**Importance of timely delivery in software project management**.

Timely delivery is crucial because it:

* Ensures customer satisfaction.
* Prevents cost overruns due to extended timelines.
* Keeps the project competitive in fast-evolving markets.
* Avoids contract penalties or legal implications.

To ensure deadlines are met the following can be done;

* Clear planning, define milestones and deliverables early.
* Agile development, which uses iterative cycles for flexibility.
* Risk management by identifying potential roadblocks in advance.
* Resource allocation by ensuring the right team members are assigned tasks efficiently.
* Monitoring and communication, using project management tools like Jira or Trello for tracking progress.

**Importance of cost control in software**

Cost control is important in software projects as it;

1. Prevents financial losses and budget overruns.
2. Ensures project feasibility and sustainability.
3. Allows for resource allocation.

To prevent budget overruns the following best practices can be followed;

* Accurate budget forecasting by considering all costs including unforeseen ones.
* Agile budgeting through adjusting financial plans based on project phases.
* Regular expense tracking use tools like Microsoft project or Asana.
* Scope management, avoid scope creep by documenting clear requirements.
* Get cost-benefit analysis to evaluate features based on value addition.

**Differences between agile and waterfall methodologies.**

1. Agile is flexible as it accommodates any changes at any stage while waterfall flexibility is low as changes require revisiting previous stages.
2. Agile has a iterative development approach that evolves through sprints unlike waterfall which is sequential, each phase has to be completed before moving to the next.
3. Customer involvement in agile is high as there is frequent feedback loops while in waterfall there is low customer involvement as the client input is mainly at the start and end.
4. In agile there is continuous testing throughout the project while in waterfall testing happens after development.
5. Agile is best for dynamic and evolving projects while waterfall is essential for well-defined stable projects.

**Advantages and disadvantages of agile and waterfall;**

Agile is faster, flexible, and user-focused but can lead to scope creep.

Waterfall is structured and predictable but rigid to changes.

The following are phenomenon when to use either agile or waterfall;

Agile is best for startups SaaS products and apps requiring frequent updates. E.g. building a mobile app that requires frequent and continuous feedback.

Waterfall is best for government projects, infrastructure software or banking systems. E.g. building a payroll system.

**Quality assurance in software projects.**

The following are methods of ensuring quality assurance in projects;

* **Code reviews-** always review code to catch errors early.
* **Automated testing**- having automated testing is also essential for efficiency by using tools such as Selenium or Junit for efficiency.
* **Manual testing** – manual testing is also essential i.e. User acceptance testing (UAT) to ensure usability.
* **Continuous integration and deployment (CI/CD)-** having automated testing and integration of new code.
* **Version control-** using Git to track changes and rollback if necessary.

It is therefore important to have high standards as it;

* Reduces software failures.
* Enhances user experience.
* Improves security and compliance.

**Defining project scope and work breakdown structure (WBS)**

-Defining scope ensures alignment between stakeholder expectations and deliverables.

-Work Breakdown Structure (WBS) is a hierarchial breakdown of tasks;

1. It helps allocate responsibilities.

2. simplifies tracking and management.

3. prevent scope creep.

**Benefits of a detailed project schedule and Gantt Charts**

* Provides a clear roadmap.
* Enhances accountability.
* Helps allocate resources effectively.

**How Gantt Charts Assist:**

* Visualize task dependencies.
* Track progress over time.
* Identify potential bottlenecks.

**Core issues addressed by a software**

A learning management system that tracks students’ progress in the competency based curriculum and helping to identify learners’ strengths.

The LMS is significant as it tracks student progress ensuring stakeholders I.e. teachers, learners and parents understand the progress of the learner.

A well-defined problem ensures:

* Focused software development.
* Avoiding unnecessary features.
* Meeting end-user needs effectively.

How would you describe your software solution in a way that captures its essence without diving into technical details?

I’m going to describe the software without any technical jargon and keeping the presentation as simple as possible.

**Key features that make software stand out.**

* **Unique selling proposition (USP)-** what differentiates it from competitors.
* **Scalability –** can it grow with business needs?
* **User experience (UX).** Is it easy to use and interact with?

Market size and growth potential.

Market data helps in:

* Forecasting demand
* Securing funding
* Planning scalability

**Market trends for positioning software**

* AI/ML integration.
* Cloud- based solutions
* Cybersecurity enhancement
* Subscription – based pricing models.