1. **Discuss the aspects where waterfall model is different from the prototype model.**

| Waterfall model | Prototype model |
|---|---|
| Linear and sequential. Each phase must be completed before moving to the next. | Iterative and cyclic. A prototype is built, tested, and refined based on user feedback. |
| Low – users are involved mainly at the beginning (requirements) and end (testing). | High – users continuously interact and provide feedback during prototype iterations. |
| Rigid – once a phase is finished, going back is difficult. | Flexible – easy to make changes based on feedback. |
| Suitable when requirements are clearly defined from the start. | Useful when requirements are unclear or evolving. |
| Less time in planning, but risky if requirements change. | More time and cost initially, but results in better user satisfaction. |
| Final product is delivered at the end of the project. | A working prototype is shown early and refined into the final product. |

2. **Explain the examples when incremental and iterative model of development is preferred.**

**Incremental model**

The system is developed in parts or increments. Each increment adds new features to the previous versions.

It is preferred when the project can be divided into independent modules and partial delivery is possible.

For example: online shopping system: first increment for login and registration, next for product catalogue then the payment gateway.

**Iterative model**

The system is developed through related cycles or iterations, improving the system each time based on feedback.

It is preferred when requirements are not fully known and need continuous refinement.

For example- game development, initial versions is tested by the users, then improved with graphics and features in each iteration.

In short, use incremental model for modular, structured projects needing progressive delivery and use iterative model for evolving projects requiring repeated improvement.

3. **Suppose you are a software engineer of your software development team. What would you recommend if you were to recommend the methodology for the following nature of projects with reasoning?**

i) First project is a news portal site that requires a member registration. Once the user becomes the member, he is allowed to read articles and make comments. He is pretty much certain of the design and has clear cut idea of the overall system that clients is looking for and the client needs it prepared in a month. At the same time, he is working on another project (details not necessary) simultaneously for which he has to attend frequent meetings with the client and has the deadline is within a few days.

ii) Another project is an e-learning platform that is very much different than other existing systems in Nepal. The client has necessary scripts of all the course contents but is reluctant whether the delivery of contents be it in an animated form or live video teaching form. There are other numerous details that he must

consider but cannot think of it right now. all he knows is he is determined to own an e-learning system.

**Answer**

**Project 1: news portal system**

**Nature**

- Requires membership registration.
- Client already knows the design and requirements.
- Clear deadline (within one month).

**Recommended model: waterfall model**

**Reasoning**

- Requirements are clearly defined and stable.
- Client already finalized the design and main features.
- Short project duration (1 month) suits a linear, fast-paced approach.
- No frequent feedback or iterations are needed.

Waterfall model ensures quick, structured development with minimal changes.

**Project 2: E-Learning System**

**Nature:**

- Client is unsure about the exact form of content (animated, recorded, live).
- Requirements are evolving.
- Needs user feedback to finalize.

**Recommended model: prototype model**

**Reasoning**

- Requirements are unclear and need refinement through examples.
- Prototype will help client visualize and decide on the final system.

- Iterative feedback ensures the final system meets client expectations.

Prototype model is ideal as it allows experimenting with different content formats before final development.

iii) A banking software system that must comply with a new financial regulation. All requirements are fixed, non-negotiable, and require strong documentation and an audit trail.

iv) A Civil Aviation Authority project for developing a new air traffic control system. The project is highly complex, mission-critical, and safety-sensitive, with major unknown technical risks.

**Project: Banking software system**

**Nature:**

- Must comply with a newly passed financial regulation.
- Requirements are fixed and non-negotiable.
- Strong need for documentation and audit trail.
- No change allowed once development starts.

**Recommended model: V-model (verification and validation model)**

**Reasoning**

- Requirements are clearly defined and must be strictly followed.
- Each development phase is directly linked to a testing phase, ensuring quality and compliance.
- High level of documentation fits legal and financial standards.
- Emphasizes validation and verification — essential for secure and compliant banking systems.

The V-Model ensures accuracy, traceability, and full compliance with regulations — perfect for banking and finance-related projects.

**Project: Air Traffic Control System (Civil Aviation Authority of Nepal)**

**Nature:**

- Highly complex, mission-critical system.
- High risk — failure could lead to catastrophic consequences.
- Involves unknown technical and safety challenges.
- Needs careful risk analysis and strict quality control.

**Recommended model: Spiral Model**

**Reasoning:**

- Spiral model combines iterative development with risk management.
- Each phase includes identification, analysis, and mitigation of potential risks.
- Allows continuous testing, prototyping, and validation at every stage.
- Ideal for large, safety-critical systems like air traffic control or defense systems.

The Spiral Model minimizes risks through early identification and testing while ensuring flexibility in managing complex technical requirements.