

DEVOPS

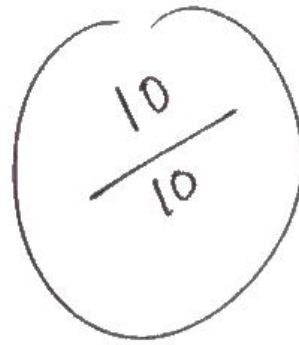
ASSIGNMENT - 1

Name: Devathi. Srida
Roll No: 20AT1A05F5
Branch: CSE - III
Section: C

1. List and Explain various software development models.

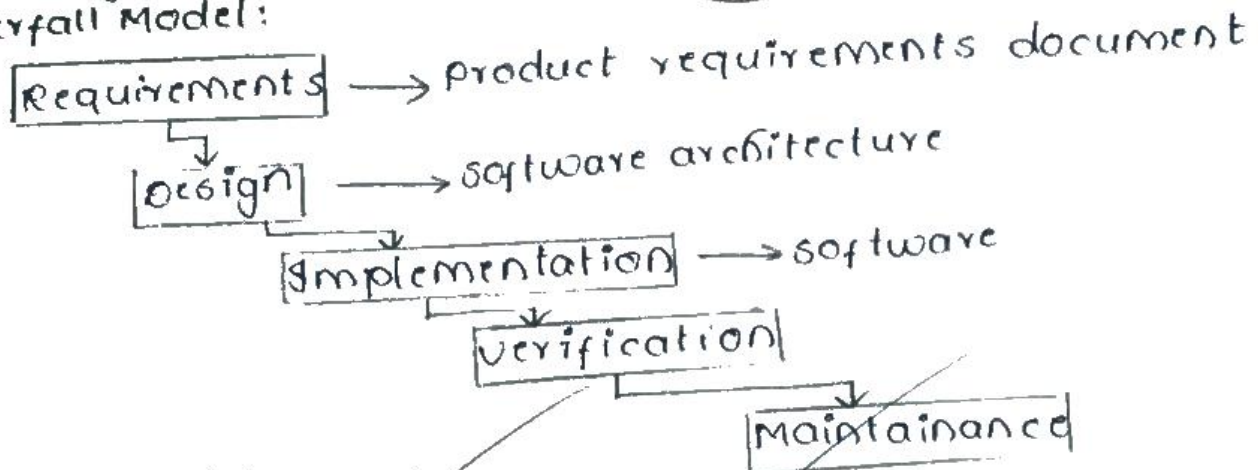
8 types of software development models:

1. Waterfall Model
2. V-Model
3. Incremental Model
4. RAD Model
5. Agile Model
6. Iterative Model
7. Spiral Model
8. Prototype Model



11/3/23

1. Waterfall Model:



Waterfall Model is a linear sequential flow, often used with projects that have defined set of requirements.

Advantages:

Simple and understandable, Waterfall Model is a manageable method.

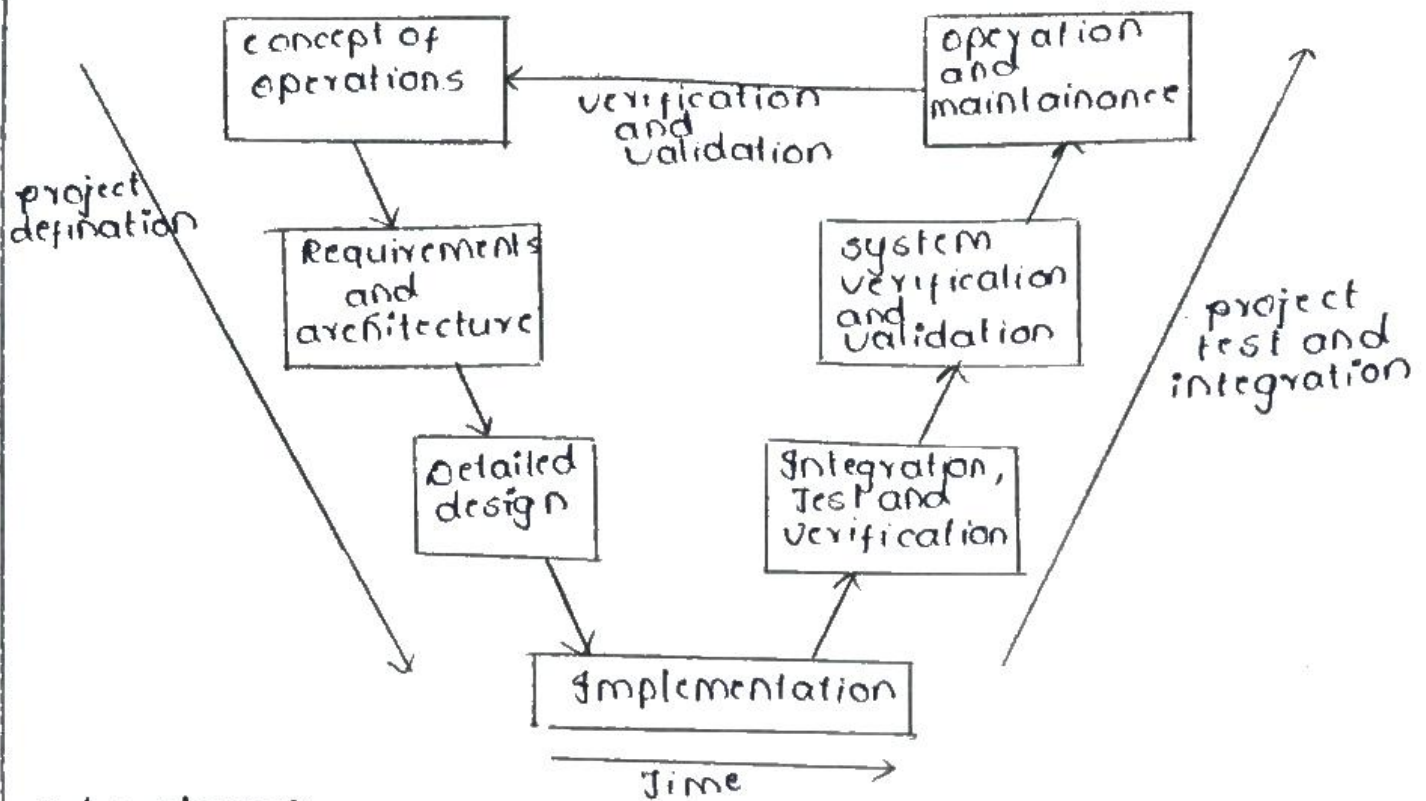
Disadvantages:

Because of its rigid structure Waterfall Model doesn't work well for complex projects.

2. V-Model:

→ An extension of Waterfall Model is V-Model, it also functions as sequential flow.

→ For every phase in downward sequence there is corresponding testing phase in the following upward sequence.



Advantages:

- V-Model is a simple process that is great for smaller projects
- V-Model can yield higher chance of success due to the test plans of development stage and regularly schedule updates through out its life cycle.

Disadvantage:

Similar to waterfall model V-model is very rigid in nature so it isn't ideal for applications that may require changes/updates through software lifecycle.

3. Incremental Model:

Incremental model consist of iterative and incremental development stages. The incremental model is essentially comprised of several mini waterfall cycles.

Advantage:

Incremental model is great solution for projects that need accomodation for some change request between increments.

Disadvantage:

Disadvantage of incremental model is need for strategic planning and documentation. This method also tends to require more resources, staff and monetary, behind the project.

4. RAD Model:

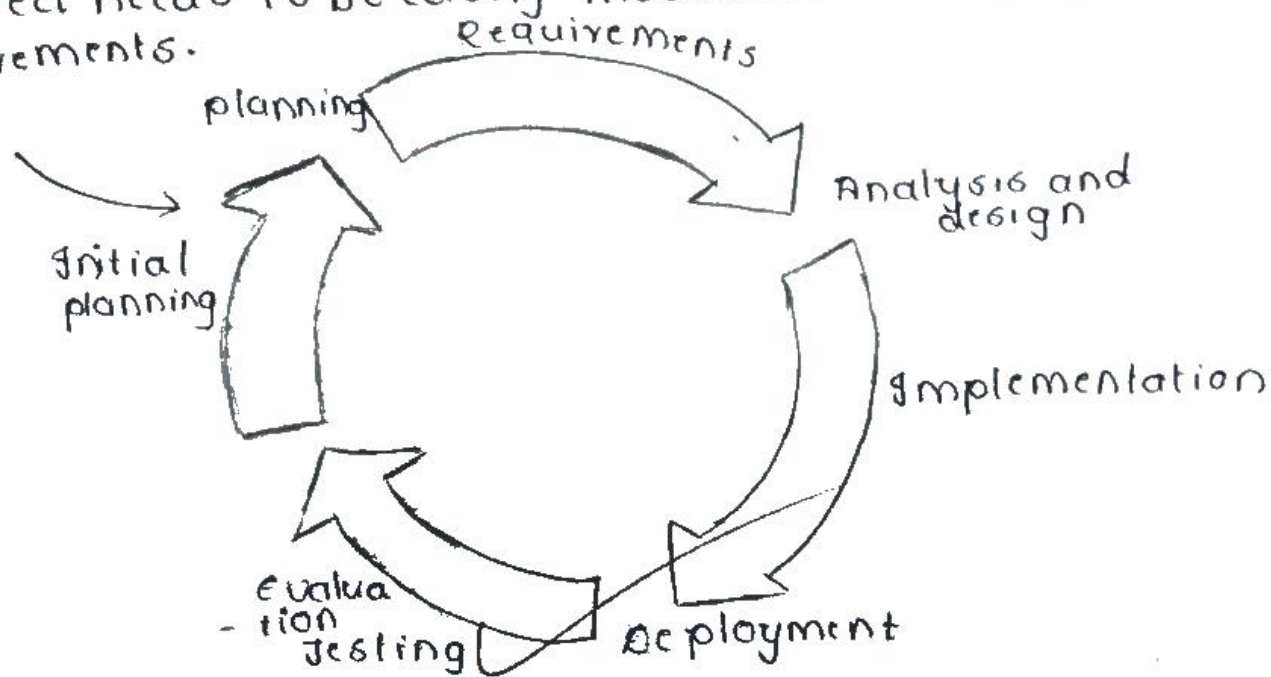
Rapid Application Development, RAD model is the modification of incremental model. The different components are assembled into working prototypes

Advantage:

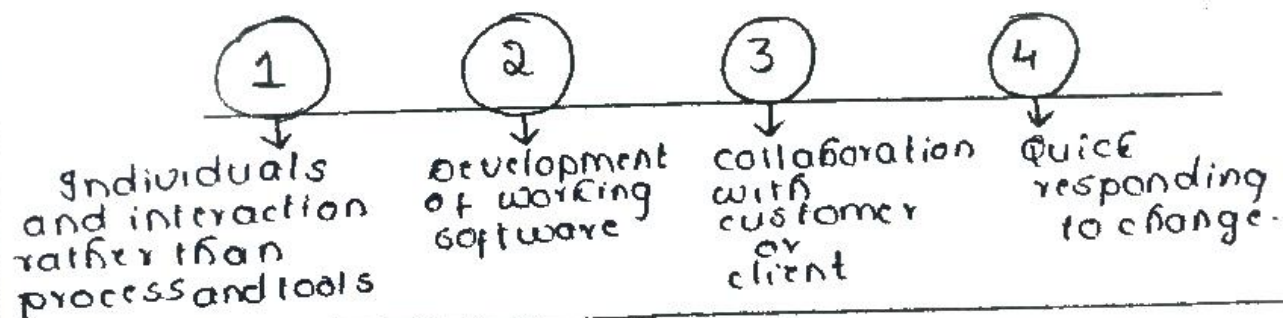
RAD model allows for reduced development time and allows for more customer feedback throughout software development.

Disadvantage:

The applicability of RAD model is limited as project needs to be easily modularized into several increments.



5. Agile Model:



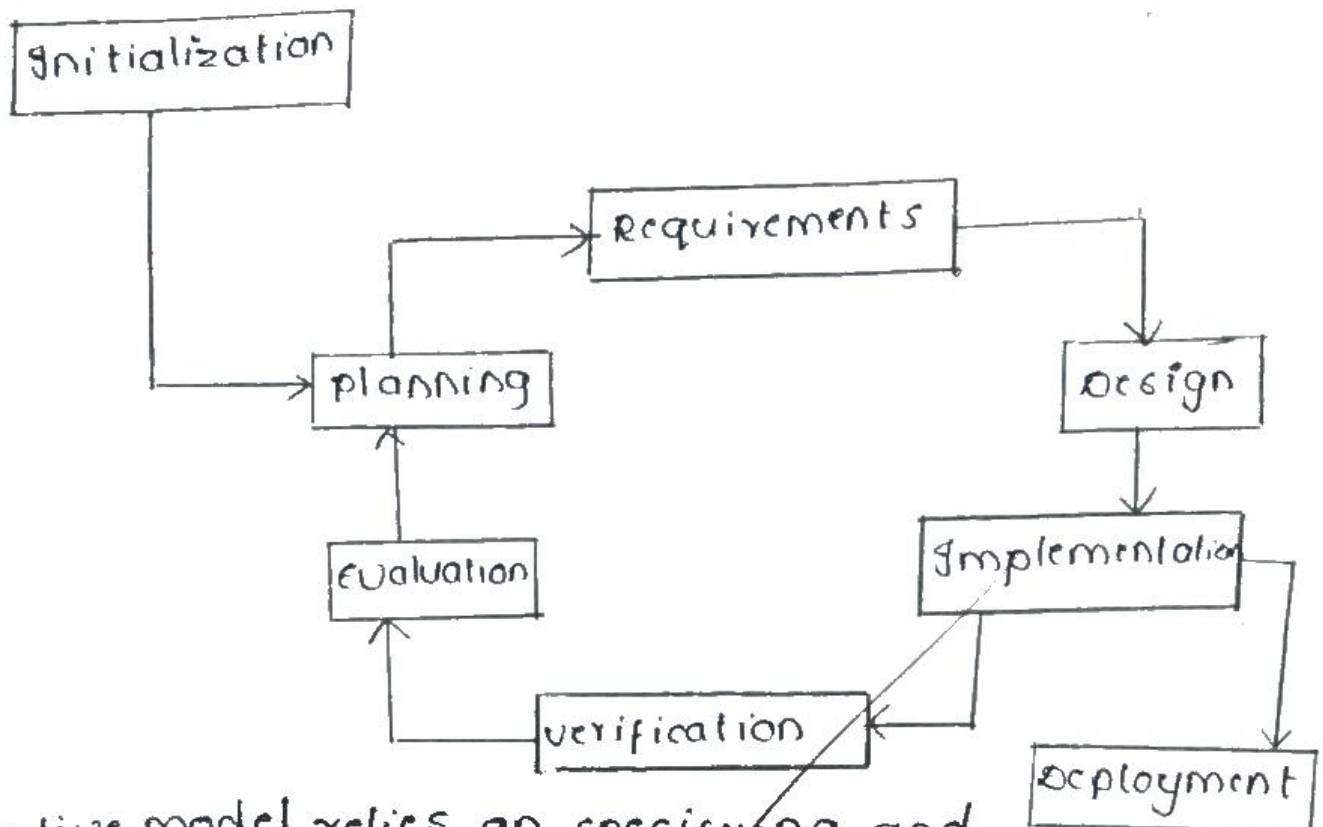
→ This model is rooted in process adaptability and user engagement with rapid delivery of functionality software components.

Advantages:

Agile model decreases amount of time to yield individual system features. It also calls for a lot of communication and continuous feedback.

Disadvantage: documentation is minimal.

6. Iterative model:



→ Iterative model relies on specifying and implementing individual parts of software rather than attempting to start with full specification requirements.

Advantage:

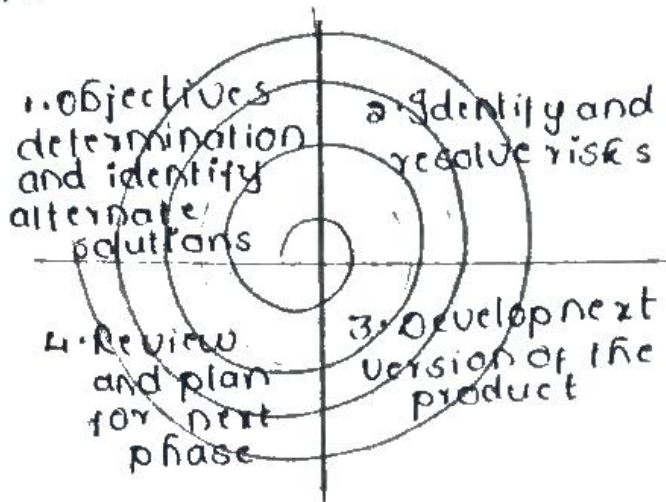
Since the product is developed gradually, it's easy to identify problems early when using this software development model.

Disadvantage:

Because each iteration phase is rigid with no overlaps, iterative model can take longer and be more costly.

7. Spiral Model:

Spiral model combines elements of both iterative and waterfall model. Spiral model has four phases: identification, design, construct/build, evaluation and risk analysis.



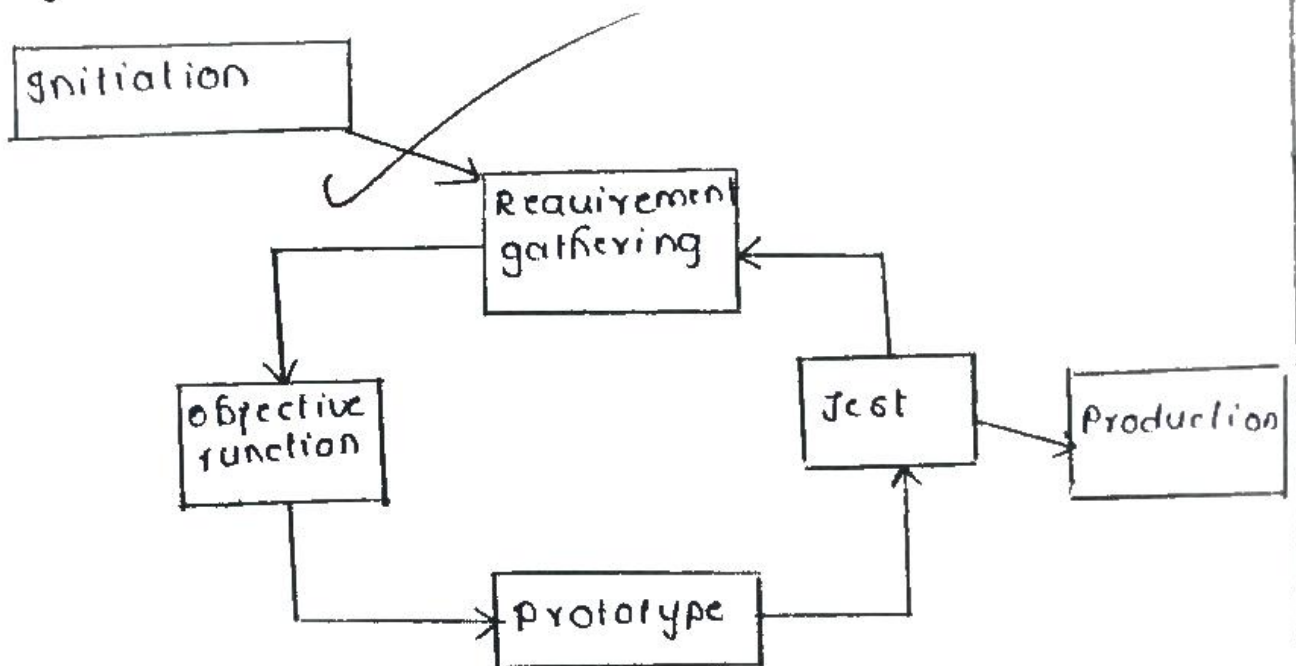
Advantages:

1. Risk handling
2. Good for large projects.
3. Flexibility in requirements.
4. Iterative and incremental approach.

Disadvantages:

1. Complex and expensive
2. Time consuming

8. Prototype Model:



prototype model relies on creating prototypes of software applications or system software that are used to visualize various components of software.

Advantage:

Reduce time and costs

Disadvantage:

Model can cause user confusion between prototype and finished product.

2. Explain branching work flow in GIT.

The core idea behind feature branch workflow is that all feature development should take place in a dedicate branch instead of main branch. This encapsulation makes it easy for multiple developers to work on particular feature without distributing main codebase.

Steps to create branches and merge them.

→ create a new folder named "pro" in d-drive

→ cd d:

→ cd pro

→ cd college

college is one of the folder in "pro" folder.

→ git init

initialized empty git repository in d:/pro/college/git/

→ vi main.html

→ git add main.html

→ d:/pro/college/master)

git commit -m "updated first file"

→ git checkout -b student

(we switched to another branch named "student")

→ (Sana @LAPTOP-6C2C978 MINGW64/d/pro/college (student))

→ vi student.html

(To write a HTML code in the editor)

→ git add student.html

→ git commit -m "updated second file"

- git checkout master
(switched to branch "master")
- git checkout -b faculty
(switched to new branch "faculty")
- sana @LAPTOP-6CC0978 MINGW64\ld\pro\college (faculty)
vi faculty.html
(to write a HTML code in the editor)
- git add faculty.html
- git commit -m "updated third file"
- git checkout master
(again switched to branch "master")
- git merge student
(merging one of the branch with main branch master)
- git log --oneline --graph --all
- * 9256a1d (faculty) updated third file
- | * aca4e80 (HEAD → master, student) updated second file
- | /
- * 989216c updated first file.