Week 4
7. JavaFX Documentation: https://openjfx.io/
8. ZXing Library: https://github.com/zxing/zxing
9. MySQL 8.0 Documentation
10. Sommerville, I. Software Engineering (10th Edition)

1.6 Definitions and Acronyms

API: Application Programming Interface

CRUD: Create, Read, Update, Delete

DAO: Data Access Object

GUI: Graphical User Interface

JAR: Java Archive

JDBC: Java Database Connectivity

MVC: Model-View-Controller

QR: Quick Response Code

RBAC: Role-Based Access Control

SDD: Software Design Description

SPMP: Software Project Management Plan

SQL: Structured Query Language

SRS: Software Requirements Specification

STS: Software Test Specification

UI/UX: User Interface/User Experience

WBS: Work Breakdown Structure

Sprint: Time-boxed iteration (2 weeks) in Agile development

Velocity: Story points completed per sprint

Baseline: Approved version under change control

Milestone: Significant project event or completion point

2. Project Organization

2.1 External Interfaces

Course Instructor: Dr. Saeed Matar Alshahrani - Project advisor, deliverable approval, final evaluation

GitHub: Version control and collaboration platform

Cloud Providers: AWS/Azure/DigitalOcean for database hosting

2.2 Internal Structure

Team Leader: Jawad Ali Alnatah

- Overall coordination, backend development, stakeholder communication

Backend Team:

- Mustafa AbdulKarim (Database & Backend)
- Jawad Ali Alnatah (Backend & Integration)

Frontend Team:

- Ahmed Hussain Alghazwe (Desktop UI - JavaFX)

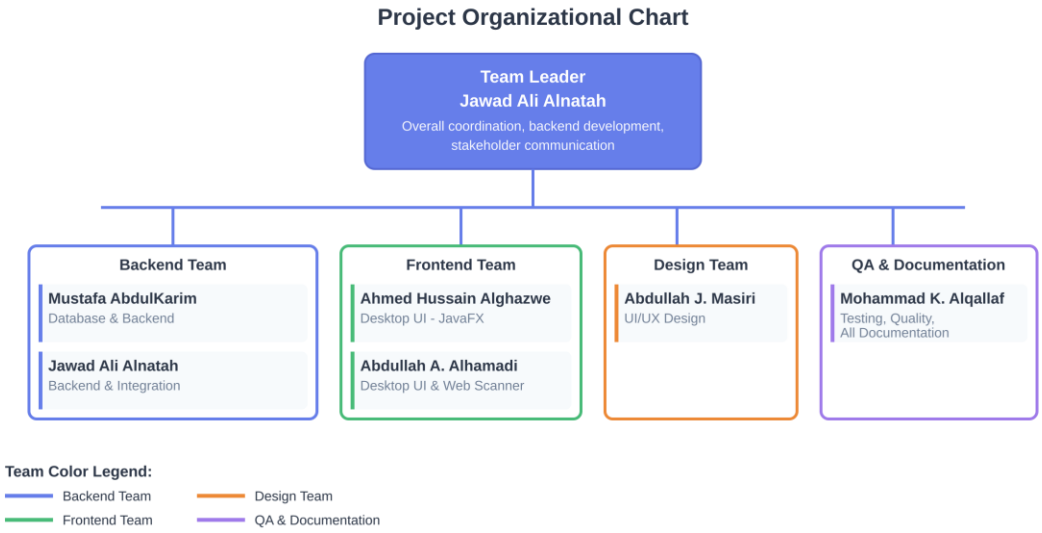
- Abdullah Abdulaziz Alhamadi (Desktop UI & Web Scanner)

Design Team:

- Abdullah Jaffer Masiri (UI/UX Design)

QA & Documentation:

- Mohammad Khalid Alqallaf (Testing, Quality, All Documentation)



2.3 Roles and Responsibilities

Role	Primary Responsibilities	Deliverables
Jawad	Backend development, project management, integration	SPMP, backend modules, integration
Mustafa	Database design/implementation, QR generation, deployment	Database schema, ER diagrams, SQL scripts
Abdullah J.	UI/UX design, wireframes, style guide	Mockups, design artifacts, SDD sections
Ahmed	Desktop UI implementation (inventory, suppliers, PO, admin)	JavaFX screens, controllers

Abdullah A.	Desktop UI (dashboards, reports), web QR scanner	Dashboard UI, web scanner page
Mohammad	Testing, QA, all documentation compilation	STS, User Manual, test reports, all docs

3. Managerial Process Plans

3.1 Estimates & Staffing

Effort Estimation:

- Total: ~2,100 person-hours (6 members × 16 weeks × ~22 hrs/week)
- Method: Analogy-based estimation with PERT for complex tasks
- Based on similar academic projects, adjusted for team size and complexity

Staffing by Phase:

Phase	Duration	Backend Dev	Frontend Dev	UI/UX Design	QA & Docs
Planning	Weeks 1-4	2 members	1 member	1 member	1 member
Design	Weeks 5-8	2 members	1 member	1 member	1 member
Implementation	Weeks 7-13	2 members	2 members	1 member	1 member
Testing	Weeks 14-15	2 members	2 members	1 member	1 member
Documentation	Weeks 11-16	1 member	1 member	1 member	1 member

Team Member Assignment:

Role	Team Members	Primary Focus
Backend Development	Jawad, Mustafa	Database, server logic, QR code
Frontend Development	Ahmed, Abdullah A.	JavaFX UI, web QR scanner
UI/UX Design	Abdullah J.	Wireframes, mockups, style guide
QA & Documentation	Mohammad	Testing, all documentation

3.2 Work Plan

3.2.1 Work Breakdown Structure (Summary)

Project Management - Planning, monitoring, communication, closeout

Requirements - Gathering, analysis, SRS documentation

Design - Architecture, database, UI/UX, SDD documentation

Implementation - Environment setup, database, backend, frontend, web scanner, integration

Testing - Test planning, unit/integration/system testing, STS, bug fixing

Deployment - Cloud setup, application packaging, deployment

Documentation - Technical docs, user manual, meeting minutes

QA & CM - Reviews, audits, version control

IAU
جامعة الإمام عبدالرحمن بن فيصل
IMAM ABDULRAHMAN BIN FAISAL UNIVERSITY
كلية علوم الحاسب وتقنية المعلومات
College of Computer Science and Information Technology

CSC 305: Software Engineering

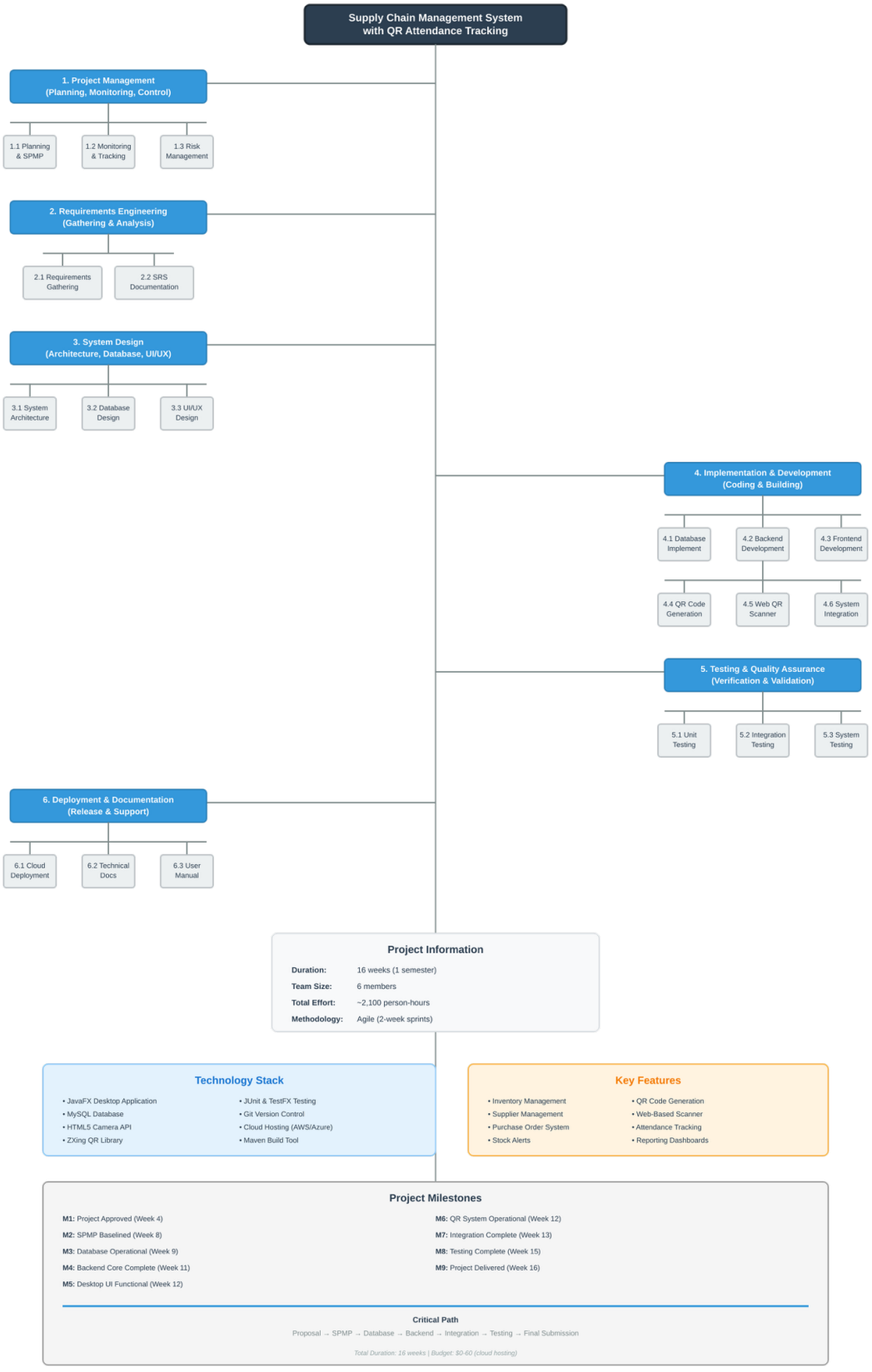
Term: 1

Academic Year: 2025/2026

Course Coordinator: Dr. Rahma Ahmed

Work Breakdown Structure (WBS)

Integrated Supply Chain Management System with QR-Based Attendance Tracking



3.2.2 Schedule

Milestone	Target Date	Success Criteria
M1: Project Approved	Week 4	Proposal approved
M2: SPMP Baselined	Week 8	SPMP submitted
M3: Database Operational	Week 9	Schema implemented & tested
M4: Backend Core Complete	Week 11	All backend modules functional
M5: Desktop UI Functional	Week 12	Major UI screens implemented
M6: QR System Operational	Week 12	QR generation & scanning working
M7: Integration Complete	Week 13	All modules integrated
M8: Testing Complete	Week 15	All tests passed, bugs fixed
M9: Project Delivered	Week 16	All deliverables submitted

Critical Path: Proposal → SPMP → Database → Backend → Integration → Testing → Final Submission

3.2.3 Resource Allocation (Person-Hours by Phase)

Phase	Jawad	Mustafa	Abdullah J.	Ahmed	Abdullah A.	Mohammad	Total
Planning	60	50	60	50	50	90	360
Design	70	80	90	60	60	50	410
Implementation	140	140	40	130	130	60	640
Testing	40	40	20	40	40	180	360
Documentation	50	40	60	30	30	180	390

Deliverables: Project Idea Form, Project Proposal, Initial SPMP draft

Acceptance Criteria:

- Proposal approved by instructor
- All team members understand roles
- Clear scope agreement reached

Predecessors: None

Successors: All subsequent activities

PM-1: Project Planning

- **Resources:** Team Leader (Jawad), All team members
- **Estimated Duration:** Weeks 1-4 (80 hours total)
- **Deliverables:** Project Idea Form, Project Proposal, Initial SPMP draft
- **Acceptance Criteria:**
  - Proposal approved by instructor
  - All team members understand roles
  - Clear scope agreement reached
- **Predecessors:** None
- **Successors:** All subsequent activities

PM-2: Project Monitoring & Control

- **Resources:** Team Leader (Jawad), QA Lead (Mohammad)
- **Estimated Duration:** Weeks 1-16 (continuous, 120 hours total)
- **Deliverables:** Weekly progress reports, Status Reports (Week 9, 13), meeting minutes
- **Acceptance Criteria:**
  - All milestones tracked
  - Issues identified and resolved within 48 hours
  - Status reports submitted on time
- **Predecessors:** PM-1
- **Successors:** PM-3

PM-3: Project Closeout

- **Resources:** All team members
- **Estimated Duration:** Week 16 (40 hours total)
- **Deliverables:** Final presentation, retrospective document, lessons learned
- **Acceptance Criteria:**
  - All deliverables submitted
  - Presentation delivered successfully
  - Archive complete on GitHub
- **Predecessors:** All implementation and testing activities
- **Successors:** None

3.2.4 Budget Allocation

Category	Cost	Notes
Personnel	$0	Academic project
Software/Tools	$0	All open source
Cloud Database	$0-60	Free tier preferred, DigitalOcean backup
Web Hosting	$0	GitHub Pages/Netlify
Total	**$0-60**	

3.3 Project Tracking Plan

3.3.1 Requirements Control

- **Baseline:** SRS approved Week 11
- **Change Control:** Change request → Impact assessment → Team approval → Instructor notification (if significant)

- **Traceability:** Requirements → Design → Code → Tests matrix maintained

3.3.2 Schedule Control

- **Monitoring:** Daily standups (Slack), weekly meetings, bi-weekly sprint reviews
- **Metrics:** Sprint velocity, burndown charts, milestone achievement
- **Thresholds:** Green (on track), Yellow (5-15% behind), Red (>15% behind)
- **Actions:** Yellow = reassign resources; Red = emergency meeting, scope reduction

3.3.3 Budget Control

- **Monitoring:** Bi-weekly cloud service usage review
- **Alerts:** Set billing alerts at $10, $30, $50
- **Actions:** If approaching $60, reduce usage or switch to free alternatives

3.3.4 Quality Control

- **Code Standards:** CamelCase naming, max 50 lines/method, Javadoc comments
- **Coverage Target:** ≥70% overall, ≥90% critical modules
- **Reviews:** 100% code reviewed via pull requests
- **Metrics:** Test coverage, defect density, build success rate

3.3.5 Reporting

- **Weekly:** Progress updates in team meetings
- **Formal:** Status Reports (Week 9, Week 13)
- **Final:** Complete project summary Week 16

3.3.6 Metrics Collection

- Defect count (GitHub Issues) - continuous
- Velocity (story points/sprint) - bi-weekly
- Build success rate - continuous
- Test pass rate – daily

3.4 Risk Management Plan

Top 10 Risks

ID	Risk	Prob	Impact	Mitigation	Contingency
R1	Team member unavailable	L	H	Cross-training, documentation	Redistribute work
R2	JavaFX learning curve delays	M	M	Early training, pair programming	Simplify UI
R3	QR security vulnerabilities	M	H	Research best practices, encryption	Accept with documentation
R7	Schedule delays	M	H	Realistic estimates, buffer time	Reduce scope
R8	Conflicting course deadlines	H	M	Early planning, time management	Negotiate extensions
R10	Poor code quality	M	M	Code reviews, standards	More testing time
R11	Major architectural flaws	L	H	Design reviews, early prototyping	Major refactoring
R13	Git merge conflicts	M	L	Branch strategy, frequent syncs	Manual resolution
R17	Scope creep	M	M	Strict scope, change control	Freeze scope post-SRS
R19	MySQL platform issues	L	M	Test multiple platforms early	Accept minor differences

Monitoring: Top 5 risks reviewed weekly, full register bi-weekly

3.5 Closeout Plan

Week 16 Activities:

- Verify all deliverables submitted
- Archive code and documentation in GitHub
- Conduct team retrospective (2-3 hours)

- Document lessons learned
- Create final project report
- Team celebration

4. Technical Process Plans

4.1 Process Model: Agile (Scrum)

Rationale: Iterative development, flexibility for learning curve, ideal team size (6), fits 16-week timeline with multiple deliverables

Sprint Structure:

- **Length:** 2 weeks
- **Total Sprints:** 6-7 sprints (Weeks 5-16)
- **Ceremonies:** Sprint Planning (2-3 hrs), Daily Standup (15 min via Slack), Sprint Review (1-2 hrs), Retrospective (1 hr)

Sprint Breakdown:

- **Sprint 0 (Weeks 1-4):** Project setup, proposal
- **Sprint 1 (Weeks 5-6):** SPMP, database design, UI wireframes
- **Sprint 2 (Weeks 7-8):** Database implementation, backend start
- **Sprint 3 (Weeks 9-10):** Backend core, SRS
- **Sprint 4 (Weeks 11-12):** Frontend UI, QR system
- **Sprint 5 (Weeks 13-14):** Integration, SDD
- **Sprint 6 (Weeks 15-16):** Testing, finalization, STS

Definition of Done:

- Code complete and reviewed
- Unit tests pass (>70% coverage)
- Integration tests pass

- Documentation updated
- Accepted by team leader

4.2 Methods, Tools, and Techniques

Languages: Java 11+, SQL, HTML5, CSS3, JavaScript

Frameworks/Libraries:

- JavaFX (desktop UI), Scene Builder
- JDBC/Hibernate (database)
- ZXing (QR generation)
- html5-qrcode (web QR scanning)
- JUnit 5, TestFX, Mockito (testing)

Development Tools:

- IDE: IntelliJ/VS code, Eclipse
- Database: MySQL 8.0, MySQL Workbench
- Version Control: Git, GitHub
- Build: Maven
- Design: Figma/Draw.io, Scene Builder

Standards:

- Object-Oriented Design (OOD)
- MVC architectural pattern
- 3NF database normalization

4.3 Test-Driven Development Infrastructure

Development Environment:

- Personal laptops (8GB+ RAM, i5+ processor)
- JDK 11+, MySQL 8.0 local instances
- IDE, Scene Builder, Git

Testing Environment:

- Local MySQL with test data
- Multiple browsers for web scanner
- Smartphones (iOS/Android) for QR testing

Production/Demo Environment:

- Cloud MySQL database (AWS/Azure/DigitalOcean) - Week 13
- Web hosting for QR scanner (GitHub Pages/Netlify) - Week 12
- Desktop app distributed as executable JAR

Network: Home/university WiFi, GitHub (HTTPS/SSH), Slack/Discord

4.4 Product Acceptance

Acceptance Authority: Saeed Matar Alshahrani

Functional Criteria:

- User login, inventory CRUD, stock alerts
- Supplier and purchase order management
- QR code generation and smartphone scanning
- Attendance recording with timestamps
- Reporting dashboards
- Role-based access control

Non-Functional Criteria:

- Usability: Intuitive UI
- Performance: <2 sec response times
- Security: Password hashing, SQL injection prevention

Acceptance Methods:

- Live demonstration (Week 16/17)
- Code and documentation review
- Testing verification (STS)

- Q&A session

Acceptance Checklist (Week 16):

- All code committed to GitHub
- All tests pass
- JAR file created and tested
- Database deployed to cloud
- Web scanner deployed and accessible
- All documentation complete
- Demo environment tested
- Presentation rehearsed

5. Supporting Process Plans

5.1 Documentation Plan

Document	Primary Author	Due Date
SPMP	Jawad (all contribute)	Week 8
SRS	Mohammad (all contribute)	Week 11
SDD	Abdullah J. (all contribute)	Week 13
STS	Mohammad	Week 15
User Manual	Mohammad, Abdullah J.	Week 16
Technical Docs	Mohammad	Week 16
Code Comments	Each developer	Ongoing

Standards:

- Follow provided templates
- Professional language, no spelling errors

- All sections complete (no TBD)
- Version control in GitHub

Review Process: Draft → Self-review → Peer review → QA Lead review → Approval → Submission

5.2 Quality Assurance

QA Lead: Mohammad Khalid Alqallaf

Activities:

- Code reviews (100% via PRs)
- Requirements/Design reviews (Weeks 10, 12)
- Process audits (bi-weekly)
- Testing coordination
- Metrics tracking

Quality Standards:

- Code: Follow standards, >70% coverage, no critical warnings
- Design: High cohesion, low coupling, SOLID principles
- Documentation: Complete, professional, error-free

5.3 Configuration Management

Tool: Git + GitHub

Branching Strategy:

- **main:** Production-ready (protected, requires PR approval)
- **develop:** Integration branch
- **feature/<name>:** New features
- **bugfix/<name>:** Bug fixes

Commit Standards:

- Format: <type>: <summary> (e.g., feat: add QR scanner, fix: login validation)
- Commit daily, push regularly

Pull Request Process:

1. Create PR from feature to develop
2. Peer review required
3. All tests must pass
4. Approval → Merge → Delete feature branch

Version Numbering: Semantic versioning (MAJOR.MINOR.PATCH)

- v0.1.0, v0.2.0... → v1.0.0 (final submission)

Baselines:

- SRS Baseline (Week 11)
- SDD Baseline (Week 13)
- Code Baseline v1.0 (Week 16)

5.4 Problem Resolution

Tool: GitHub Issues

Process:

1. Report problem (any team member)
2. Triage and prioritize (daily standup)
3. Assign to responsible person
4. Resolve and test fix
5. Review via PR
6. Close issue

Priority Levels:

- **Critical:** Fix immediately (<24 hrs)
- **High:** Fix within 1-2 days
- **Medium:** Fix within current sprint
- **Low:** Fix when time allows

Root Cause Analysis: For critical/high issues and recurring problems (5 Whys technique)

6. Additional Plans (Optional)

6.1 Security Plan

Requirements:

- Password hashing (BCrypt)
- Role-based access control (RBAC)
- SQL injection prevention (prepared statements)
- QR code encryption/signing to prevent forgery
- Input validation and sanitization
- HTTPS for web QR scanner

Responsibilities: Jawad (authentication), Mustafa (QR security), All (input validation), Mohammad (security testing)

6.2 Deployment Plan

Schedule:

- Week 13: Cloud database setup and deployment
- Week 12-13: Web scanner deployment
- Week 15-16: JAR packaging and distribution
- Week 16: Final deployment verification

Installation: User installs Java 11+ JRE, downloads JAR, configures database connection, runs application

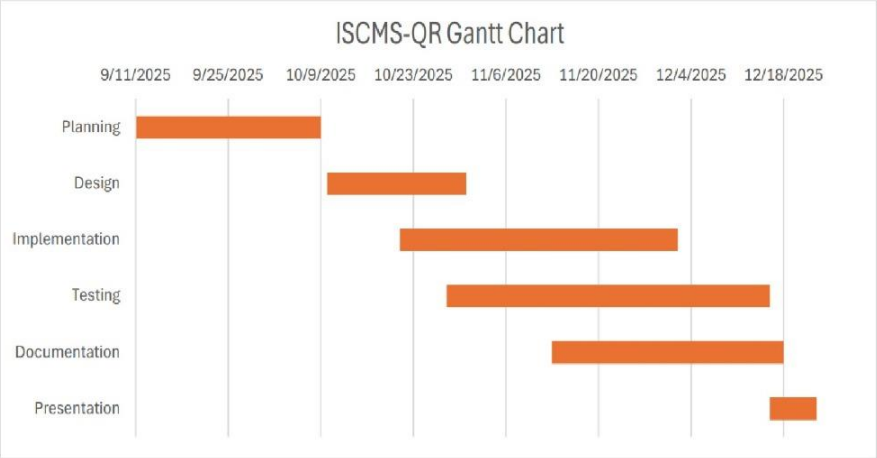
6.3 Training Plan

Team Training: See Section 3.1.4 (JavaFX, ZXing, HTML5 Camera API, Agile, TestFX, Git)

User Training: User Manual (Week 16) with step-by-step instructions, screenshots, organized by role

7. Appendices

Appendix A: Gantt Chart



Appendix B: WBS Diagram

IAU
جامعة الإمام عبدالرحمن بن فيصل
IMAM ABDULRAHMAN BIN FAISAL UNIVERSITY
كلية علوم الحاسب وتقنية المعلومات
College of Computer Science and Information Technology

CSC 305: Software Engineering

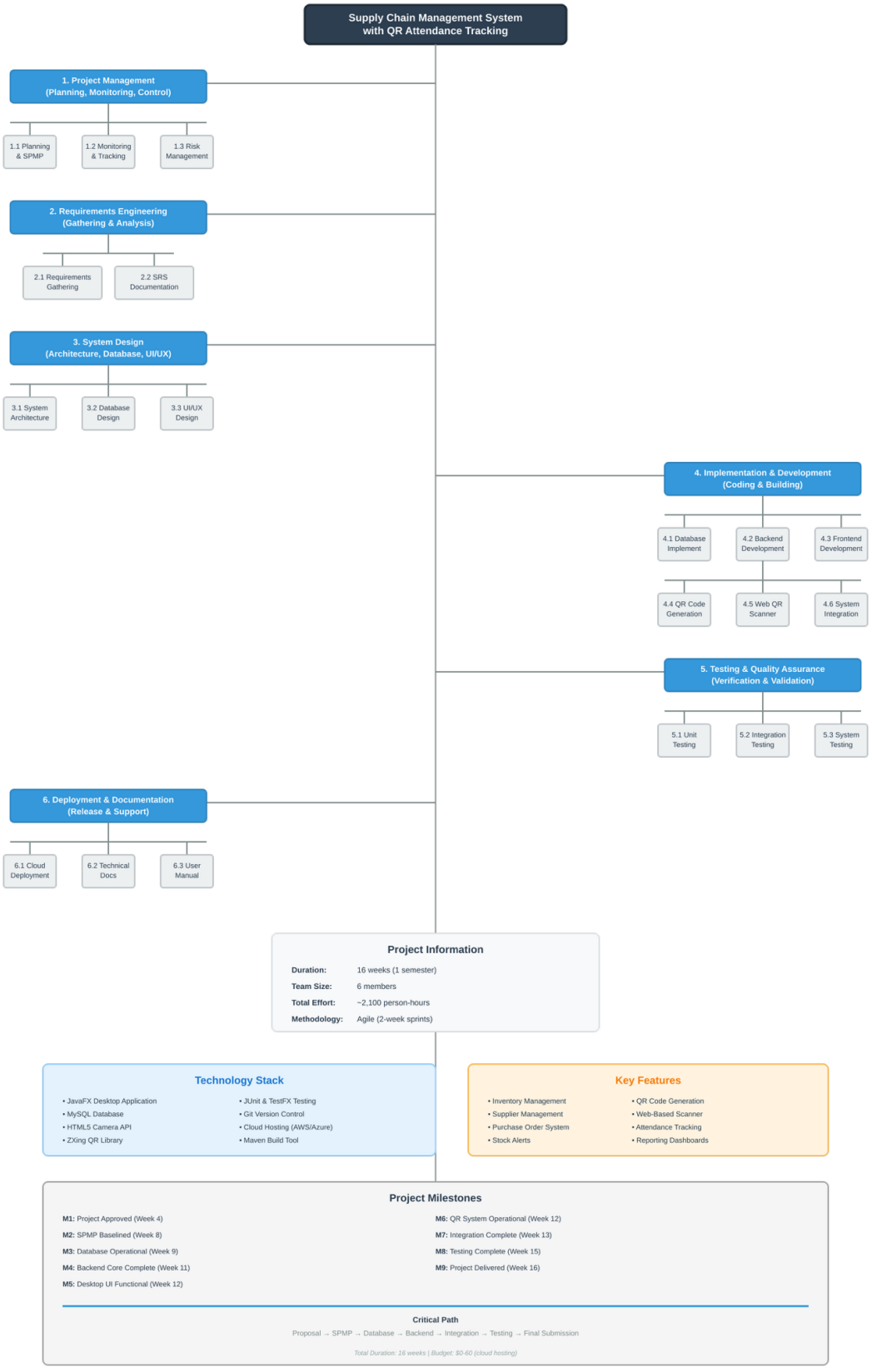
Term: 1

Academic Year: 2025/2026

Course Coordinator: Dr. Rahma Ahmed

Work Breakdown Structure (WBS)

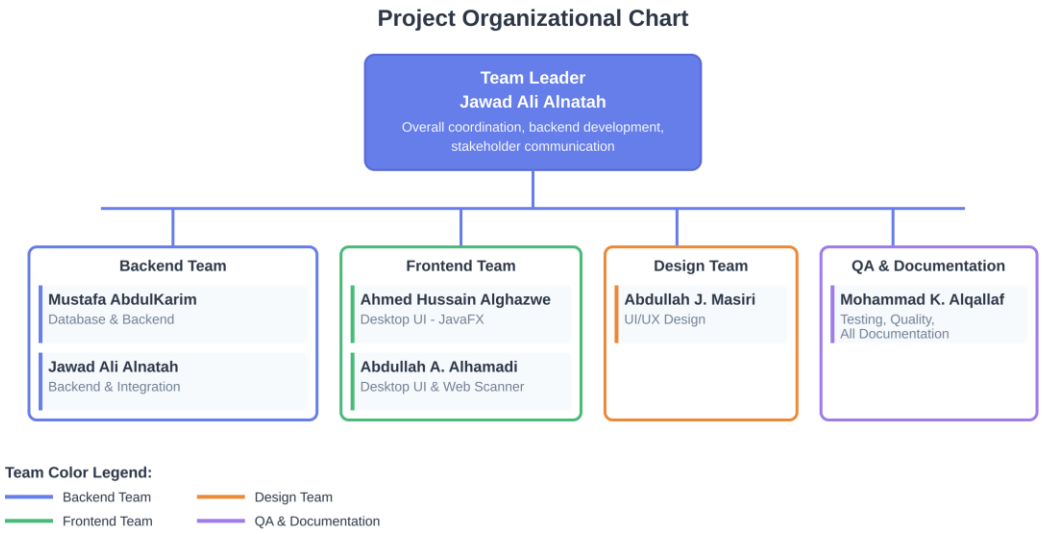
Integrated Supply Chain Management System with QR-Based Attendance Tracking



Appendix C: Meeting Schedule

Meeting	Frequency	Duration	Medium
Daily Standup	Weekdays	15 min	Slack (async)
Team Meeting	Weekly	1-2 hrs	TBD
Sprint Planning	Bi-weekly	2-3 hrs	TBD
Sprint Review	Bi-weekly	1-2 hrs	TBD
Sprint Retrospective	Bi-weekly	1 hr	TBD

Appendix D: Project Organization Chart



Evaluation

To be completed by the instructor or supervisor.

3. Project SPMP (18 points / 3 marks) – Week 8

Section	Excellent (Full Points)	Good (75%)	Fair (50%)	Poor (25% or below)	Points
1. Project Overview (3 pts)	Clear purpose, well-defined scope, objectives measurable, risks realistic, deliverables & schedule complete.	Purpose/scope adequate, some objectives unclear, limited risks, minor gaps in deliverables/schedule.	Vague or incomplete scope, objectives weak, little risk identification.	Missing or irrelevant.	/3
2. Project Organization (2 pts)	Well-structured team org, clear roles/responsibilities, external interfaces defined.	Team roles mostly clear, some vagueness in structure or interfaces.	Roles incomplete or unclear, poor organization.	Missing or irrelevant.	/2
3. Managerial Process Plans (5 pts)	Estimates realistic, WBS & schedule clear, tracking plan defined, risks detailed with mitigation, closeout plan solid.	Most elements present but some weak (e.g., vague WBS or shallow risk analysis).	Several parts incomplete or poorly justified.	Very minimal or missing.	/5
4. Technical Process Plans (3 pts)	Process model justified, tools/methods listed, infrastructure described, acceptance criteria realistic.	Adequate but some parts generic or weak justification.	Vague process model, missing tools/infrastructure details.	Very poor or absent.	/3

Section	Excellent (Full Points)	Good (75%)	Fair (50%)	Poor (25% or below)	Points
5. Supporting Process Plans (3 pts)	Documentation plan complete, QA standards clear, version control explained, bug tracking method defined.	Most parts covered but not in detail.	Minimal coverage (e.g., only documentation).	Missing or irrelevant.	/3
6. Additional Plans (1 pt)	At least one plan (training, security, maintenance) included.	Mentioned but vague.	Minimal or irrelevant.	Not included.	/1
7. Overall Quality (1 pt)	Well-written, well-structured, free of major grammar/formatting issues, follows template fully.	Clear but some formatting/clarity issues.	Hard to follow, weak formatting.	Poorly presented or incomplete.	/1

Note: If a student is listed as a member at the beginning of the SPMP **but not mentioned within the tasks, roles, or responsibilities throughout the document**, that student **will not receive the same score as active contributors**.

- Marks for that student may be reduced to reflect lack of contribution.
- Instructors should verify **role allocation tables, WBS assignments, and responsibilities** for evidence of individual involvement