SecureMed Team Contributions

Project Overview

Team: 5 Members

Duration: 12 Weeks (4 Sprints)

Total Effort: ~600 hours (5 members × 15 hrs/week × 8 weeks of development)

Team Members & Contributions

1. Stefan Dumitrasku - **Backend Lead & System Architect**

Primary Responsibilities:

- Database schema design and implementation
- Backend API development (Flask)
- Encryption system (Fernet AES-256)
- System integration and testing
- Risk analysis framework

Sprint Contributions:

- **Sprint 1:** Designed database schema with encryption for sensitive fields
- **Sprint 2:** Implemented database CRUD functions with encryption layer
- **Sprint 3:** Integrated Flask backend with React frontend, resolved API issues
- **Sprint 4:** Conducted end-to-end system testing, performance optimization

Key Deliverables:

- `backend.py` - Main Flask API server
- `setup_database.py` - Database initialization
- `encrypt_data.py` - Data encryption utilities
- `risk_analysis.py` - Threat modeling and risk matrix
- Integration testing suite

Hours: ~120 hours

---

2. Ana Salazar - **Security Engineer & Authentication Specialist**

Primary Responsibilities:

- Authentication & authorization systems
- Security testing and vulnerability assessment
- HTTPS/TLS implementation
- API security (JWT tokens, rate limiting)
- Final security audit

Sprint Contributions:

- **Sprint 1:** Researched HIPAA requirements and security standards
- **Sprint 2:** Built secure Flask API endpoints with authentication
- **Sprint 3:** Implemented JWT authentication, enforced HTTPS, conducted penetration testing
- **Sprint 4:** Performed comprehensive security audit, documented findings

Key Deliverables:

- Password reset system with DOB + SSN verification
- Role-based access control (RBAC)
- Session management
- Security audit documentation
- Input sanitization and SQL injection prevention

Hours: ~120 hours

---

3. Jordan Burgos - **Frontend Developer & UI/UX Designer**

Primary Responsibilities:

- React component development
- User interface design
- Tailwind CSS styling
- Frontend-backend integration
- Responsive design implementation

Sprint Contributions:

- **Sprint 1:** Created initial mockups and wireframes
- **Sprint 2:** Built React + Tailwind prototype (patient list, forms)
- **Sprint 3:** Expanded frontend functionality (audit display, edit forms, PDF triggers)
- **Sprint 4:** UI/UX polish, responsive layouts, accessibility improvements

Key Deliverables:

- `dashboard_react.html` - Admin dashboard components
- `user_dashboard_react.html` - Nurse dashboard
- `edr.html` - EDR panel interface
- `directory.html` - PHI destinations directory
- Complete UI component library

Hours: ~120 hours

---

4. Jeremiah Luzincourt - **Cybersecurity Analyst & Scanner Developer**

Primary Responsibilities:

- Vulnerability detection module
- Threat demonstration and analysis
- Scanner integration
- EDR (Endpoint Detection & Response) features
- HIPAA violation tracking

Sprint Contributions:

- **Sprint 1:** Researched common healthcare vulnerabilities
- **Sprint 2:** Demonstrated SQL injection and prevention methods
- **Sprint 3:** Built threat detection module prototype (2 vulnerability types)
- **Sprint 4:** Enhanced scanner to detect 5+ vulnerability types, integrated with reports

Key Deliverables:

- Vulnerability scanner engine
- EDR panel with real-time threat detection
- HIPAA violation logging system
- Threat demonstration scripts
- Live remediation features ("Apply Patch" buttons)

Hours: ~120 hours

---

5. Mumin Tahir - **Documentation Lead & Report Generation Specialist**

Primary Responsibilities:

- PDF report generation
- Technical documentation
- ReportLab integration

- Report formatting and design
- User guides and manuals

Sprint Contributions:

- **Sprint 1:** Created project documentation structure
- **Sprint 2:** Built PDF generation prototype with ReportLab
- **Sprint 3:** Automated PDF reporting integration with backend
- **Sprint 4:** Finalized PDF layout with SecureMed branding, prepared demo materials

Key Deliverables:

- `generate_report.py` - PDF generation system
- Risk analysis reports (JSON + visual)
- 130+ page comprehensive documentation
- User guides and setup instructions
- Presentation materials

Hours: ~120 hours

Sprint Summary

Sprint 1 (Weeks 1-2): Planning & Research

Focus: Requirements gathering, HIPAA research, technology selection

Deliverables:

- Database schema design (Stefan)
- Security requirements document (Ana)
- UI mockups (Jordan)
- Vulnerability research (Jeremiah)
- Documentation structure (Mumin)

Sprint 2 (Weeks 3-4): Core Development

Focus: Backend implementation, initial frontend, security demos

Deliverables:

- Database implementation with encryption (Stefan)
- Secure API endpoints (Ana)
- React/Tailwind prototype (Jordan)
- SQL injection demo (Jeremiah)
- PDF generation prototype (Mumin)

Sprint 3 (Weeks 5-6): Integration & Expansion

Focus: Full-stack integration, security enforcement, feature expansion

Deliverables:

- Backend-frontend integration (Stefan)
- JWT auth + HTTPS + penetration testing (Ana)
- Expanded UI functionality (Jordan)
- Threat detection module (Jeremiah)
- Automated PDF integration (Mumin)

Sprint 4 (Weeks 7-8): Testing & Polish

Focus: System testing, security audit, UI polish, final preparations

Deliverables:

- End-to-end testing + debugging (Stefan)
- Comprehensive security audit (Ana)
- UI/UX improvements (Jordan)
- Enhanced scanner + reporting (Jeremiah)
- Final PDF formatting (Mumin)

Collaborative Efforts

Pair Programming Sessions:

- Stefan & Ana: API security implementation
- Jordan & Stefan: Frontend-backend integration
- Jeremiah & Ana: Vulnerability testing and validation
- Mumin & Jeremiah: Scanner output formatting

Code Reviews:

- All code changes reviewed by at least 2 team members
- Security-critical changes reviewed by Ana + one other member
- UI changes reviewed by Jordan + Stefan

Weekly Meetings:

- Sprint planning (every 2 weeks)
- Daily stand-ups (async via Slack)
- Demo sessions (end of each sprint)
- Retrospectives and continuous improvement

Technology Stack (Team Decision)

Backend: Python + Flask (Stefan & Ana)

Frontend: React 18 + Vanilla JS (Jordan)

Database: SQLite with Fernet encryption (Stefan)

Security: HTTPS, JWT, RBAC (Ana)

Testing: unittest + requests (Stefan & Jeremiah)

Documentation: Markdown + ReportLab (Mumin)

Version Control: Git + GitHub (All)

Final Statistics

- **Total Lines of Code:** ~3,500+
- **API Endpoints:** 25+
- **React Components:** 8
- **Database Tables:** 5
- **Test Cases:** 34 (14 unit + 20 advanced)
- **Sprint Meetings:** 5 completed, 1 more planned
- **Hours Logged:** ~600 hours total

Individual Strengths Utilized

Stefan: System architecture, database design, integration

Ana: Security expertise, authentication, compliance

Jordan: UI/UX design, frontend development, user experience

Jeremiah: Cybersecurity analysis, threat modeling, vulnerability detection

Mumin: Technical writing, documentation, report formatting

This project represents a true team effort where each member's expertise contributed to a professional, production-ready healthcare security solution. □