# **SE\_DAY4\_Software-Project-Management**

### **1. Importance of Timely Delivery in Software Project Management**

Timely delivery ensures that the software meets market demands, maintains stakeholder trust, and avoids cost overruns. **Project managers can ensure deadlines are met by:** ✔️ Setting clear milestones and deadlines.  
 ✔️ Using Agile methodologies for flexibility.  
 ✔️ Monitoring progress with project management tools (e.g., Jira, Trello).

### **2. Effective Cost Control in Software Projects**

Cost control prevents financial losses and ensures resource efficiency. **Strategies to avoid budget overruns:** ✔️ Accurate cost estimation and budgeting.  
 ✔️ Continuous monitoring and reporting.  
 ✔️ Risk management to handle unforeseen expenses.

### **3. Agile vs. Waterfall Methodologies**

| **Feature** | **Agile** | **Waterfall** |
| --- | --- | --- |
| **Approach** | Iterative & flexible | Sequential & structured |
| **Changes** | Easily adaptable | Hard to modify once started |
| **Best For** | Dynamic, evolving projects | Well-defined, stable projects |

✔️ Agile is **faster, adaptive, and collaborative** but can lack structure.  
 ✔️ Waterfall is **clear, structured, and predictable** but less flexible.

### **4. When to Use Agile vs. Waterfall**

**Agile:** Best for software startups, mobile apps, or web development. *(Example: A fintech startup building a payment app)* **Waterfall:** Best for regulated industries like healthcare or construction software. *(Example: A hospital management system)*

### **5. Methods for Quality Assurance in Software Projects**

✔️ Automated testing (unit, integration, regression).  
 ✔️ Code reviews and continuous integration (CI/CD).  
 ✔️ User feedback and beta testing.  
 **Why?** High-quality software reduces bugs, enhances user satisfaction, and improves maintainability.

### **6. Defining Project Scope & Work Breakdown Structure (WBS)**

✔️ Clear project scope prevents scope creep and misalignment.  
 ✔️ **WBS** breaks projects into smaller, manageable tasks, improving planning and execution.

### **7. Benefits of a Detailed Project Schedule & Gantt Charts**

✔️ Keeps tasks on track and ensures accountability.  
 ✔️ **Gantt charts** visualize progress, dependencies, and deadlines, making project management easier.

### **8. Core Issues Addressed by Your Software**

✔️ Identify key pain points (e.g., inefficiency, high costs, poor user experience).  
 ✔️ Ensure alignment with user needs for maximum impact.

### **9. Importance of Clearly Defining the Problem**

✔️ Helps build a **focused** and **effective** solution.  
 ✔️ Prevents unnecessary features and resource wastage.

### **10. Describing Software Without Technical Jargon**

✔️ Focus on the **problem it solves** and **user benefits**.  
 ✔️ Example: *“Our app helps businesses automate invoices, saving time and reducing errors.”*

### **11. Key Features That Make Software Stand Out**

✔️ Unique selling points (e.g., AI-driven insights, seamless integrations, superior UX).

### **12. Market Size & Growth Potential**

✔️ Analyze demand, competitors, and industry trends.  
 ✔️ Use market research reports and customer surveys.

### **13. Using Market Trends for Positioning & Development**

✔️ Adapting to emerging technologies (AI, blockchain, cloud).  
 ✔️ Meeting evolving customer expectations (e.g., mobile-first solutions).