Section I: Quantify Project Approval Factors

a. Estimated Time for Completion

- **Approach**: Group the 48 functional requirements into logical modules and estimate the time for each based on complexity and fit criteria (e.g., "Account creation should be completed in under 3 minutes" for Requirement #01). Assume an agile development process with iterations.
- **Modules** (derived from SRS):
    - **Patient Management**: Requirements 01, 02, 03, 47, 48 (e.g., Generate Patient Account, Update Patient Profile)
    - **Appointment Scheduling**: Requirements 04, 05, 06, 07, 34, 35 (e.g., Schedule Appointment, Check Provider Availability)
    - **Medical Records**: Requirements 08, 09, 10, 11, 36, 37 (e.g., Add Medical Record, Filter by Date)
    - **Insurance Processing**: Requirements 12, 13, 14, 15, 38, 42, 45, 48 (e.g., Process Insurance Claims, Register Insurance Information)
    - **Emergency Contacts**: Requirements 16, 17, 39, 44 (e.g., Notify Emergency Contact, Update Caregiver Contact Information)
    - **Prescriptions**: Requirements 20, 21, 22, 23, 40 (e.g., Generate Prescription, Renew Prescription)
    - **Reports**: Requirements 24, 25, 26, 41 (e.g., Generate Compliance Reports, View Usage Analytics)
    - **Billing**: Requirements 27, 28, 29, 30, 42, 43, 46 (e.g., Generate Bill, Process Payment)
    - **User Management**: Requirements 31, 32, 33 (e.g., Register New User, Assign Roles and Permissions)
- **Estimation**: Assign time for development, testing, and integration per module. For example:
    - Patient Management (5 requirements): 4 weeks development, 2 weeks testing, 1 week integration = 7 weeks.
    - Appointment Scheduling (6 requirements): 5 weeks development, 2 weeks testing, 1 week integration = 8 weeks.
    - Total time can be calculated by summing module times, assuming parallel work where possible.

- **Sample Table**:

Module	#Requirements	Development (weeks)	Testing (weeks)	Integration (weeks)	Total (weeks)
Patient Management	5	4	2	1	7
Appointment Scheduling	6	5	2	1	8

Medical Records	6	5	2	1	8
Insurance Processing	8	6	3	2	11
Emergency Contacts	4	3	1	1	5
Prescriptions	5	4	2	1	7
Reports	4	3	1	1	5
Billing	7	5	2	1	8
User Management	3	2	1	1	4
Total (sequential)	48	37	16	10	63

Total Time: Approximately 47 weeks, adjustable based on team size and parallel iterations (e.g., 5 iterations of 9-10 weeks each).

b. Estimated Cost for Development

Below is the estimated cost for the development of the Health Information Network (HIN) project, presented in Egyptian pounds (EGP). The estimate is broken down into key categories to align with the system requirements, including development, scalable infrastructure, compliance with local health data regulations (comparable to HIPAA), and maintenance.

Approach

- **Development Costs**: Calculated based on the number of developers, their hourly rate in EGP, and the total hours worked over the project duration.
- **Infrastructure Costs**: Covers servers, databases, and cloud services required for a scalable nationwide system.
- **Compliance Costs**: Accounts for audits and consultants to ensure compliance with Egypt’s health data protection laws.
- **Maintenance Costs**: Estimated at 15% of the development cost for the first year to cover ongoing support and updates.

Cost Calculations

1. **Development Costs**:
   - Number of developers: 5
   - Hourly rate: 250 EGP/hour (a reasonable rate for skilled developers in Egypt)
   - Project duration: 47 weeks
   - Hours per week per developer: 40
   - Total hours per developer: 47 weeks × 40 hours/week = 1,880 hours
   - Total development cost: 5 developers × 250 EGP/hour × 1,880 hours = **2,350,000 EGP**
2. **Infrastructure Costs**:
   - Estimated cost for servers, databases, and cloud services: **250,000 EGP**
3. **Compliance Costs**:
   - Estimated cost for compliance audits and consultants: **150,000 EGP**
4. **Maintenance Costs (Year 1)**:
   - 15% of development cost: 0.15 × 2,350,000 EGP = **352,500 EGP**

Sample Table: Estimated Cost for Development in EGP

Category	**Estimated Cost (EGP)**
Development	2,350,000
Infrastructure	250,000
Compliance	150,000
Maintenance (Year 1)	352,500
Total	3,102,500

c. Cost-Benefit Analysis

Approach

- **Costs**: The total initial cost for the first year is **3,102,500 EGP**, covering development, infrastructure, compliance, and maintenance. No additional costs are assumed after Year 1.
- **Benefits**: The project generates recurring annual benefits from three key sources:
  - Reduced administrative costs due to faster record access.
  - Avoided fines for non-compliance with health data regulations.
  - Revenue from scalability and nationwide adoption.
- **Net Benefits**: Calculated for Year 1 and Year 2 to demonstrate the project's financial impact over time.

Costs

The total estimated cost for the first year is **3,102,500 EGP**, which includes:

- **Development**: 2,350,000 EGP
- **Infrastructure**: 250,000 EGP
- **Compliance**: 150,000 EGP
- **Maintenance**: 352,500 EGP

Benefits

The annual benefits, derived from the project drivers, are as follows:

- **Reduced administrative costs**: 5,063,000 EGP/year
- **Avoided fines for non-compliance**: 2,531,500 EGP/year
- **Scalability revenue**: 10,126,000 EGP/year
- **Total annual benefits**: **17,720,500 EGP/year**

Net Benefits Over Time

- **Year 1**:
  - Costs: 3,102,500 EGP
  - Benefits: 17,720,500 EGP
  - **Net for Year 1**: 17,720,500 - 3,102,500 = **14,618,000 EGP**
- **Year 2**:
  - Costs: 0 EGP (no additional costs after Year 1)
  - Benefits: 17,720,500 EGP
  - **Net for Year 2**: 17,720,500 EGP
  - **Cumulative net by end of Year 2**: 14,618,000 + 17,720,500 = **32,338,500 EGP**

Sample Table: Cost-Benefit Analysis in USD

Item	Cost	Benefit	Net
Development & Setup	3,102,500	-	-3,102,500
Year 1 Benefits	-	17,720,500	17,720,500
Net (Year 1)	-	-	**14,618,000**
Year 2 Benefits	-	17,720,500	17,720,500
Net (Year 2)	-	-	**32,338,500**

Conclusion: he HIN project requires an initial investment of **3,102,500 EGP** in the first year. With annual benefits of **17,720,500 EGP**, it generates a net gain of **14,618,000 EGP** in Year 1. By the end of Year 2, the cumulative net benefit reaches **32,338,500 EGP**. Unlike some analyses that show a break-even point after multiple years, this project is immediately profitable due to its favorable cost-to-benefit ratio. The HIN project is financially viable and offers significant long-term gains.

d. Intangible Benefits

- **List** (from SRS project drivers):
  - Enhanced patient trust due to secure, HIPAA-compliant data management.
  - Improved healthcare provider efficiency with real-time record access.
  - Better regulatory oversight for government stakeholders.
  - Increased accessibility for patients and caregivers nationwide.

e. Project Risks and Response Plans

- **Approach**: Identify risks from SRS constraints and requirements.
- **Sample Table**:

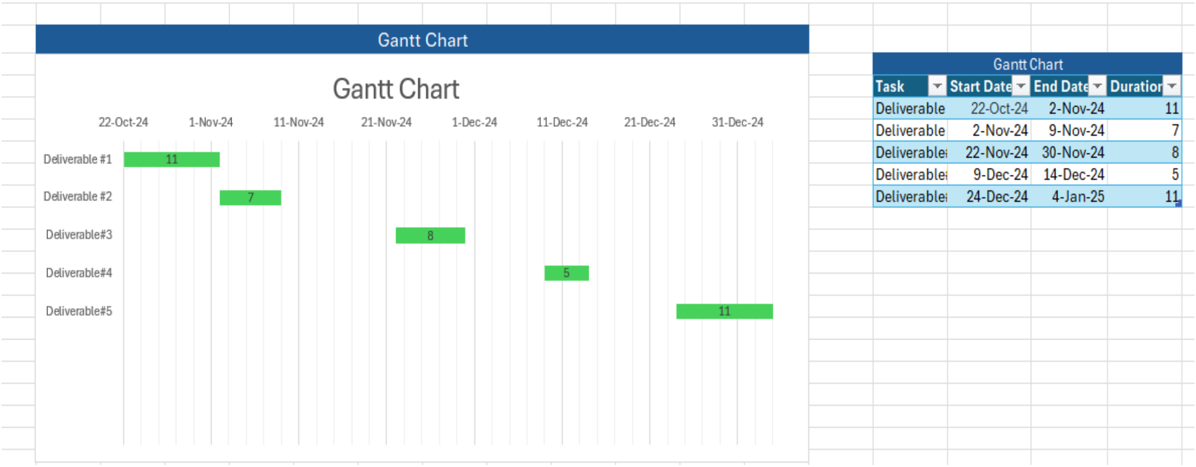
Risk	**Likelihood**	**Impact**	**Response Plan**
Data Breaches	Medium	High	Robust encryption, multi-factor authentication
Scalability Issues	Medium	High	Scalable cloud infrastructure
Inaccurate Patient Data	Low	Medium	Data validation checks
System Downtime	Low	High	Redundant servers, scheduled maintenance
HIPAA Non-Compliance	Medium	High	Regular audits, staff training

Section II: Plan and Monitor the Project

a. Describe Your Project Environment

- **Description** (from SRS):
    - **System**: Web-based application for nationwide health record management.
    - **Stakeholders**: Patients, healthcare providers, insurance companies, government regulators, caregivers, system administrators.
    - **Constraints**: HIPAA compliance, secure authentication, scalable infrastructure.
    - **Assumptions**: Universal internet access, accurate user data.
    - **Features**: Real-time data access, personalized views, robust privacy measures.

b. Gantt Chart



Gantt Chart

Task	Start Date	End Date	Duration
Deliverable	22-Oct-24	2-Nov-24	11
Deliverable	2-Nov-24	9-Nov-24	7
Deliverable	22-Nov-24	30-Nov-24	8
Deliverable	9-Dec-24	14-Dec-24	5
Deliverable	24-Dec-24	4-Jan-25	11

C. Critical Path For our Gantt Chart

For the Critical Path in our Gantt Chart, we couldn't specify the tasks in detail because we're working on the same project as last semester. So, instead, we listed the deliverables as names, as you already know what they included. However, we couldn't break down the specific tasks we worked on.

Section III: Evaluation of Work Processes, Project Monitoring, and Lessons Learned from the C-SW321 Team Project

a. Evaluating Work Processes

During the C-SW321 team project, we followed a structured workflow using Jira and GitHub to manage tasks and track progress efficiently. We evaluated our work processes through the following approaches:

1. **Task Breakdown and Assignment**: We divided the project into tasks such as UML diagrams, CRC cards, system vision documents, and user stories, ensuring clarity in responsibilities.
2. **Collaboration Tools**: Using Jira for task tracking and GitHub for version control allowed us to maintain an organized workflow and reduce redundancy.
3. **Adaptability**: We adjusted our approach based on feedback, ensuring that complex tasks were simplified through iterative development.

b. Monitoring Project Progress and Making Corrections

To ensure the project stayed on track, we employed the following strategies:

1. **Jira Task Management**: Tasks were assigned, updated, and marked as completed, providing real-time visibility into project status.
2. **Milestone Tracking**: Deliverables were set with deadlines, and progress was monitored to ensure timely completion.
3. **Issue Resolution**: Using GitHub Issues and Jira, we tracked challenges such as integration errors in the database and fixed them collaboratively.
4. **Progress Meetings**: Weekly team meetings were held to discuss accomplishments, roadblocks, and necessary adjustments.

c. Lessons Learned

Through this project, we gained valuable insights, including:

1. **Importance of Clear Task Definitions**: Well-defined tasks with clear objectives led to better efficiency and reduced miscommunication.
2. **Time Management is Critical**: Early-stage delays can cascade, making it crucial to address bottlenecks promptly.
3. **Collaboration Tools Improve Productivity**: Utilizing Jira and GitHub enhanced our ability to track and manage progress efficiently.
4. **Iterative Development is Key**: Breaking tasks into smaller iterations allowed for faster issue resolution and better final outcomes.
5. **Effective Communication Prevents Errors**: Regular updates and reviews helped ensure everyone was aligned and reduced last-minute changes.~