

Assignment-1:

i) Product Backlog for user stories:

User story 1: As a user, I want to log in securely so that I can access my account.

Tasks:

a) Backend:

- ✓ Implement authentication
- ✓ Develop login API.
- ✓ Encrypt passwords and implement validation.

b) Frontend:

- ✓ Design login UI.

c) Testing and Deployment:

- ✓ Write unit tests for login functionality.

User Story 2: As a user, I want to search for products by category to find items easily.

Tasks:

- Backend

b. Frontend

c. Testing and Deployment

ii) Prioritization in Sprint Planning:

a. Value to the customer

b. Technical feasibility

c. Sprint plan Decision

iii) Tracking with a scrum board:

A scrum board will track task progress with three columns:

To Do (planned)	In Progress	Done (completed)
Develop login API	Design login form UI	Implement password encryption
Implement search API	Implement search filters	Redirect user after login
Optimize search performance	Integrate API with UI	Validate login functionality

Assignment-2 :

Agile is the best choice for this project due to its high adaptability, iterative approach and cost effective evolution. It allows frequent adjustments based on client feedback while managing risks through incremental development and continuous testing.

Comparison of methodologies for high-risk, evolving project:

1. Spiral model:

- Risk management
- Adaptability: Allows changes at each phase but requires detailed planning.

Best for high risk project.

2. Agile methodology:

- Risk management: Reduce risk through continuous feedback.
- Adaptability: Highly flexible.

- ✓ Best for: Project with evolving requirements, tight budgeting and close client collaboration.

3. Extreme Programming (XP)

- ✓ Risk management: Addresses risk through rapid iterations.
- ✓ Adaptability: Extremely flexible but requires high client involvement.

- ✓ Best for: Small to medium teams with rapidly changing requirements.

Assignment - 3 :

Comparison of Development methodologies:

1. Waterfall:

- ✓ Best for: Fixed requirements and strict deadlines.
- ✓ Predictability: High, as all phases are planned upfront.
- ✓ Customer collaboration: Minimal feedback.

✓ Risk management: Limited flexibility, high risk

if changes arise late, difficult to track

2. Agile:

✓ Best for: Evolving requirements and continuous feedback.

✓ Predictability: Low, but adaptable

✓ Customer collaboration: High

✓ Risk management: Lower risk due to incremental changes.

3. Extreme Programming (XP):

✓ Best for: Rapidly changing requirements.

✓ Predictability: Low but ensures continuous improvement

✓ Customer collaboration: Very high with direct engagement.

✓ Risk management: Reduces risk through test-driven development.

4. Spiral model: Iterative framework

- ✓ Best for: High risk projects requiring iterative risk assessment.
- ✓ Predictability: Moderate, with flexibility for adjustments.
- ✓ Customer collaboration: Periodic feedback at each iteration.
- ✓ Risk management: Strong focus on risk analysis before each development phase.

Best methodology for each project.

Project A → waterfall

Project B → Agile or XP

Assignment-4:

Principle of software engineering ethics:

1: Public interest: Prioritize user safety, privacy and societal well-being.

2. Client & Employer Responsibility: Act in the best interest of stakeholders while maintaining honesty and integrity.
3. Product Quality: Ensure reliability, security, and efficiency in software.
4. Professional competence: Continuously improve skills and apply best practices.
5. Fairness and honesty
6. Confidentiality.
7. Respect for law and ethics.

Professional Responsibility issue:

- ✓ Security and privacy violation
- ✓ Bias and Discrimination

Role of ACM/IEEE code of Ethics

- ✓ Provides ethical guidelines
- ✓ Encourages transparency, fairness, and accountability.

#Assignment#

Functional Requirements :

1. Flight booking : Users can search and book flights.
2. Payment processing : Secure online payments via multiple methods.
3. User authentication : Login and account management for passengers.
4. Flight status and notification : Real-time updates on flight schedules.
5. Booking cancellation and refund.

Non-functional Requirements :

1. Performance : System should handle 1000+ concurrent users.
2. Security : Data encryption and multi-factor authentication.
3. Scalability : Supports growing users and expanding features.

4. Usability: Intuitive UI fast response time.

5. Maintainability: Modular design for easy updates and bug fixes.

Assignment - 6:

V-model of Testing in a plan-Driven software process:

The v-model (verification and validation model) is a sequential software development process where testing activities run parallel to development activities.

Illustration of v-model:

Requirements → Acceptance Testing

Design → System Testing

Architecture → Integration Testing

Coding → Unit Testing

Development flows downwards (left side of v) while

corresponding testing activities occurs upwards (right side of V).

phases and corresponding testing activities:

1. Requirement analysis → Acceptance testing
2. System design → System testing
3. Architecture design → Integration testing
4. Module design → Unit testing

Relationship between Development and Testing:

- ✓ Each development phase has a corresponding testing phase.
- ✓ Testing starts early, reducing defects in later stages.
- ✓ Ensure quality and minimize project risks.

Assignment-7:

Prototype development process: Prototype Development Process involves creating a preliminary version of the software to understand requirements and gather user feedback.

key stages of prototyping:

1. Requirement gathering
2. Quick design
3. Prototype development
4. User evaluation
5. Refinement based on feedback

Benefits of prototyping:

User feedback: Helps refine requirements early and improve usability.

- ✓ Risk Reduction: Identifies potential issues early.
- ✓ Iterative Development: Enables continuous improvement and adaptation to user needs.

The prototyping model enhances clarity, reduces risks, and ensures a better final product throughout the development process.

Assignment - 8:

Process Improvement Cycle:

The process improvement cycle in software engineering involves continuously evaluating and improving software development processes to increase efficiency and quality.

Key stages of the process improvement cycle:

1. Process measurements: collect data on current processes.

2. Analysis of existing processes.

2. Process Analysis: Analyze the collected data to identify inefficiencies.
3. Process improvement
4. Process implementation
5. Process control

commonly used process metrics:

1. Defect density
2. code churn
3. Cycle time
4. velocity

These metrics help monitor and refine the software process, enhancing efficiency, quality and overall performance.

Methodology: Utilizing various assessment criteria of best practices from various p.t.o.

#Assignment-09:

SEI capability maturity model (cmm):

The SEI capability maturity model (cmm) is a framework developed by the Software Engineering Institute to help organizations improve their software development processes. It defines five maturity levels that indicate the maturity of an organization's processes and provide a path for improvement.

Five levels of cmm:

1. Level 1: Initial (Ad hoc)

Description: Processes are unpredictable, poorly controlled and reactive.

2. Level 2: Managed

Increased project predictability and better management practices lead to improve

efficiency and reduced project failure.

3. Level-3: Defined

With standard processes in place, teams can follow best practices.

4. Level-4: Quantitatively managed

Performance is measured and managed using data, enabling more accurate prediction.

5. Level-5: Optimizing

The organization continuously refines its processes, leading to ongoing improvement in

quality, cost reduction and innovation ensuring

long-term success and competitiveness.

Impact on software development and organizational performance:

The CMMI provides organizations with a structured approach to software process improvement. By progressing through each level, an organization

can systematically enhance the development practice, leading to higher quality software, improved project management and better overall performance.

Assignment - 10:

Cone Principles of Agile Software Development:

1. customer collaboration
2. responding to change
3. working software
4. Individuals and interactions
5. Sustainable Development
6. Continuous improvement

Application in different environments:

- ✓ startups, Rapid Prototyping and Agile Iteration

✓ Large enterprises: Scaled Agile (SAFe) to align multiple teams.

✓ Regulated industries: Hybrid model balancing Agile flexibility with compliance.

Benefits of Agile methods:

- ✓ Faster delivery and early feedback
- ✓ Higher adaptability to changing requirements.
- ✓ Increased collaboration and team motivation

Challenges of Agile methods:

- ✓ Difficult to scale in large organizations.
- ✓ Requires continuous customer involvement.
- ✓ Less effective for projects with fixed requirements.

Assignment-12

Entity Relationship diagram: ER diagram is a diagram that represents relationships among entities in a database. It is commonly known as an ER diagram. An ER Diagram in DBMS plays a crucial role in designing the database.

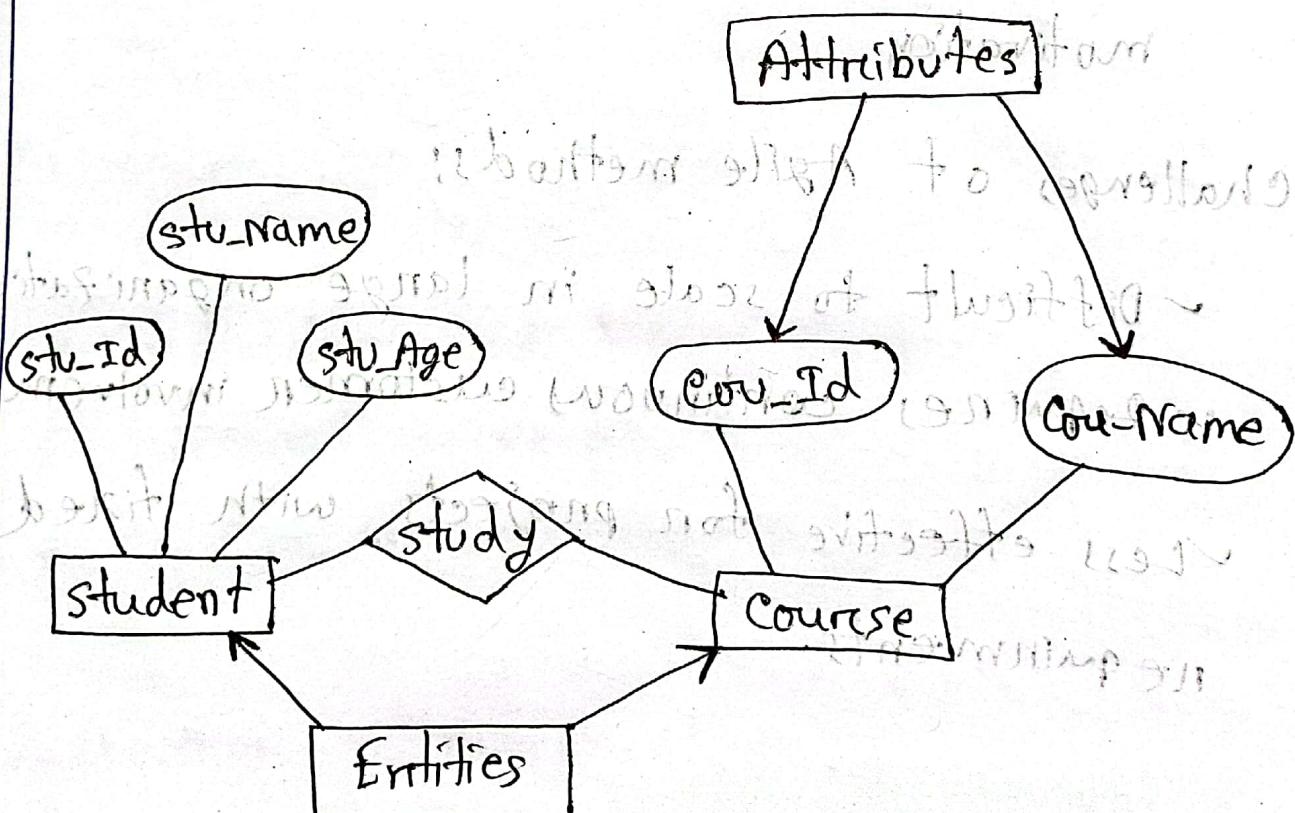


figure: ER Diagram

The following diagram showcase two entities - student and course and their relationship. The relationship described between student and course is many-to-many as a course can be opted by several students and a student can opt for more than one course. student entity possesses attributes - stu_id, stu_name and stu_Age. The course entity has attributes such as cou_id and cou_name.

Why use ER Diagram in DBms?

- ER diagram helps you conceptualize the database and lets you know which fields need to be embedded for a particular entity.
- ER Diagram gives a better understanding of the information to be stored in a database.