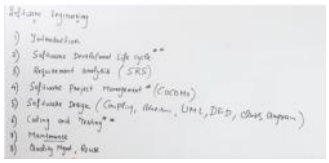
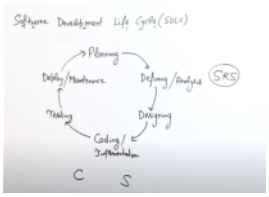
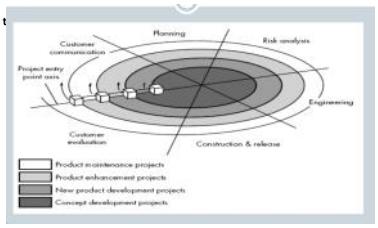
Software Engineering

07 January 2024 23:10

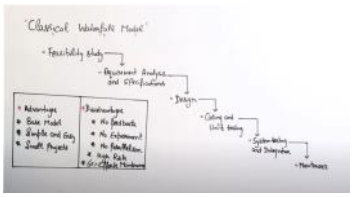


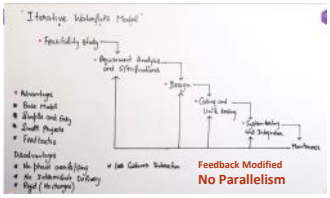
Software engineering is the systematic application of engineering principles t development, maintenance, and management of software for effective and reliable solutions.



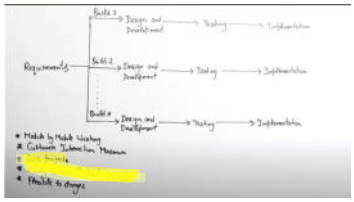


Modified Version of Classical Waterfall Model

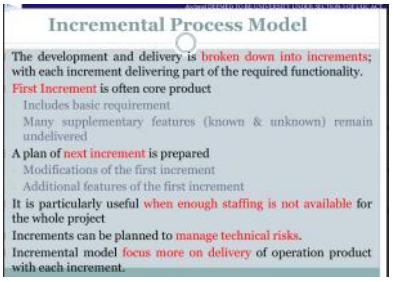


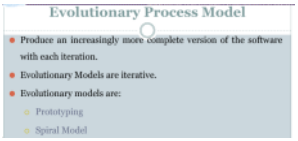


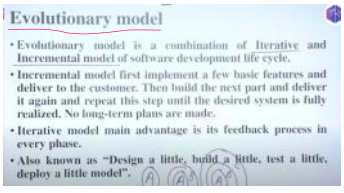
Incremental Model

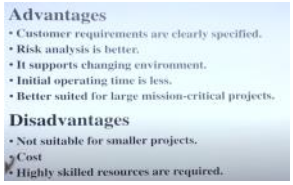


Isme hum alag alag model banate hai jaise hum university management system bna rahe hai to usme hum teacher ke liye alag module banayenge student ke liye alag se banayenge admin ke liye alag se..etc







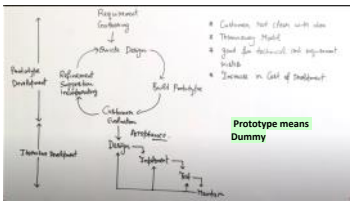


Phele ek module banaya fir uspar feedback karte rahe fir dusra model banaya uspar feedback and similarly process chalta rhaa

Jaise delhi metro hai usme phele yellow line fir feedback for green line fir feedback fir blu

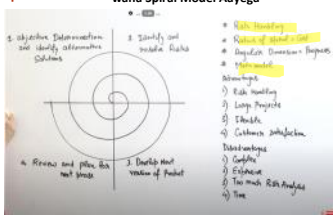
o etc..

Prototype Model (use when a customer not clear with the idea)

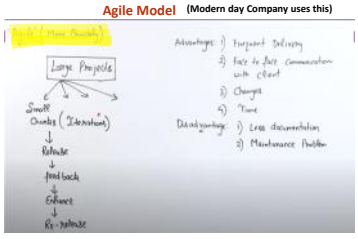


Spiral Model

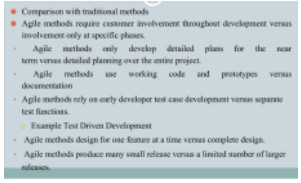
Jaha par bhi RISK work dikh gya waha Spiral Model Aayega



Agile Model **(Modern day Company uses this)**



Agile methods require customer involvement throughout development versus involvement only at specific phases.



Agile method applicability

- Product development where a software company is developing a small or medium-sized product for sale.
- Custom system development within an organization, where there is a clear commitment from the customer to become involved in the development process, and where there are not many external rules and regulations that affect the software.
- Because of their focus on small, tightly-integrated teams, there are problems in scaling agile methods to large systems.

Problems with agile methods

- It can be difficult to keep the interest of customers who are involved in the process.
- Team members may be unsuited to the intense involvement that characterizes agile methods.
- Prioritizing changes can be difficult where multiple stakeholders are involved.
- Maintaining simplicity requires extra work.

Q.1 [CO1][4 Marks] "A company wants to dev
Spotify's engineering team faced challenges i
including difficulty in managing the complexity
their distributed team members".
Based on the above case study which model wo
and why? Explain in details.

Q. 1 IC011[4 Marks] "A company wants to develop a new music streaming platform lik
Spotify's engineering team faced challenges in the early stages of their development p
including difficulty in managing the complexity of their codebase and coordinating the
Based on the above case study which model would you proposed to develop the abov
and why? Explain in details.

Classical Waterfall	Iterative Waterfall	Prototype Model	Incremental Model	Evolutionary Model	Spiral Model	Agile Model
Basic, Rigid, Inflexible, Not for Real Project	Basic, Problem is well understood	User Requirement Not clear, Costly, No Early Peek on Requirements → High User Involvement → Reusability	Module by Module Delivery, Easy to test and debug	Large Projects	Risk, Not for Small Projects, Less Experience Can work	Flexible, Advanced, Parallel, Process divided into sprints

SCRUM

- One of the most popular agile methodology.
- Scrum is a lightweight, iterative and incremental framework.
- Scrum breaks down the development phases into stages or cycles called "sprints".
- The development time for each sprint is maximized and dedicated, thereby managing only one sprint at a time.
- Scrum Team has scrum master and product owner with constant communications on the daily basis.
- Keywords: Backlog, Sprint, Daily Scrum, Scrum master, Product owner.

Advantages

- Freedom & Adaption
- High-quality, low-risk product.
- Reduce the development time up to 40%
- Scrum customer satisfaction is very important.
- Reviewing the current sprint before moving to new one.

Disadvantages:

- More efficient for small team size.
- No changes in the sprint.

Scrum:

Advantages:
- Flexibility and adaptability
- Frequent deliverables
- Transparency
- Collaboration
- Continuous improvement

Disadvantages:
- Learning curve
- Dependency on team cohesion
- Uncertainty in early estimates
- Not suitable for all projects

Extreme Programming (XP):

Advantages:
- Improved code quality
- Customer satisfaction
- Quick adaptation to changes
- Early defect detection
- Highly collaborative

Disadvantages:
- Requires a skilled team
- May be overly flexible
- Challenging for large teams
- Documentation challenges

Incremental Model:

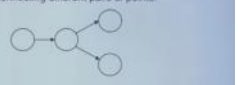
Advantages:
- Early deliveries
- Risk reduction
- Customer feedback
- Easier testing and debugging
- Flexibility in development

Disadvantages:
- Integration challenges
- Dependency management
- Documentation overhead
- Not suitable for small projects
- Increased cost

PERT/CPM

Important term :

1. **Network :** A network consist of a set of points and set of lines connecting different pairs of points.



2. **Events :** Events are commonly represented by circles in network diagram.

3. **Activity :** The arrows are called activity.

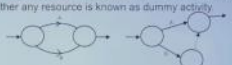
TYPE OF ACTIVITY

1. **Predecessor activity :** An activity which must be completed before one or more other activities start is known as predecessor activity.
2. **Successor activity :** An activity which started immediately after one or more of other activity are completed is known as successor activity.

Example :

Here A is predecessor activity of B and B is successor activity of A.

3. **Dummy activity :** An activity which does not consume either any resource is known as dummy activity.



Example : Draw a network diagram to express the following relationship :

Activity	Immediate Predecessor
A	-
B	-
C	A
D	A,B

Solution :

CONSTRUCTION OF NETWORK DIAGRAM

1. Activity of Node (AON) network.
2. Activity on Arrow (AOA) network.

	AOA Network	AON Network
1. Activity A		
2. B must follow A		
3. B & C must follow A		
4. C must follow A & B		

EVENT :- Activity of Node

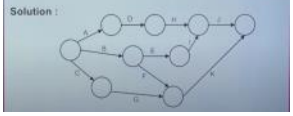
ACTIVITY :- Activity of arrow

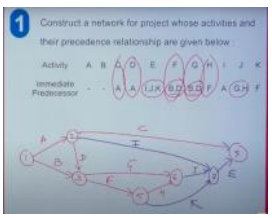
Question

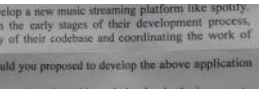
Example : Construct a network for the project whose activities and their predecence relationship are given below.

Activity	A	B	C	D	E	F	G	H	I	J	K
Predecessor	-	-	-	A	B	B	C	D	E	H,I	F,G

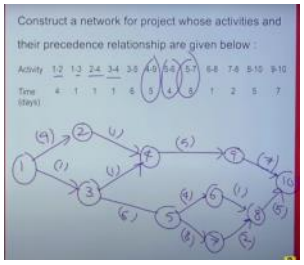
Solution :



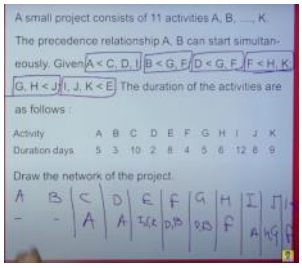


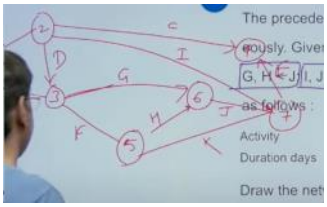

elop a new music streaming platform like spotify.
the early stages of their development process,
of their codebase and coordinating the work of
uld you proposed to develop the above application

e spouty.
rocess,
work of
e application

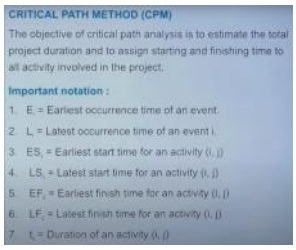


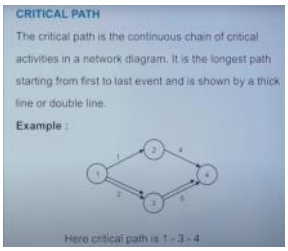
TRICK :- Jo activity nahi aa rahy vo smj lena ki last mein complete hogi means vo last mein connect hogi

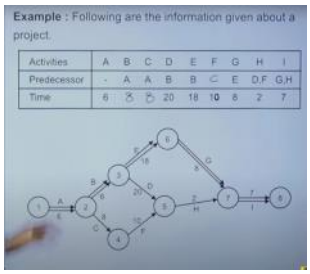


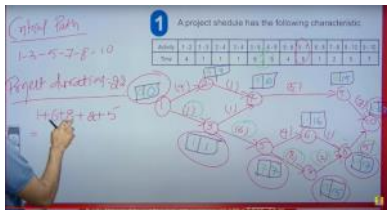


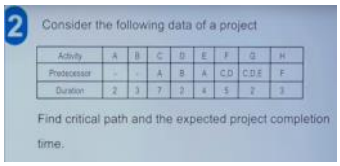
Critical Path Method

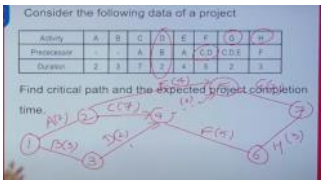


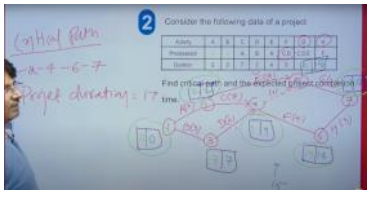






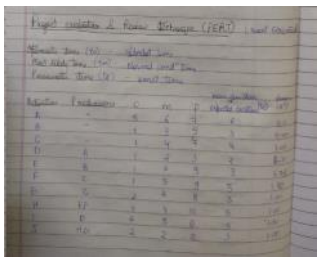


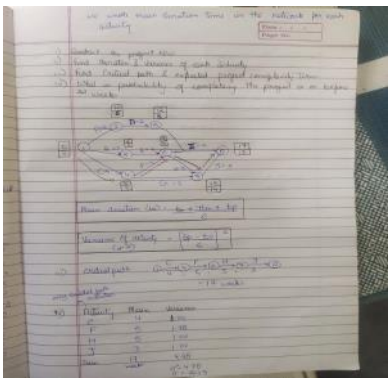


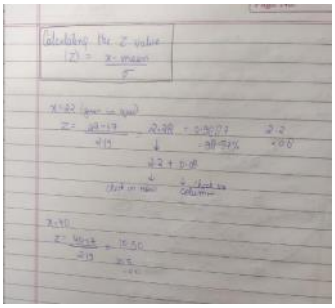


Yeh Wala question Important hai Kyuki isme Dummy aa raha hai

Project Evaluation & Review technique



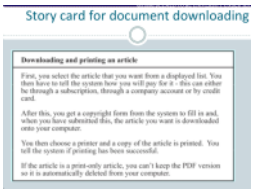


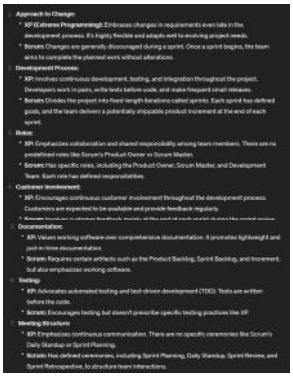


Extreme Programming

In XP, user requirements are expressed as scenarios or user stories.

These are written on cards and the development team break them down into implementation tasks. These tasks are the basis of schedule and cost estimates.





- **Approach to Change:**
  - **XP (Extreme Programming):** Embraces changes in requirements even late in the development process. It's highly flexible and adapts well to evolving project needs.
  - **Scrum:** Changes are generally discouraged during a sprint. Once a sprint begins, the team aims to complete the planned work without alterations.
- **Development Process:**
  - **XP:** Involves continuous development, testing, and integration throughout the project. Developers work in pairs, write tests before code, and make frequent small releases.
  - **Scrum:** Divides the project into fixed-length iterations called sprints. Each sprint has defined goals, and the team delivers a potentially shippable product increment at the end of each sprint.
- **Roles:**
  - **XP:** Emphasizes collaboration and shared responsibility among team members. There are no predefined roles like Scrum's Product Owner or Scrum Master.
  - **Scrum:** Has specific roles, including the Product Owner, Scrum Master, and Development Team. Each role has defined responsibilities.
- **Customer Involvement:**
  - **XP:** Encourages continuous customer involvement throughout the development process. Customers are expected to be available and provide feedback regularly.
  - [illegible]
- **Documentation:**
  - **XP:** Values working software over comprehensive documentation. It promotes lightweight and just-in-time documentation.
  - **Scrum:** Requires certain artifacts such as the Product Backlog, Sprint Backlog, and Increment, but also emphasizes working software.
- **Testing:**
  - **XP:** Advocates automated testing and test-driven development (TDD). Tests are written before the code.
  - **Scrum:** Encourages testing but doesn't prescribe specific testing practices like XP.
- **Meeting Structure:**
  - **XP:** Emphasizes continuous communication. There are no specific ceremonies like Scrum's Daily Standup or Sprint Planning.
  - **Scrum:** Has defined ceremonies, including Sprint Planning, Daily Standup, Sprint Review, and Sprint Retrospective, to structure team interactions.