

Course: Software Project Management

Week 3: Common Software Project Problems

Lecturer: Yimer Amedie (MSc.)

Addis Ababa Science and Technology University, Ethiopia

Contents

- Introduction
- Overview on software project failure
- Software projects Vs Other projects
- Common problems, impacts and possible solutions

Learning Outcomes

After completing this lesson, you will be able to:

- Analyze real world project failures to extract lessons and avoid repeating mistakes.
- Identify and describe common software project problems, their causes and impacts
- Apply best practices to prevent or mitigate these problems.

Introduction

- Software development is a complex process that involves
 - multiple stakeholders
 - technologies and methodologies.
- Software projects are prone to various challenges that can hinder their successful completion.
- Regardless of advancements in tools and practices, many software projects still fail due to common problems.

Introduction

- Understanding common software project problems helps project managers and development teams
 - To identify risks and implement preventive measures
- This lesson covers
 - common software project problems
 - the causes of the problems
 - impacts, and possible solutions.

Overview on Software Project Failure

- Project Failure refers to the inability of a project to meet its objectives
 - such as delivering the intended results within the agreed-upon time, budget, and quality standards.
- It typically means that
 - the project either does not fulfill its purpose,
 - is significantly delayed, or
 - exceeds the allocated budget.

Overview on Software Project Failure

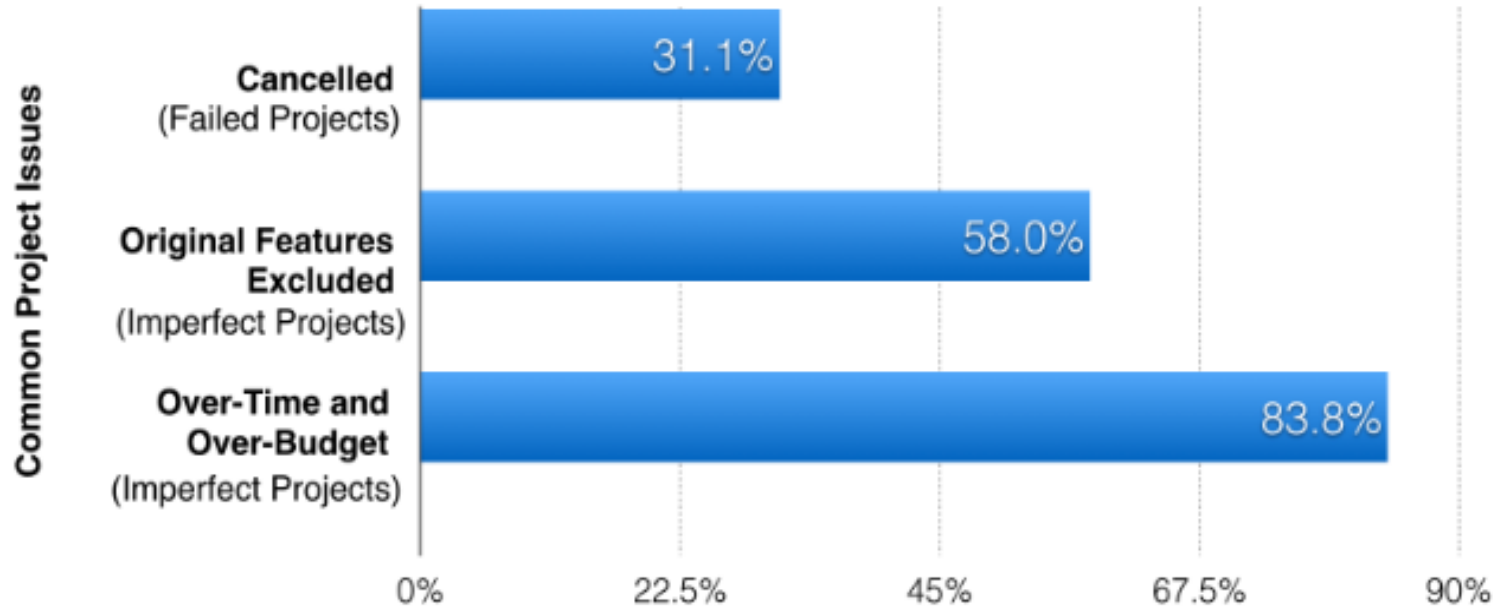


Figure 1: Overview of Project issues and failures (arc, 2017)

Overview on Software Project Failure

- According to the PMI
 - **11.4%** of projects fail completely (rejected).
 - **36%** of projects partially fail (delivered late/over budget with reduced features)
 - Software projects
 - Highest failure rate (42%) due to rapid tech changes.
 - Agile projects are 28% more successful than waterfall.

Overview on Software Project Failure

- Software Project Failure
 - a specific type of project failure that occurs when a software project does not achieve its planned outcomes.
- The failure may result in
 - unusable software
 - cost overruns
 - unmet user expectations, or
 - complete project rejection.

Overview on Software Project Failure

- Software projects are expensive due to it requires hardware, software and network for its implementation and operation.
- The failure rate is very high in software projects
 - **~66%** according to the (Standish Group, 2020)
 - **42%** complete failures according to (PMI, 2022)
 - This may be due
 - Technical, managerial, and human factors

Overview on Software Project Failure

- The cause may be



Due to inadequate project planning

We Need:



Due to unclear requirements



Poor project Management

Overview on Software Project Failure

Money, Technology Advancement



Software development becomes increasingly complex



Skilled Project Manager required



Ensure successful Implementation

Overview on Software Project Failure

- The characteristics of software projects make them particularly difficult and different from other projects.
 - **Invisibility**: progress in software project is not immediately visible.
 - **Complexity and dependency**: involves multiple interdependent components
 - **Conformity**: software system has to conform to the requirement of human clients.
 - **Flexibility**: easy to change.
 - **Collaboration & Remote Work**: Software teams often work remotely or across different time zones.

Overview on Software Project Failure

- The characteristics of software projects make them particularly difficult and different from other projects.
 - **High uncertainty and frequent change:** Requirements often evolve due to changing business needs or user feedback.
 - **Stakeholder Expectations & User Experience:** Users expect software to be intuitive, fast, and bug-free.
 - **Resource Management & Specialization:** Software projects require specialized skills

Common Software Project Problems

- Software projects faces several problems which leads it to failure.
- The following are the major problems

1. Poor Requirements

- Unclear, incomplete, or constantly changing requirements lead to misaligned expectations.
- **The causes are:**
 - Lack of stakeholder involvement
 - Assumptions instead of documented needs
 - Rapidly changing business needs (especially in Agile)

Common Software Project Problems

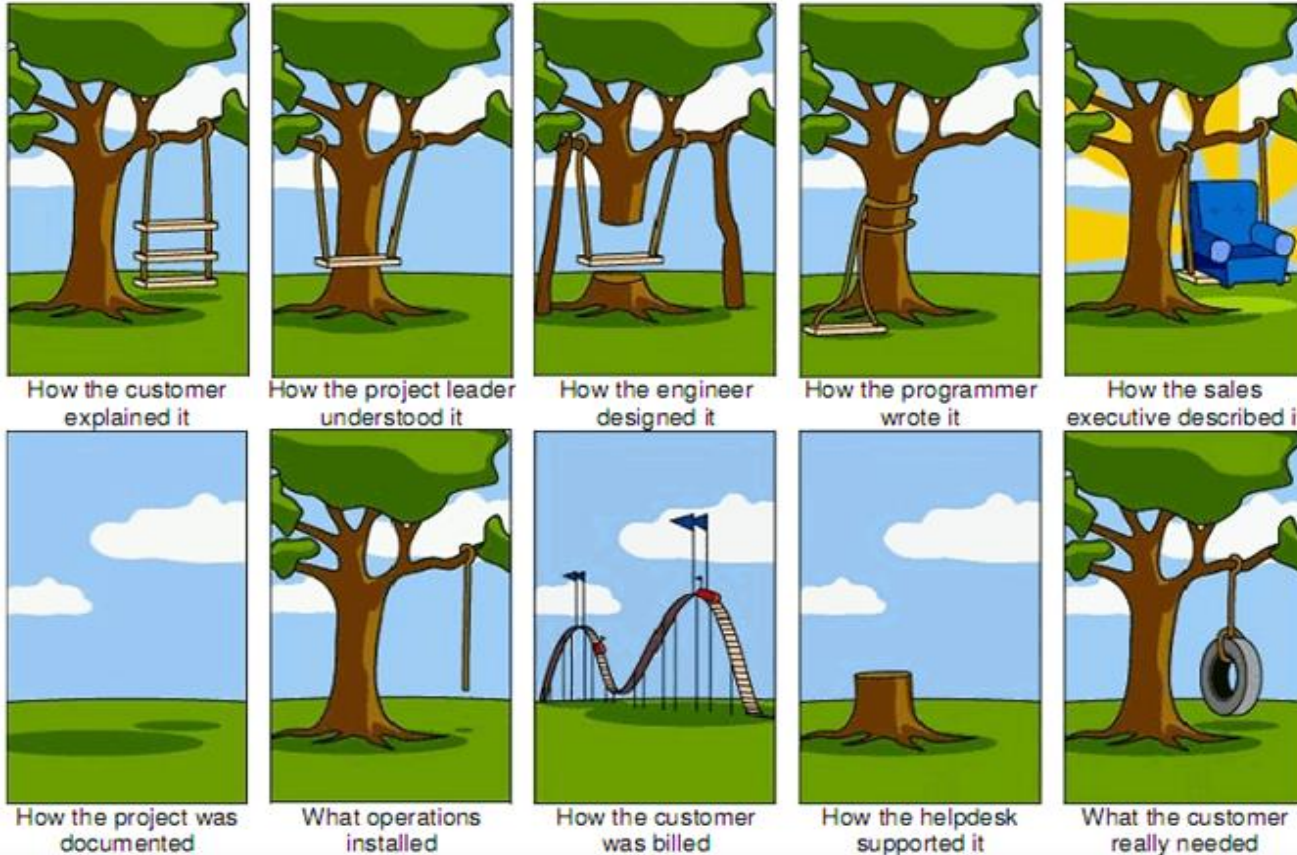


Figure 2: Mismatch between delivered software & user needs (arc, 2017)

Common Software Project Problems

2. Scope Creep

- Uncontrolled addition of features disrupts timelines.
- **The causes are:**
 - Lack of a formal change control process
 - Clients/stakeholders requesting
 - **"just one more thing"**
 - Poorly defined project scope

Common Software Project Problems

3. Unrealistic Deadline and Schedule

- Impractical timelines and deadlines lead to rushed work and poor quality.
- **The Causes:**
 - Pressure from management/clients
 - Inexperience in estimation techniques
 - Underestimating complexity

Common Software Project Problems

4. Poor Communication and Collaboration

- Lack of clear communication between team members and stakeholders.
- Miscommunication leads to errors and rework.
- **The Causes:**
 - Remote work challenges
 - Lack of structured meetings (e.g., no daily stand-ups)
 - Poor documentation

Common Software Project Problems

4. Poor Testing (Technical Debt or Poor Code Quality)

- Quick fixes lead to long-term maintenance.
- **The Causes:**
 - Tight deadlines forcing shortcuts, lack of code reviews & standards, ignoring refactoring

5. Resource Management Issue

- Wrong staffing leads to delays.
- **The Causes:**
 - Lack of skilled developers, overworked teams, poor task allocation

Common Software Project Problems

6. Poor Risk Management

- Failing to assess and address risks
- **The Causes:**
 - Lack of risk management plan

7. Security and Compliance Issue

- Failing to meet legal and security requirements.
- **The Causes:**
 - Lack of security audit
 - Weak authentication/input validation

Common Software Project Problems

8. Team Turnover and Loss of Key Personnel

- Losing critical team member
- **The Causes:**
 - Poor work environment, benefit, project instability

9. Dependency & Integration Challenges

- Third-party failures disrupt projects.
- **The Causes:**
 - Unstable APIs/libraries
 - Poorly managed microservices

Common Software Project Problems

10. Poor Maintenance & Documentation

- Unmaintainable code slows progress.
- **The Causes:**
 - Insufficient or outdated documentation
 - No documentation
 - Lack of guides
- The problems of software projects can be
 - General problems
 - Project Manager's specific problems

Common Software Project Problems

- In general the major causes of Software Project Failure are the following (PMI, 2022)
 - 38% → Poor requirements gathering.
 - 32% → Inaccurate cost/resource estimates.
 - 27% → Lack of executive sponsorship.

Common Software Project Problems

General problems:

- lack of knowledge of application area;
- lack of up-to-date documentation, commitment and training
- preceding activities not completed on time;
- lack of communication between users and technicians;
- narrow scope of technical expertise;
- changing statutory requirements;
- changing software environment;
- deadline pressure and lack of training

Common Software Project Problems

Commonly experienced problems (from the manager's point of view)

- poor estimates and plans;
- lack of quality standards and measures;
- lack of guidance about making organizational decisions;
- lack of techniques to make progress visible;
- poor role definition (who does what?)
- incorrect success criteria.

Common Software Project Problems

Example:

- Your team is developing a web application for a client. Midway through the project, the client requests additional features that were not part of the original scope. These features significantly impact the timeline and budget. As the project manager, how would you handle this situation?
 - Assess impact
 - Change request process
 - Communicate with stakeholder
 - Negotiate and decision making

Impacts and solutions

- The challenges and problems of software projects impacts the successful implementation of the project and may leads to
 - Out of deadline
 - Budget overrun
 - Poor quality and features
 - Total failure.
- Overrun is common for software projects, so that projects may completed with additional time and/or cost.
 - **Projects can be ideal, delayed or failed**

Impacts and solutions

- Using best practices to prevent or mitigate the problems is necessary to ensure long term project success by using
 - Agile methodology
 - Change control
 - Code review
 - Documentation
 - Automated testing and security auditing

Impacts and solutions

- More specifically software project failures can be handled with
 - Improving Requirements Engineering
 - Effective Scope Management
 - Realistic Scheduling & Budgeting
 - Enhanced Testing Strategies
 - Risk Management Frameworks
 - Reducing Technical Debt
 - Team & Skill Development

Impacts and solutions

- Therefore, proactively preventing software project problems is required for
 - Better control of financial, physical, and human resources
 - Improved customer relations
 - Shorter development times
 - Lower costs
 - Higher quality and increased reliability
 - Improved productivity and Better internal coordination

Common Software Project Problems

- **Example: Boeing 737 Max Software Failure (2018)**
 - **Problem:** Faulty flight control software (MCAS) led to two crashes, causing hundreds of loss of life and grounding the aircraft worldwide.
 - **Causes:**
 - Poor requirement
 - Inadequate testing and insufficient pilot training.
 - Lack of redundancy in safety-critical software.

Common Software Project Problems

- **Example: Healthcare.gov (2013)**
 - **Problem:** The website for the Affordable Care Act launched with severe performance issues, frequent crashes, and security vulnerabilities.
 - **Causes:**
 - Lack of clear requirements and poor planning.
 - Ineffective coordination among multiple contractors.
 - Rushed implementation without proper testing.

Common Software Project Problems

- **Example: Lessons Learned**
 - Perform thorough testing before deployment.
 - Provide adequate training and documentation for end-users.
 - Ensure proper coordination between teams.
 - Break down large projects into manageable phases.

Summary

- Addressing common software project problems requires proactive planning, effective communication, and diligent risk management.
 - **Clear requirements** prevent misalignment.
 - **Realistic planning** avoids burnout.
 - **Scope control** keeps projects on track.
 - **Good communication** reduces errors.
 - **Managing technical debt** ensures long-term quality.

Summary

- **Automated testing** catches bugs early.
- **Proper resource allocation** prevents delays.
- **Dependency management** avoids integration issues.
- **Security-first approach** protects applications.
- **Good documentation** eases maintenance.
- By understanding these problems and applying best practices, software teams can **increase success rates** and **deliver high-quality products efficiently**.

References

1. PMI. (2023). *Pulse of the profession 2023: Powering change*. Project Management Institute. Retrieved from <https://www.pmi.org/>
2. PMI. (2022). *Why projects fail: Key insights and lessons learned*. Project Management Institute. Retrieved from <https://www.pmi.org/>
3. Arc. (2017, February 22). *10 Reasons Why Software Projects Succeed*. Retrieved March 31, 2025, from Codementor: <https://www.codementor.io/@codementorx/software-projects-failure-rate-success-factors-5hsxodqru>
4. Standish Group. (2020). *CHAOS Report 2020: Project success and failure rates*. Standish Group International. Retrieved from <https://www.standishgroup.com/>