Course: Software Project Management

Week 2: Software Project Constraints and Trade-Offs

Lecturer: Yimer Amedie (MSc.)

Addis Ababa Science and Technology University, Ethiopia

Contents

- Introduction
- Characteristics of a project
- Constraints of a software project
- Management of scope, time and cost in software projects
- Software project trade offs
- The strategies to manage software project constraints and trade offs

Learning Outcomes

After completing this lesson, you will be able to:

- Describe a key characteristics of a software project
- Identify and explain the software project constraints
- Understand the triple constraint triangle of a project management
- Recognize key management areas including scope, time and cost
- Analyze trade offs and decision making strategies.

Introduction

- Software projects involves SDLC and are complex which requires careful planning, execution and management.
- The question of what makes a project successful is an important issue.
  - Project management success **Vs** project success (Caccamese, 2012).
- A software project is considered to be successful, where the **scope** is achieved within the given **time**, **budget**, and **resources**, without compromising **quality** while proactively managing **risks**.

Success = Balancing Constraints Effectively

Introduction

- Therefore, considering the constraints and trade offs are very critical in managing projects.
  - **Constraints**: the limiting factors within which the project should operate
  - The **trade offs** involves choosing the best to balance competing demands. So, one aspect of a project can be affected to gain advantage with another.
- The constraints and trade offs are influenced by the project characteristics. ➔ how well constraints are managed?

Characteristics of a software Project

- The defining attributes or essential features that distinguish a project.

- **Uniqueness:** Each software project is distinct, with specific goals, deliverables, and challenges
- **Temporary Nature:** Each software project has a defined start & end date, unlike ongoing operations.
- **Clear Objectives:** A software project aim to achieve specific goals or produce unique deliverables.
- **Progressive Elaboration:** Details evolve as the project progresses.

Characteristics of a Project

- **Resource Constraints:** Limited time, budget, manpower, and materials affect decision-making.
- **Stakeholder Involvement:** Different stakeholders (clients, teams, users) influence the project.
- **Risk & Uncertainty:** Software project carries potential risks that must be managed.
- **Cross-functional Teams:** Software projects often require collaboration between various departments and expertise.

Characteristics of a Project

- Generally, the following characteristics distinguish projects (Bob Hughes, 2011)
  - non-routine tasks are involved
  - planning is required
  - specific objectives are to be met or a specified product is to be created
  - the project has a predetermined time span
  - work is carried out for someone

Characteristics of a Project

- Generally, the following characteristics distinguish projects (Bob Hughes, 2011)
  - work involves several specializations
  - people are formed into a temporary work group to carry out the task
  - work is carried out in several phases
  - the resources that are available for use on the project are constrained
  - the project is large or complex.

Constraints of a Software Project

- A project is constrained by cost, scope and time. Software projects are also constrained by these triple constraints of a project management
  - **Scope:** A product, service or result that the project will produce
  - **Time:** Amount of time required to complete all activities of the project
  - **Cost:** Money, material or personnel to complete the project

Used as a measure of project success

Constraints of a Software Project

- These three elements are **interdependent**, and changes to one will often impact the others. For example
  - Tight deadline may require reducing scope or increasing cost.

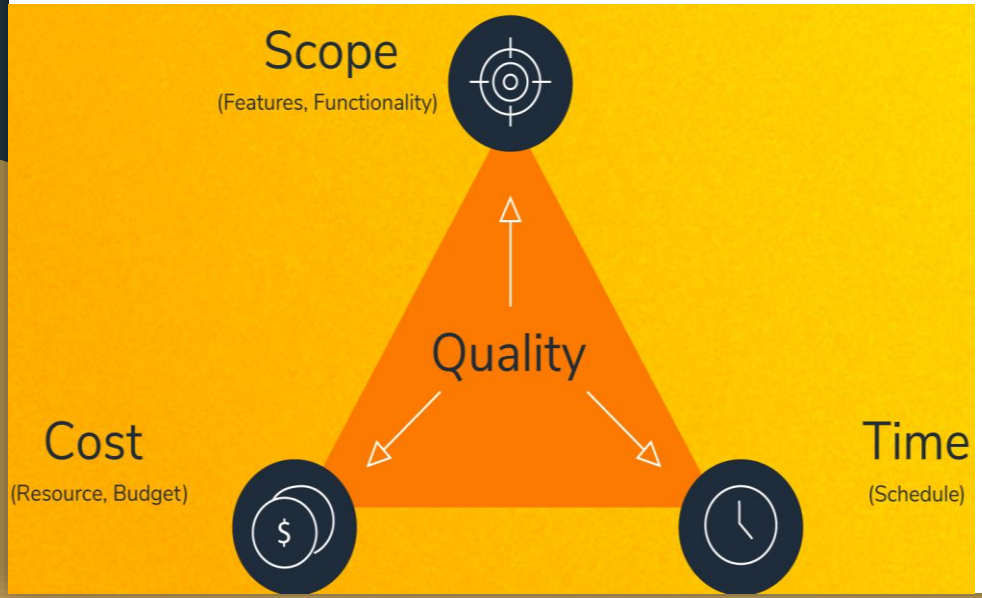

Figure 1: the triple constraints in project management (Żurawiecki, 2023)

Constraints of a Software Project

- So, considering and balancing these three competing goals is required to ensure project success ➔by Project Manager
- Project constraints can be categorized as:
  - **Inflexible / fixed:** must be forced to avoid negative impact on the project
  - **Flexible / negotiable**: can be modifiable to accommodate changes
  - Example: Time may be flexible, while scope & cost may be inflexible

Constraints of a Software Project

- Software projects often face additional constraints that impact success.
  - **Quality:** How good should it be? poor quality is a failure
  - **Resources:** who and what are needed? People, technology
  - **Risk:** what could go wrong?, potential obstacles
  - **Customer Satisfaction:** are users happy with the product?
  - **Compliance & Legal:** Does the project meet required standards? Is the project legally sound?

Constraints of a Software Project

Activity:

- A university is developing an **Online Learning System** to enhance remote education. The project has strict constraints in terms of scope, time, cost, quality, resources, and risk. The university has allocated **6 months** and a budget of **$200,000** for development. The system must include **course management, student enrollment, video streaming, and assignment submission** features. The project team consists of **5 developers, 1 UI/UX designer, and 1 project manager.**

Constraints of a Software Project

- **During the development phase:**
  - The university requests an additional **real-time chat feature** without extending the deadline or budget. Unexpected **server downtime** causes a **2-week delay** in deployment. Initial testing reveals **poor video streaming quality**, requiring rework.
- **Questions:**
  - How should the project manager handle additional features while maintaining time and cost constraints?
  - What strategies can be used to recover from the 2-week delay caused by server downtime?
  - What measures can be taken to improve video streaming quality without significantly increasing costs?

Constraints of a Software Project

- **Answer:**
  - The project manager can negotiate with stakeholders to defer the real-time chat feature to a future phase or remove a less critical feature to balance scope.
  - The team can adopt parallel development for pending tasks, use overtime (if budget allows), or prioritize critical features to meet the deadline.
  - Optimize video compression techniques, use a better CDN, and conduct thorough performance testing to enhance video streaming.

Managing Scope, Time & Cost in Software Projects

- Although the additional constraints mentioned above impacts the project success, effective **management** of the triple constraints / iron triangle (which are primary) remains essential for balancing them and achieving project success. In the following few slides
  - **Scope management, Time management** and **Cost management** will be discussed

Scope Management

- Before the actual project work begin, it is important to confirm that the scope of the project is through and detailed.
- The **scope** includes features, functionalities and deliverables.
- The scope is the critical element or constraint of a **software project.**
  - Functional requirements (features, user stories)
  - non functional requirements (security, performance)
- Properly managing it enable the team to complete the project successfully.

Scope Management

- Scope management
  - A process of defining, controlling, and managing the work required to deliver a project successfully. **What is and is not included in the project**
  - It ensures that the project includes all the necessary tasks and
  - only those tasks required to complete the project objectives.
- Example: Project ➔ **[Online Learning System]**
  - registration, course enrollment, video streaming 🎯➔ **Scope – in**
  - Payment Processing, live classes ❌➔ **Scope – out**

Scope Management

- Effective scope management
  - helps prevent scope creep
    - uncontrolled expansion of project scope / size
  - ensures the project stays on track in terms of time, cost, & quality
- The project scope management process includes (PMI,2013)
  1. Plan scope management
  2. Collect requirement
  3. Define scope
  4. Create WBS
  5. Validate Scope
  6. Control Scope

Scope Management

- **Example ➔ [Online Learning System], the steps**
  - **Collect requirements:** ➔students, instructors, administrators
    - Students need user registration & course enrollment features, instructors need a grading system
  - **Define scope:** registration, course enrollment, and video streaming
    - Exclude payment system, live classes from the current phase
  - **Create WBS:** database design, backend development, and frontend integration. Refer lecture 6.

Scope Management

- **Example ➔ [Online Learning System], continued**
  - **Validate Scope:** ➔ Present the scope document to stakeholders and obtain their approval.
    - Conduct a scope review meeting to ensure all stakeholders agree on the deliverables
  - **Control Scope:** Monitor & control scope changes throughout the project
    - Use a change control process to evaluate
    - approve any requests for additional features

Scope Management

- Why scope management is important?
  - It ensures that the project team and stakeholders have a clear understanding of what is included (and excluded) in the project.
  - It helps prevent scope creep, which can lead to delays, cost overruns, and compromised quality.
  - It provides a baseline for measuring project progress and success.
  - It aligns the project with the organization's goals and stakeholder expectations.

Time Management

- Most of software projects require additional time at the end of the project because of
  - Delayed activities at the beginning.
  - Some activities take more time than expected
- Time management is a critical aspect of software project management
  - It involves planning, scheduling, and controlling the time required to complete project tasks and deliverables.

Time Management

- In software projects, it involves the following key components:
  1. **Activity Definition**
     - Identify and define the tasks required to complete the project.
     - **Example:** For an **Online Learning System**, tasks are database design, backend development, and frontend integration.
  2. **Activity Sequencing**
     - Determine the order in which tasks should be executed based on dependencies.
     - **Example:** Backend development cannot start until the database design is completed.

Time Management

3. **Activity Duration Estimation**
   - Estimate the time required to complete each task.
   - **Example:** Database design may take 2 weeks, backend development 4 weeks, and frontend integration 3 weeks.
4. **Schedule Development**
   - Create a project schedule that outlines the start and end dates for each task.
5. **Schedule Control:** Monitor and control the project schedule to ensure it stays on track.

Time Management

- Why time management is important?
  - It ensures that the project is completed within the agreed timeline.
  - It helps allocate resources efficiently and avoid bottlenecks.
  - It provides a clear roadmap for the team to follow.
  - It enables early identification of potential delays and allows for timely corrective actions.
  - It aligns the project with stakeholder expectations, and business goals.

Cost Management

- Cost is the life blood of any projects.
- Software projects are considered to be highly expensive and needs proper cost management.
- Cost management ensures the project stays within the approved budget. It includes:
  - **Cost Estimation:** Predicting the expenses required.
  - **Cost Budgeting:** Allocating funds to different tasks.
  - **Cost Control:** Tracking expenses and preventing budget overruns.

Cost Management

- **Example:**
  - If the university sets a fixed **budget of $200,000**. what is the role of cost management?
  - Cost management should ensure
    - Development, Testing, and
    - training expenses stay within that amount.
  - What if costs rise unexpectedly?
    - **The team must either reduce scope or extend the timeline** to avoid budget overruns

Cost Management

- Therefore, cost management is important, because it
  - Prevents overspending, ensuring financial sustainability.
  - Helps in prioritizing resources, focusing on what’s most important.
  - Ensures that stakeholders approve financial decisions before spending.
  - Supports better contract management with vendors and service providers.

Trade offs in Software Project

- Trade-offs are expected in software projects, as it is rarely possible to optimize all constraints simultaneously. The project managers must make wise decisions to balance competing demands. The tradeoffs are:
  - **Scope vs. Time**:
    - If the project falls behind schedule, the team may decide to launch with only core features and add advanced features in a later update.
    - **Trade-Off:** Reducing scope to meet the deadline.

Trade offs in Software Project

- **Cost vs. Quality**:
  - To ensure high-quality video streaming, the team may invest in premium hosting services, increasing costs.
  - **Trade-Off:** Higher quality at the expense of increased costs.
- **Time vs. Resources**:
  - To accelerate development, the team may hire additional developers, increasing costs but saving time.
  - **Trade-Off:** Increased resources to meet the deadline.

Trade offs in Software Project

- **Quality vs. Time**:
  - Rushing to meet the deadline may result in fewer QA tests, leading to potential bugs in the system.
  - **Trade-Off:** Sacrificing quality to meet the timeline.
- **Risk vs. Cost**:
  - The team may allocate budget for backup servers to mitigate the risk of system downtime during peak usage.
  - **Trade-Off:** Increased cost to reduce risk.

Trade offs in Software Project

- In most of the projects, scope and cost is fixed, while time is flexible.

What if, **time** is also fixed?

- If **scope, cost, and time** are all fixed, quality and risk management will suffer.
  - making it almost impossible to meet all constraints without trade-offs.
  - In such cases:
    - **Negotiate with stakeholders** to adjust one of the constraints.
    - **Prepare for potential risks**: missing features, technical debt, post-launch bug fixes

Managing constraints & Trade offs

- The importance of each constraint depends on the type of project, stakeholders' priorities, and business goals.
- To manage constraints and trade offs in software projects, **constraints prioritization** is the best strategy.
- The steps to prioritize constraints:
  1. **Identify primary constraints based on project type**
  2. **Categorize constraints as fixed, flexible or negotiable**
  3. **Define trade offs based on priorities**
  4. **Monitor constraints through the project**

Managing constraints & Trade offs

1. **Identify primary constraints based on project type**
    - Each project has a **key constraint**. For an **online learning system**,
        - **Scope Priority:** If it needs specific features (like video streaming)
        - **Time Priority:** If the system must be ready before the next semester.
        - **Cost Priority:** If the university has a fixed budget.

Example:

✓ If the university has a strict deadline → Time is the highest priority.

✓ If wants advanced AI features → Scope is prioritized over cost.

✓ If limited fund → Cost is the main constraint, scope/time must adjust.

Managing constraints & Trade offs

2. Categorize constraints as fixed, flexible or negotiable

Constraint	Fixed (Non-Negotiable)	Flexible (Can Adjust)	Negotiable (Can Be Compromised)
Scope	Assignments, video streaming	AI tutor, discussion forums	Custom themes
Time	Ready before the semester	Testing phase duration	Feature rollouts
Cost	$200,000 budget cap	Extra $10k for urgent needs	Premium support costs
Quality	No major security issues, smooth UX	UI enhancements	Advanced animations
Resources	Limited dev team	Can hire one extra dev	Extra designers if needed
Risk	Data security compliance	Backup recovery plans	Marketing risk

If **time is fixed**, but cost and scope are flexible, the project may launch with an MVP first and expand later.

Managing constraints & Trade offs

3. **Define trade offs based on priorities**

- Trade-offs must be made when constraints conflict.
- If the university **insists on launching in 6 months**, AI tutoring might be **postponed for later phases** instead of delaying the project.

If You Prioritize:	You May Need to Sacrifice:
Scope (High Features)	More time & higher cost
Time (Fast Delivery)	Reduced scope & higher cost
Cost (Fixed Budget)	Reduced scope & longer timeline
Quality (High Standards)	More time & cost

Managing constraints & Trade offs

4. **Monitor constraints through the project**
   - Since software projects evolve, **constraint priorities may change** over time.
     - **Early Phase: Scope** is important (planning features).
     - **Mid-Phase: Cost** is critical (budget tracking).
     - **Final Phase: Time** becomes top priority (meeting deadlines).

Example:

✓ In the middle of the project, if the university secures additional funding …?

➢ The budget constraint relaxes, allowing for extra features or hiring more developers to speed up delivery.

Managing constraints & Trade offs

- Generally,
  - **If time is the top priority:** Focus on MVP first, roll out extra features later.
  - **If cost is the main concern:** Use open source, avoid custom features.
  - **If scope is critical:** Allow flexible time or increase budget for development.
  - **If quality is vital:** Extend testing time, even delaying the release.
- Other strategies
  - **Agile or iterative development / flexibility**
  - **Stakeholder communications**

Managing constraints & Trade offs

- **Activity**
  - You are developing an **Online Learning System** for a university to support remote education. The project has a six month timeline & a $200M budget.
    - If the project ***timeline changes*** and you need to develop the software ***faster***, what are your options?
      A. The time constraint changing will have no impact on the budget or scope
      B. Increase the budget or decrease the quality
      C. Decrease the budget or increase the quality

Summary

- In software project, managing constraints and trade-offs is a crucial component.
- However, it is also a challenging activity for the project teams to manage the complexity and to deliver successful outcome.
- It requires
  - Comprehending of the key constraints
  - Making wise tradeoffs and
  - Using effective approaches

Summary

- Effective scope, time and cost management ensures that projects are completed on schedule, within budget, to the required quality standards.
- Regardless of the constraints,
  - flexibility, communication and a focus on priorities are keys for achieving project success.
- The characteristics of a project, such as its uniqueness, temporary nature and stakeholder involvement have a major impact in determining its constraints and trade-offs.

References

1. PMI. (2013). *A Guide to the Project Management Body of Knowledge* (5th ed.). PMI, Inc.
2. Bob Hughes, M. C. (2011). Software Project Management (5th ed.). McGraw-Hill.
3. Caccamese, A. & Bragantini, D. (2012). Beyond the iron triangle: year zero. Paper presented at PMI® Global Congress 2012—EMEA, Marsailles, France. Newtown Square, PA: Project Management Institute.
4. Żurawiecki, J. (2023, July 12). Navigating the triple constraints in project management. Retrieved March 21, 2025, from BigPicture: https://bigpicture.one/blog/triple-constraints-in-project-management/