

Course: Software Project Management

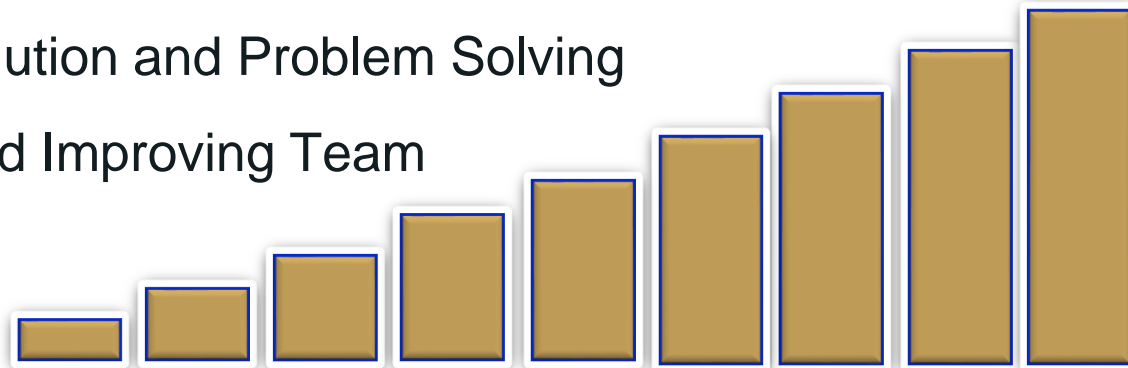
Week 12: Team Management

Lecturer: Yimer Amedie (MSc.)

Addis Ababa Science and Technology University, Ethiopia

Contents

- Introduction
- Team Formation and Development
- Team Organization and Structures
- Team Roles and Responsibilities
- Tradeoffs and enablers
- Conflict Resolution and Problem Solving
- Measuring and Improving Team Performance



Learning Outcomes

After completing this lesson, you will be able to:

- Define team management in software projects
- Identify team formation stages
- Describe structures, roles and motivation
- Apply communication and conflict resolution techniques
- Measure team performance and effectiveness

Introduction

What is team in software project?

- Team is a group of individuals with different skills and roles who collaborate to **plan, develop, test,** and **deliver** a **software product**.
- The structure and size of the team can vary depending on the scope and complexity of the project.



Introduction

- Software development projects are team-based efforts.
- It is not possible to deliver a **high-quality software product** without a team.
- However, simply forming a team is not enough, should be

Well-organized

Take responsibility

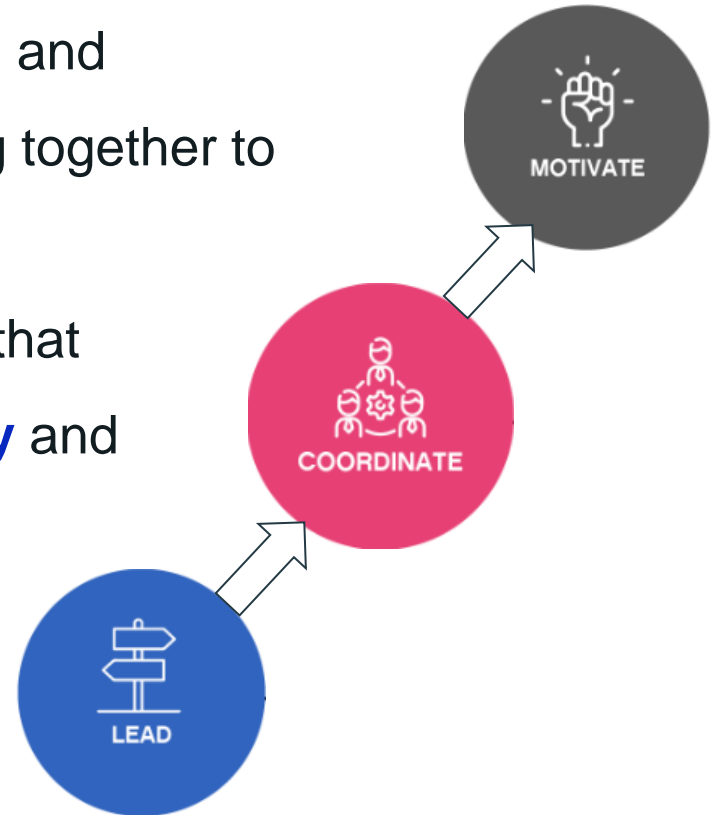
Share Common goal

For both individual tasks and the overall success of the project



What is Project Team Management?

- It is the process of **organizing, leading**, and **coordinating** a group of people working together to achieve the project goal.
- It involves creating a work environment that promotes **collaboration, accountability** and **Productivity**

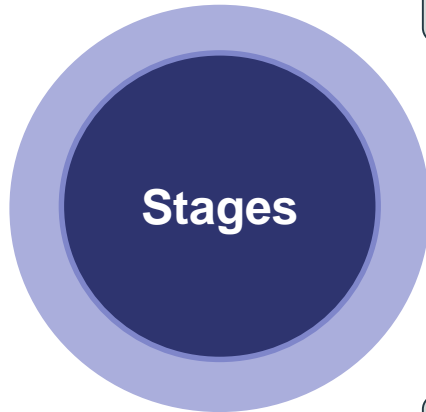


What is Project Team Management?

- Effective team management is critical to the success of software projects.
- It involves balancing
 - Technical oversight
 - People leadership
 - Process coordination to ensure timely delivery, high quality, and team satisfaction.

Team Formation and Development

- To enable people to work together as a team, it is suggested that teams go through five basic stages of development (Bob Hughes, 2011) .



**Tuckman's
model stages**

Team Formation and Development

Forming



- Team members are introduced.
- Roles and responsibilities are unclear.
- Focus is on orientation and learning about the project.
- Leader plays a dominant role.

Team Formation and Development

Storming



- Conflicts may arise due to differences in opinions or working styles.
- Team members push against boundaries.
- Leadership is challenged.
- Important to manage conflict constructively.

Team Formation and Development

Norming



- Team starts resolving conflicts and developing trust.
- Roles and responsibilities become clear.
- Team begins to collaborate effectively.
- Leader's role shifts toward facilitation.

Team Formation and Development

Performing

- Team functions as a cohesive unit.
- High performance, autonomy, collaboration
- Problem-solving, decision-making improve
- The team focuses on achieving goals.



Team Formation and Development

Adjourning



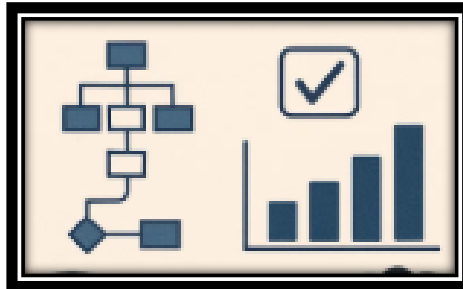
Project is completed



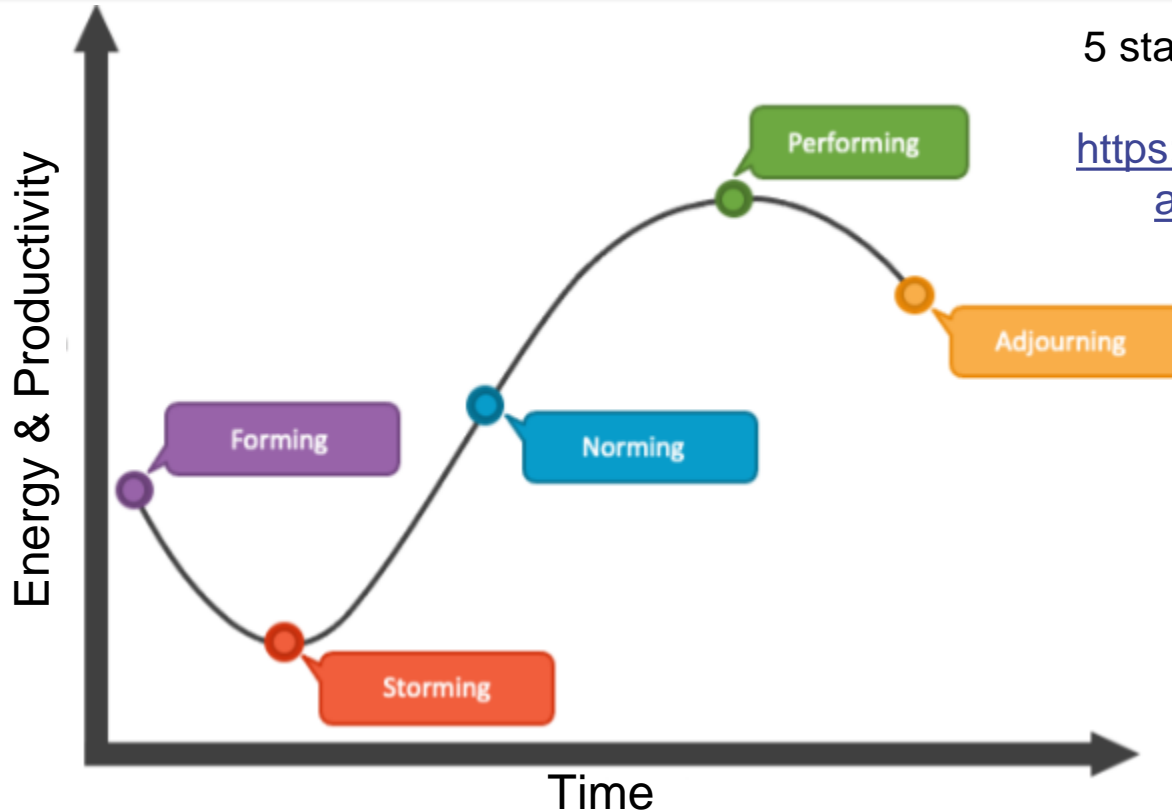
Team disbands



Important for members to reflect and celebrate accomplishments.



Team Formation and Development



5 stages of team development.
(source:
<https://www.collidu.com/presentation-5-stages-of-team-development>)

Team Formation and Development

Activity: Match the stages with their description

Stages		Descriptions
1. Forming		A. The team begins to build trust and collaborate.
2. Storming		B. Conflicts arise as individuals assert their ideas.
3. Norming		C. Team members get to know each other and define roles.
4. Performing		D. The project ends, and the team disbands
5. Adjourning		E. The team works effectively with high autonomy.

Team Formation and Development

Tools and Techniques for team development



Training

Providing relevant training programs to enhance skills.

1



Team building

Organizing activities to foster collaboration & communication.

2



Reward system

Implementing a system to acknowledge and appreciate contributions.

3

Team Organization and Structure

- Team organization refers to the structured way in which project members are
 - Grouped
 - Coordinated and
 - Assigned responsibilities to effectively deliver software products.
- The goal is to align people, skills, and workflows in a way that maximizes productivity, collaboration, and project success.



Team Organization and Structure

Factors affecting team organization

(Bob Hughes, 2011)

1. Department Structure

- How is a department organized into teams?
- Formats
 - Functional
 - Project
 - Matrix formats



Functional

Departments are grouped by areas of expertise.



Project

Teams are created for specific projects.



Matrix

Employees report to both functional and project managers.

Team Organization and Structure

Factors affecting team organization

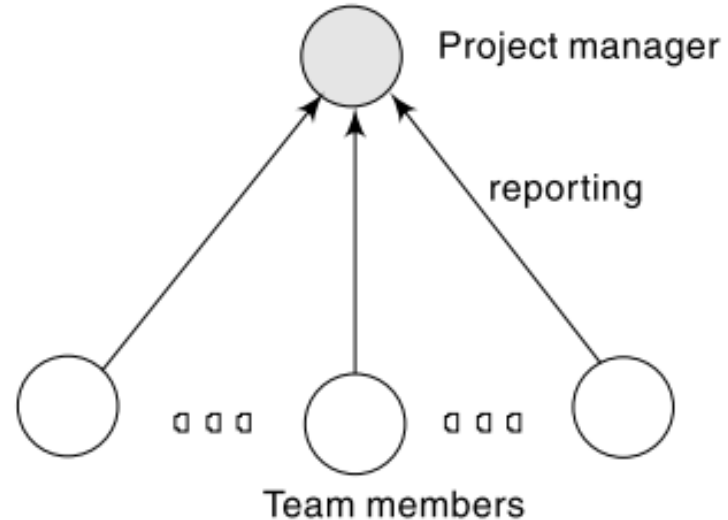
2. Team Structure

- Represents the team/individual reporting, responsibility, and communication structure.
- How are project teams structured?
- Teams
 - Chief programmer
 - Democratic
 - Mixed control

Team Organization and Structure

Chief Programmer Team

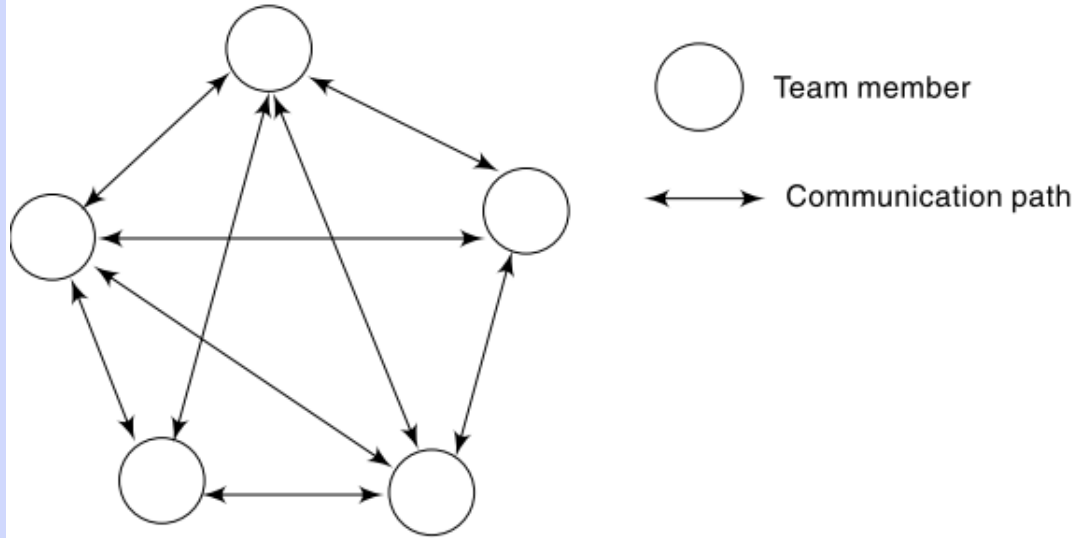
- It emphasizes having a small, highly skilled team led by a chief programmer
- It is structured to improve productivity and maintain high-quality output.



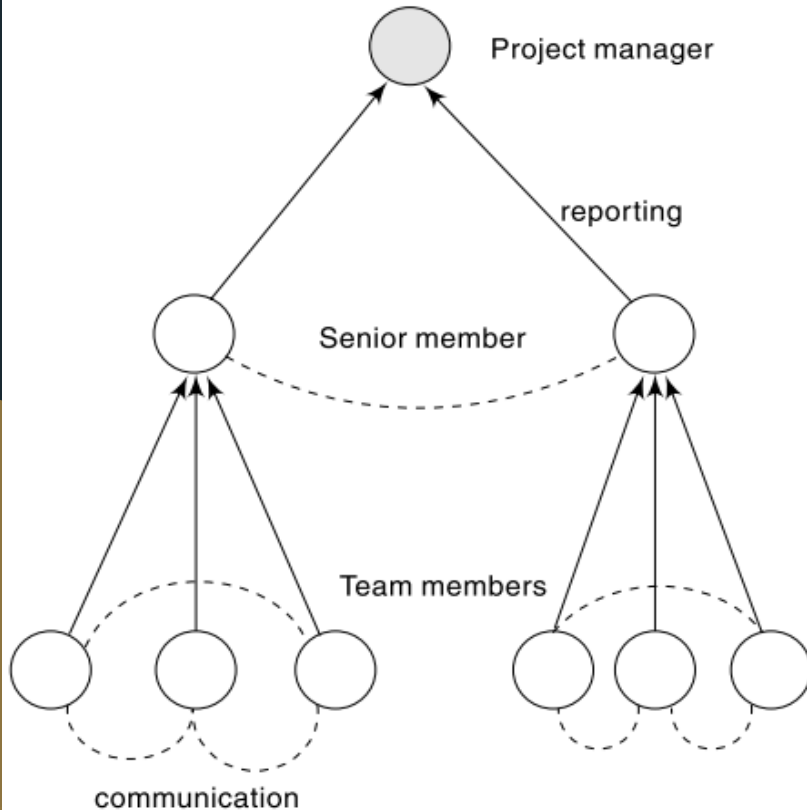
Team Organization and Structure

Democratic Team

- A collaborative model that contrasts with hierarchical structures.
- Particularly suited to creative, complex problem-solving environments.



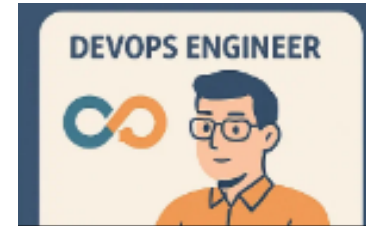
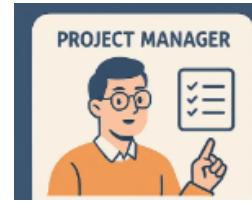
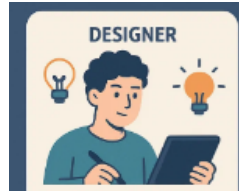
Team Organization and Structure



Mixed Control Team

- It's designed to balance the strengths of strong leadership with the benefits of collaboration and shared responsibility.

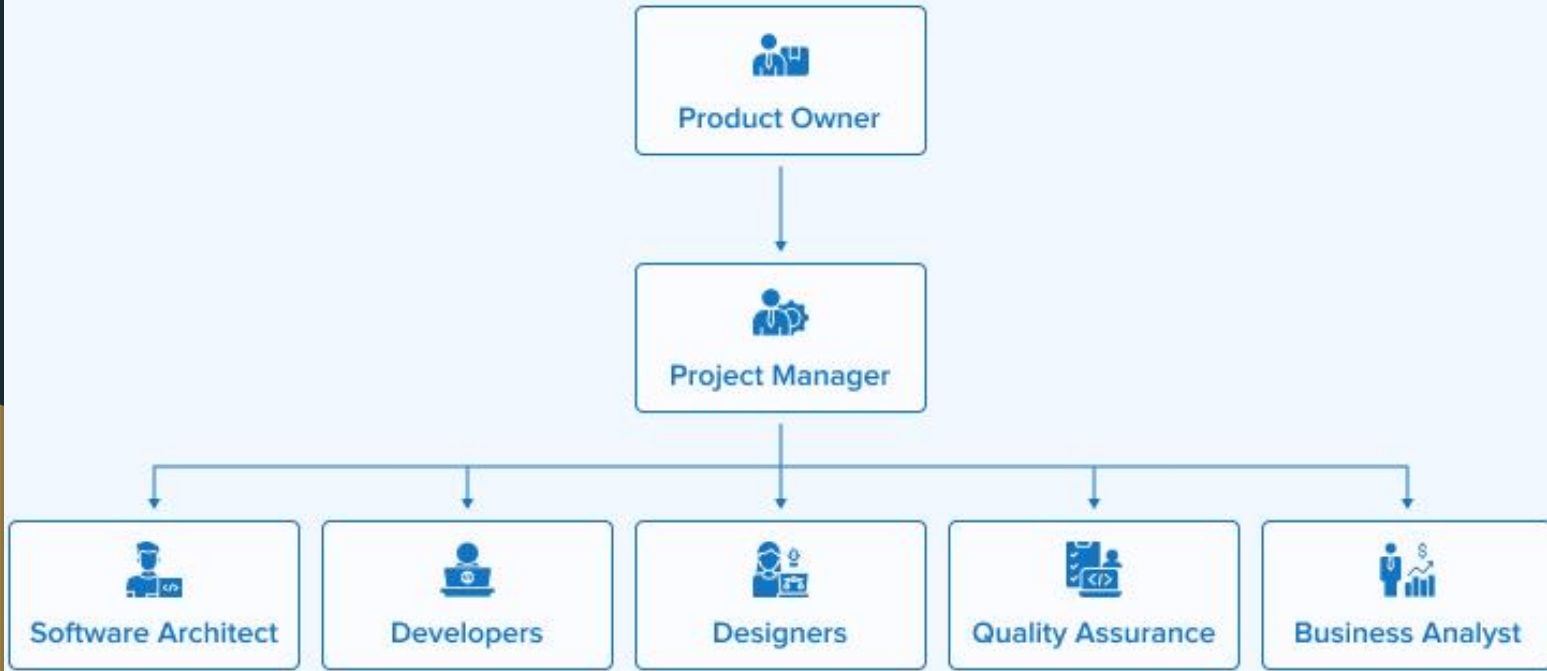
Team Roles and Responsibilities



Team Roles and Responsibilities

- Key roles:
 - **Product Owner:** serve as the bridge between business stakeholders and the development team
 - **Project Manager:** planning and overseeing the project
 - **Business Analyst:** gathering and analyzing requirement
 - **UI/UX Designer:** creating the visual design (interface and experience)
 - **Developer:** writing and testing code (coding and debugging)
 - **Tester:** QA and bug reporting
 - **DevOps Engineer:** Manage infrastructure

Team Roles and Responsibilities



software development team roles and structure (**source:**

<https://www.tatvasoft.com/outsourcing/2022/05/software-development-roles.html>

Types of Team

- Teams can be classified according to their objective.
- The Four most common forms of teams are:
 - Problem solving team: **typical size → 5-12**
 - Self-managed team: **typical size → 5-12**
 - Cross-functional team: **typical size → 6-12**
 - Virtual team: **typical size → varies widely (3 – 20+)**

Types of Team

Activity

- **How does a self-managed team differ from a traditional project team?**
 - A self-managed team operates without direct supervision, sharing responsibilities like planning and decision-making, while a traditional team relies on a manager for direction.

Types of Team - Virtual Team

Team members working from different geographic locations, often across time zones, using digital tools to collaborate



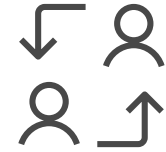
Geographically Distributed

Team members are working in different locations.



Technology driven communication

Communication relies on digital tools and platforms.



Cultural differences

Teams navigate diverse cultural norms and values.

Virtual Team - Challenges

- Communication Barriers
 - Lack of face-to-face interaction can cause misunderstandings.
- Coordination Overhead
 - Time zone and scheduling differences require more planning.
- Cultural Differences
 - May affect communication styles, feedback, and collaboration.
- Trust & Accountability
 - Harder to build team cohesion, monitor progress informally.

Tradeoffs and Enablers

- **Tradeoffs**
 - Speed Vs. Quality
 - Specialization Vs. Flexibility
 - Team size Vs. Communication Overhead
 - Autonomy Vs. Alignment
 - In-house Vs. Outsourced

Tradeoffs and Enablers

- **Enablers or success criteria**

- Clear team goals, measurements, and expected results
- Team Norms
- Individual & Shared Team Member Roles/Responsibilities
- Operating procedures/processes
- Clear & Open Communication
- Continuous Learning and Feedback

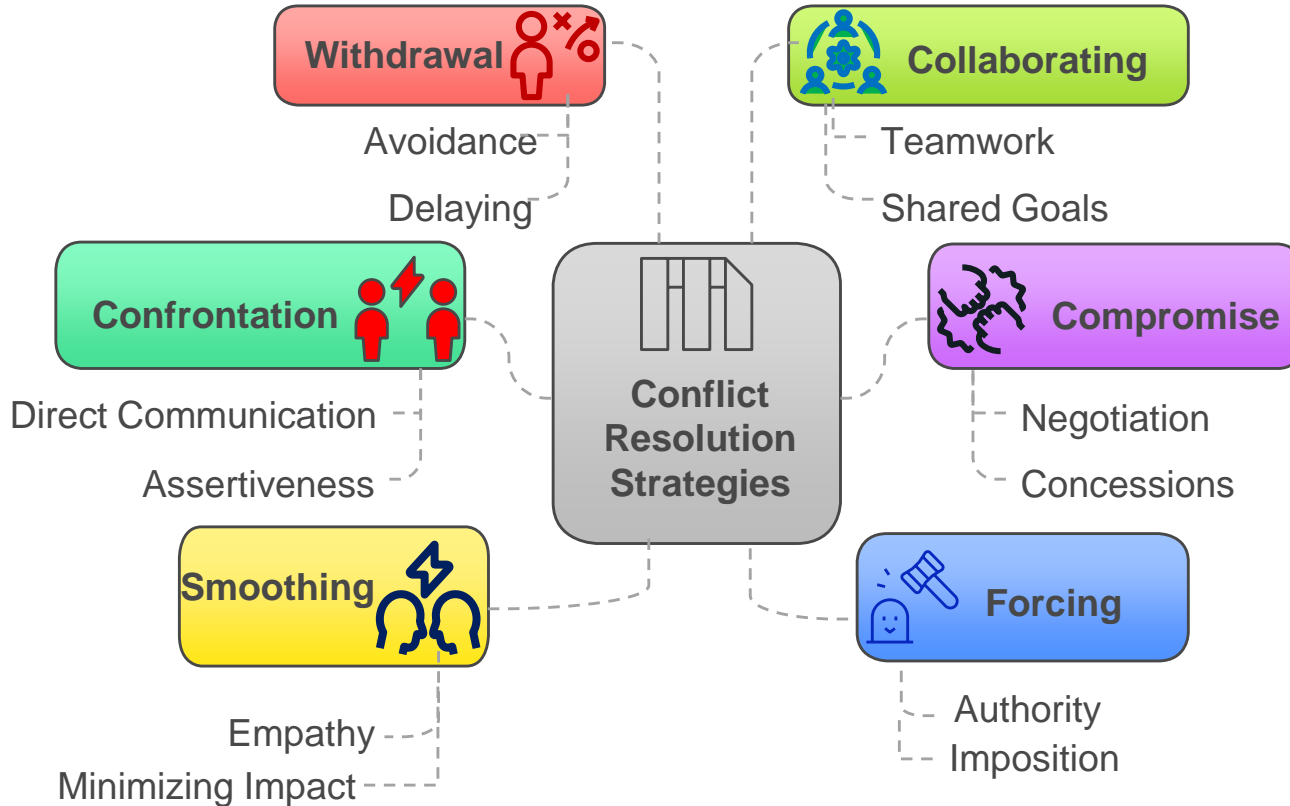
Tradeoffs and Enablers

- **Motivating and rewarding the team strategies**
 - Set Clear Goals
 - recognition and appreciation
 - Offer Growth Opportunities
 - **Public Recognition**
 - **Bonuses and Raises**
 - **Career Advancement**

Conflict Resolution & Problem Solving

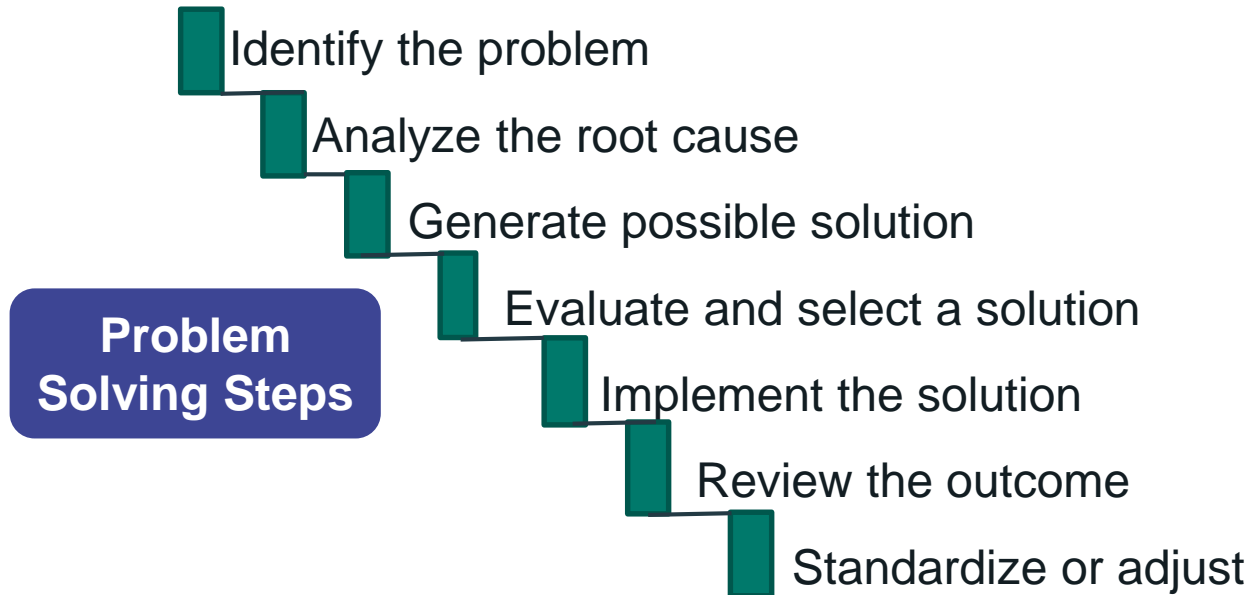
- Conflict is inevitable in any project environment (PMI, 2013), especially in software development teams, where
 - High-pressure deadlines
 - Differing technical opinions
 - Remote collaboration can easily create friction.
- The key is not to avoid conflict, but to resolve it constructively and use it as a catalyst for improvement

Conflict Resolution & Problem Solving



Conflict Resolution & Problem Solving

- A clear framework helps resolve not just conflict, but underlying technical or process-related **problems**



Conflict Resolution & Problem Solving

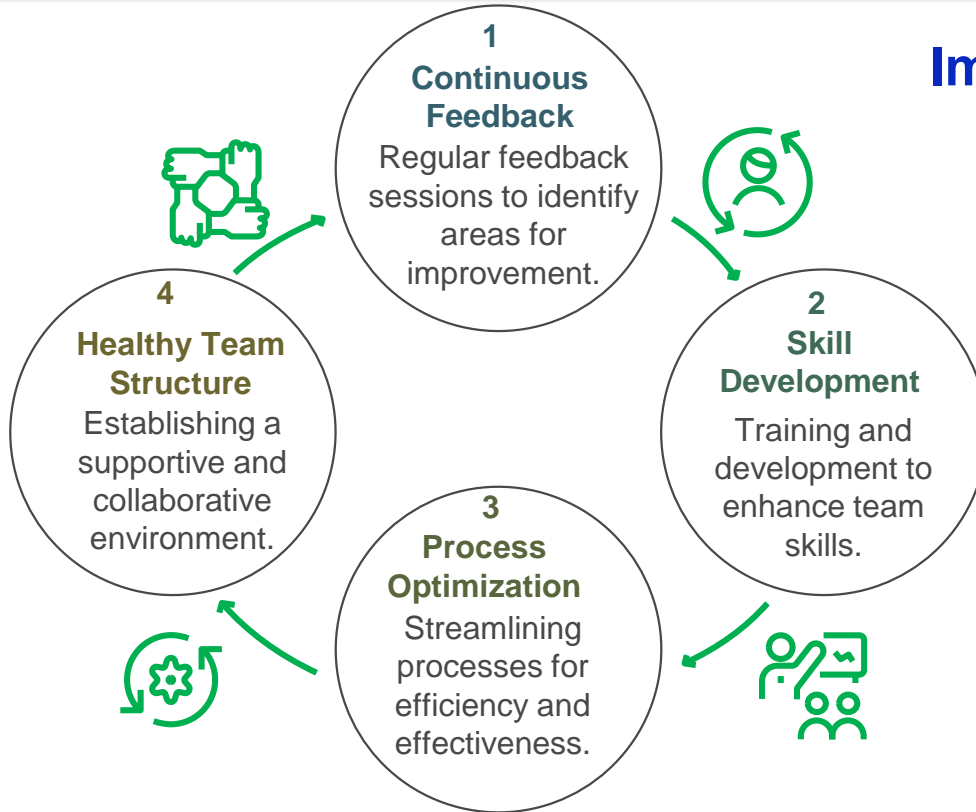
Example

- A software development team is working on an OLMS. The frontend developer and the backend developer disagree on how to implement a feature, the frontend wants to use a specific framework that the backend says will complicate integration with the existing API. Tension rises during sprint planning. The conflict begins to delay progress and affect team morale. **How can you resolve this?**
 - **Open discussion, identify root cause, consult technical lead**
 - **Agree on solution, retrospective reflection**

Measuring & Improving Team Performance

- Effectively measuring and improving team performance requires a combination of
 - Quantitative metrics, qualitative feedback, and continuous improvement practices.
- **Metrics**
 - Productivity
 - Quality
 - Collaboration and engagement
 - Predictability and delivery

Measuring & Improving Team Performance



Improving Team Performance

- Continuous Feedback Loops
- Skill development
- Process optimization
- Healthy team structure

Importance of Team Management

- Aligns people with project goals
- Enhances collaboration and coordination
- Improves productivity and engagement
- Reduces risks and delays
- Encourages accountability and ownership

Characteristics of effective teams

- Clear goals and defined roles
- Open communication and trust
- Shared decision-making
- Adaptability to change
- Focus on continuous improvement

Team Management – Tools & techniques

- Tools and techniques to manage project team (Schwalbe, 2014)
 - Observation and conversation
 - Project performance appraisals
 - Interpersonal skills
 - Conflict management:

Team Challenges

- Miscommunication
- Role ambiguity
- Cultural and personality clashes
- Resistance to change
- Distributed team issues (Remote work)

Summary

- Effective team management is a critical success factor in software project management.
- It involves planning, organizing, leading, and coordinating a group of individuals to collaboratively deliver high-quality software within time and budget constraints.

- **Key components**

Team formation & Development

Leadership & motivation

Communication

Performance measurement

Team Dissolution

Conflict Resolution

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. Schwalbe, K. (2014). Information Technology Project Management, Seventh Edition. Boston, USA: Cengage Learning