Managing software projects

Stamos Katsigiannis

Durham University

What is a project?

What is a project?

A temporary endeavor undertaken to create a unique product, service, or result.

A Guide to the Project Management Body of Knowledge (PMBOK® Guide) – Seventh Edition and The Standard for Project Management, Project Management Institute, Newtown Square, PA, 2021.

What is a project?

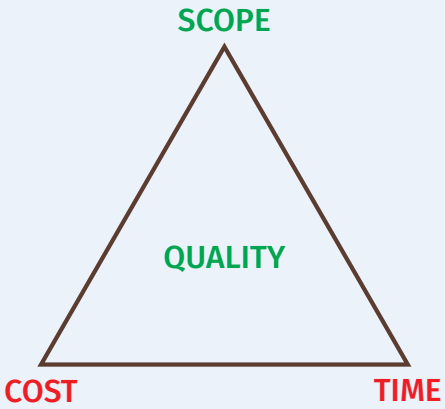
A **temporary** endeavor undertaken to create a **unique** product, service, or result.

A Guide to the Project Management Body of Knowledge (PMBOK® Guide) – Seventh Edition and The Standard for Project Management, Project Management Institute, Newtown Square, PA, 2021.

A project must have

- **beginning** and **end**
- clear **goal**

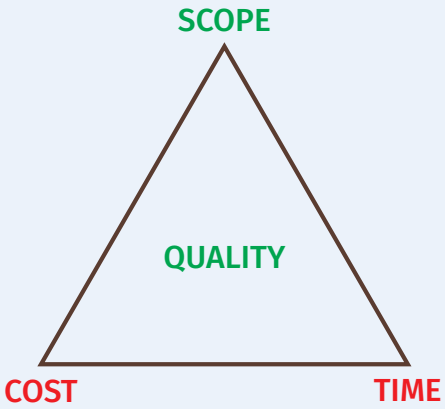
The golden/iron/project triangle

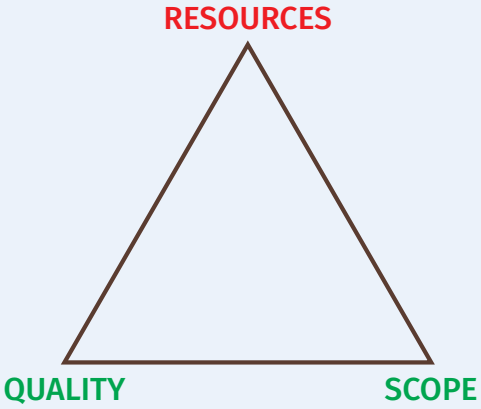


- **QUALITY**: non-functional requirements
- **SCOPE**: functional requirements
- **COST**: budget/human constraints
- **TIME**: delivery constraints

Model: SCOPE $=$ TIME $\times$ COST

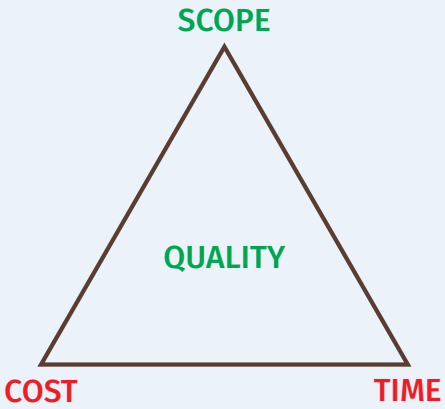
Good, fast, cheap: choose two





Difference: All constraints are now in **RESOURCES**

The iron triangle in software projects



- **QUALITY**: non-functional requirements
- **SCOPE**: functional requirements
- **COST**: budget/human constraints
- **TIME**: delivery constraints

Model: SCOPE $=$ TIME $\times$ COST

Requirements

Non-functional (QUALITY)

Functional (SCOPE)

Requirements

Non-functional (QUALITY)

- performance
- reliability
- scalability
- security
- documentation
- extensibility

Functional (SCOPE)

Requirements

Non-functional (QUALITY)

- performance
- reliability
- scalability
- security
- documentation
- extensibility

Functional (SCOPE)

- features of the software
- data software can handle
- required user interactions

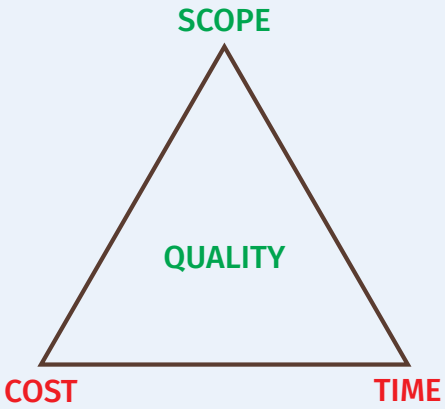
Constraints

- software
- hardware
- peopleware
- delivery deadlines

P. G. Neumann. *Peopleware in systems*. in Peopleware in Systems. Association for Systems Management, Cleveland, OH, 1976, pp. 15-18.

T. DeMarco and T. Lister. *Peopleware: Productive Projects and Teams*. Addison-Wesley, 2013.

Iron triangle—messages



- Constraint/requirements **can** change
- Any change requires adjustments
- Tool for analysis, **not** assessment
- Impact on stakeholders is missing

Model: SCOPE $=$ TIME $\times$ COST

What is software engineering?

What is software engineering?

Engineering is about getting results of the required quality within schedule and budget. This often involves making compromises—engineers cannot be perfectionists.

I. Sommerville. *Software Engineering*. Tenth edition, Pearson Education, Harlow, Essex, UK, 2016.

What is software engineering?

> Engineering is about getting results of the required quality within schedule and budget. This often involves making compromises—engineers cannot be perfectionists.
>
> I. Sommerville. *Software Engineering*. Tenth edition, Pearson Education, Harlow, Essex, UK, 2016.

Software engineering = management of software projects

What is software engineering?

> Engineering is about getting results of the required quality within schedule and budget. This often involves making compromises—engineers cannot be perfectionists.
>
> I. Sommerville. *Software Engineering*. Tenth edition, Pearson Education, Harlow, Essex, UK, 2016.

Software engineering = management of software projects

Core **hard skills**, but **soft skills** are also involved!

Summary

1. What falls under **QUALITY**?

2. What falls under **SCOPE**?

3. What falls under **RESOURCES**?

4. What is **SOFTWARE ENGINEERING**?

Summary

1. What falls under **QUALITY**?
   Non-functional requirements.
2. What falls under **SCOPE**?
3. What falls under **RESOURCES**?
4. What is **SOFTWARE ENGINEERING**?

Summary

1. What falls under **QUALITY**?
   Non-functional requirements.
2. What falls under **SCOPE**?
   Functional requirements.
3. What falls under **RESOURCES**?

4. What is **SOFTWARE ENGINEERING**?

Summary

1. What falls under **QUALITY**?
   Non-functional requirements.
2. What falls under **SCOPE**?
   Functional requirements.
3. What falls under **RESOURCES**?
   *All constraints (**COST** and **TIME**).*
4. What is **SOFTWARE ENGINEERING**?

Summary

1. What falls under **QUALITY**?
   Non-functional requirements.
2. What falls under **SCOPE**?
   Functional requirements.
3. What falls under **RESOURCES**?
   *All constraints (**COST** and **TIME**).*
4. What is **SOFTWARE ENGINEERING**?
   Techniques for managing software projects.

Questions?