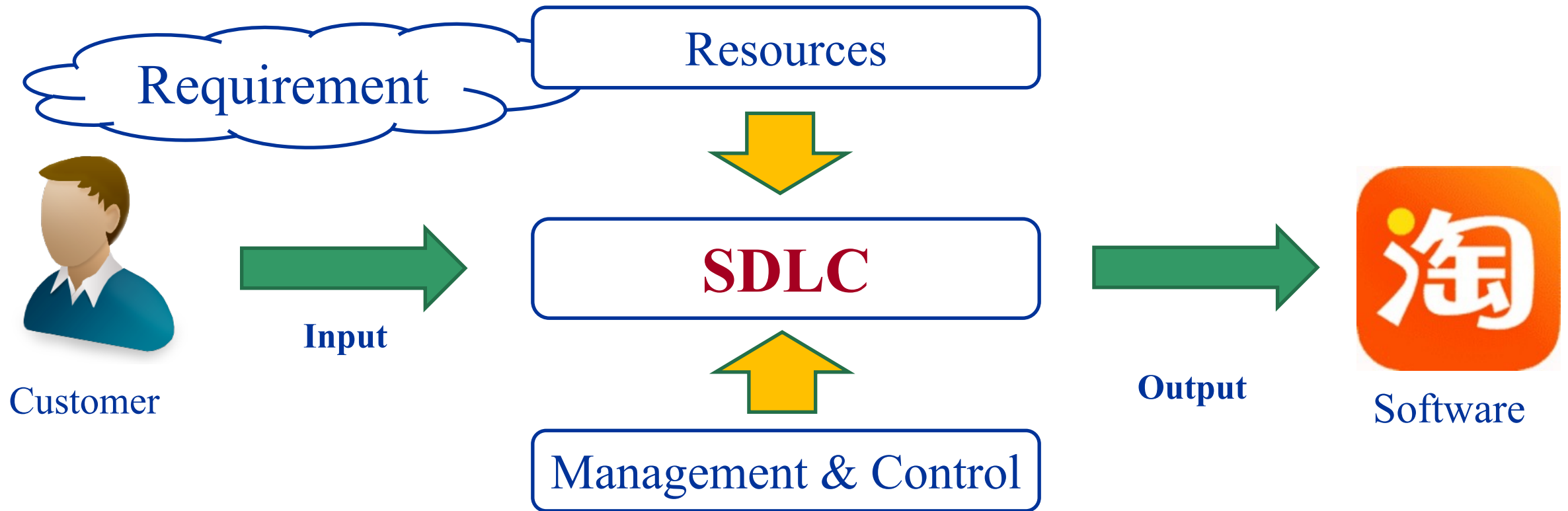
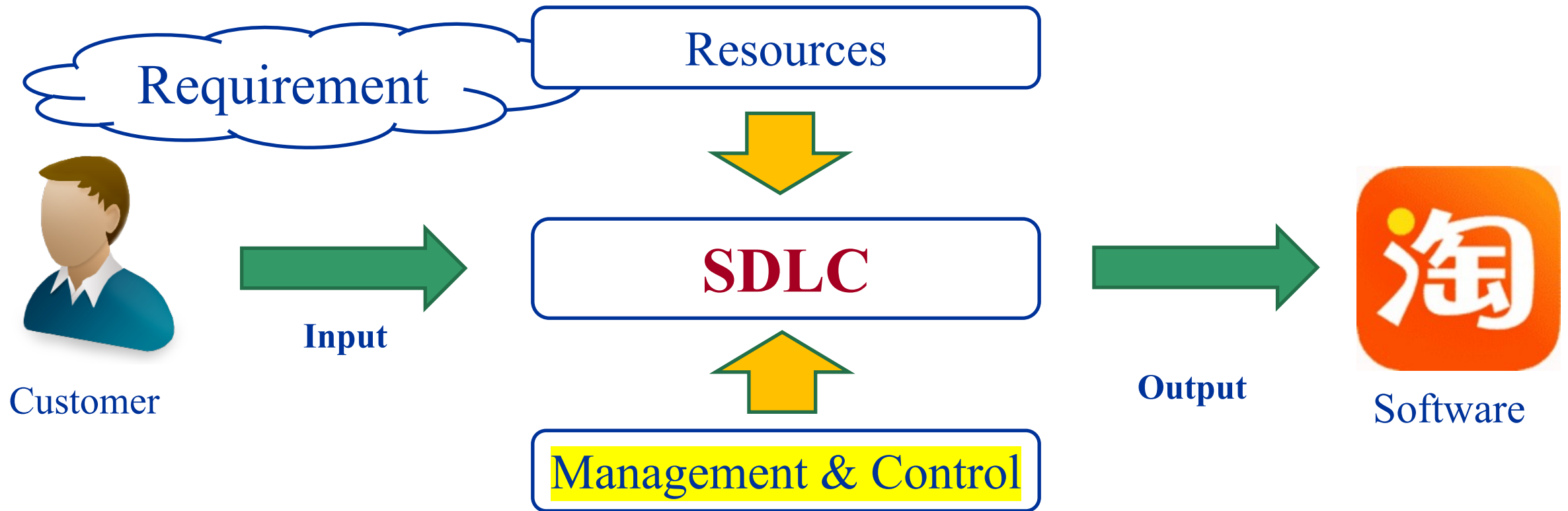


Software Development Life Cycle (SDLC)



Software Development Life Cycle (SDLC)



Lecture 4: Software Project Management (1)

Outline

- What is a Project?
- How to Make Software Projects Fail?
- What is Software Project Management?
- The Role of the Software Project Manager
- Project Scope Management

What is a Project?

What is a Project?

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

-source PMBoK, p368



Examples of IT Projects

- A large network of healthcare providers update its information systems and procedures to reduce hospital acquired diseases.
- A team of students creates a smartphone application and sells it online.
- A company develops an autonomous vehicle.
- A college upgrades its technology infrastructure to provide wireless Internet access across the whole campus as well as online access to all academic and student service information.

Project Attributes

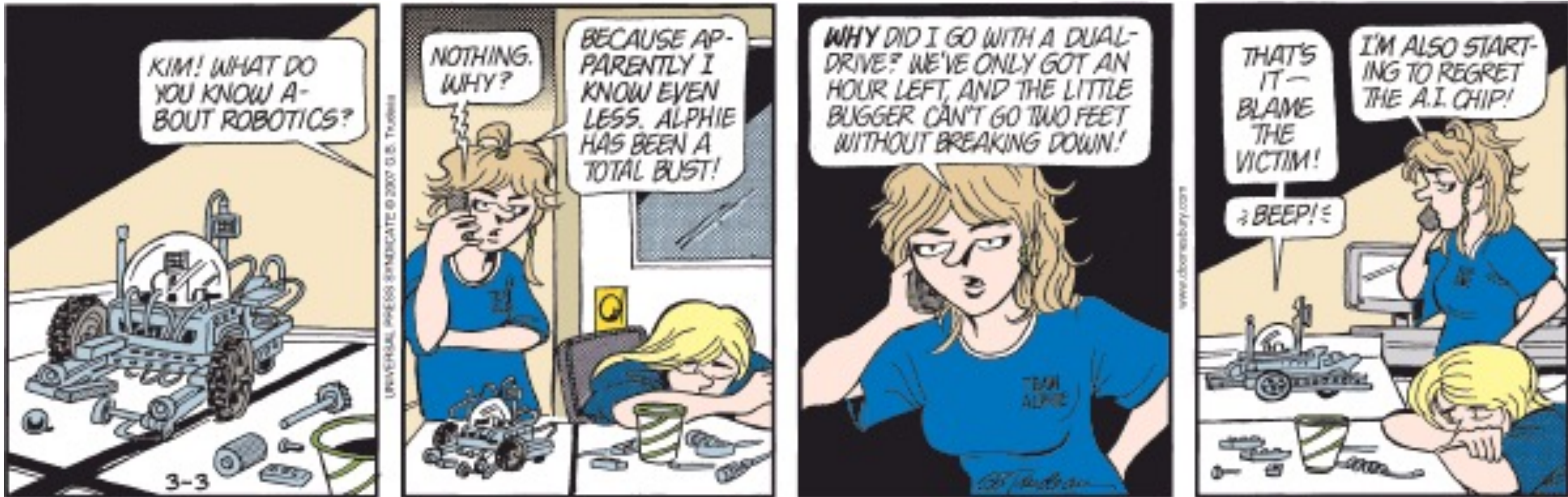
- A project has a **unique purpose**.
 - Every project should have a well-defined objective.
- A project is **temporary**.
 - A project has a definite beginning and end.
- A project is developed using **progressive elaboration**
 - Projects are often defined broadly when they begin, and as time passes, the specific details of the project become clearer.
- A project **requires resources**, often from various areas
 - Resources include people, hardware, software, and other assets.
 - Marketing, sales, IT, etc.

Project Attributes (2)

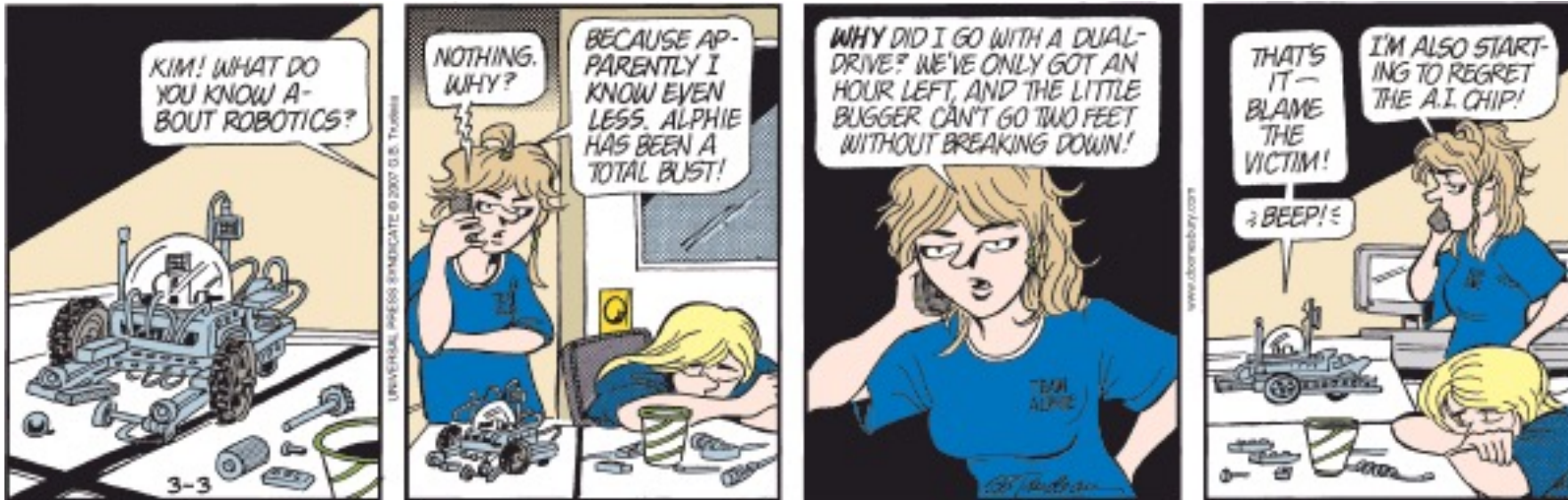
- A project should **have a primary customer or sponsor**.
 - Most projects have many interested parties or stakeholders, but for a project to succeed someone must take the primary role of sponsorship.
- A project involves **uncertainty**.
 - Because every project is unique, it is sometimes difficult to define its objectives clearly, estimate how long it will take to complete, or determine how much it will cost.

How to Make Software Projects Fail?

How to have an unsuccessful project



How to have an unsuccessful project (2)



- Team dynamics: team mate is crashed out
- Scheduling: only an hour left?
- Overambitious: dual drive? AI Chip?

How to have an unsuccessful project (3)

- Team dynamics: team mate is crashed out
 - Time & Scope Problem, Quality suffers
- Scheduling: only an hour left?
 - Time & Scheduling Problem
- Overambitious: dual drive? AI Chip?
 - Scope: Vague requirements

Common Problems with Software Projects

- Lack of quality standards and measures
- Lack of measurable milestones
- Difficult to make the progress visible
- Poor communications
- Poor documentation
- Frequent changes of requirements
- Over budget and late delivery of software

Project Success

- The project met scope, time, and cost goals.
 - **Scope** refers to **all the work involved** in creating the products of the project and the **processes used** to create them. It defines what is or is not to be done.
- The project satisfied the customer/sponsor.
 - E.g., never return calls, customer not happy with important aspects of the project
- The results of the project met its main objective.
 - E.g, making or saving a certain amount of money, providing a good return on investment, or simply making the sponsors happy

What helps Projects Succeed?

- 1. Executive support
- 2. User involvement
- 3. Clear business objectives
- 4. Emotional maturity
- 5. Optimizing scope

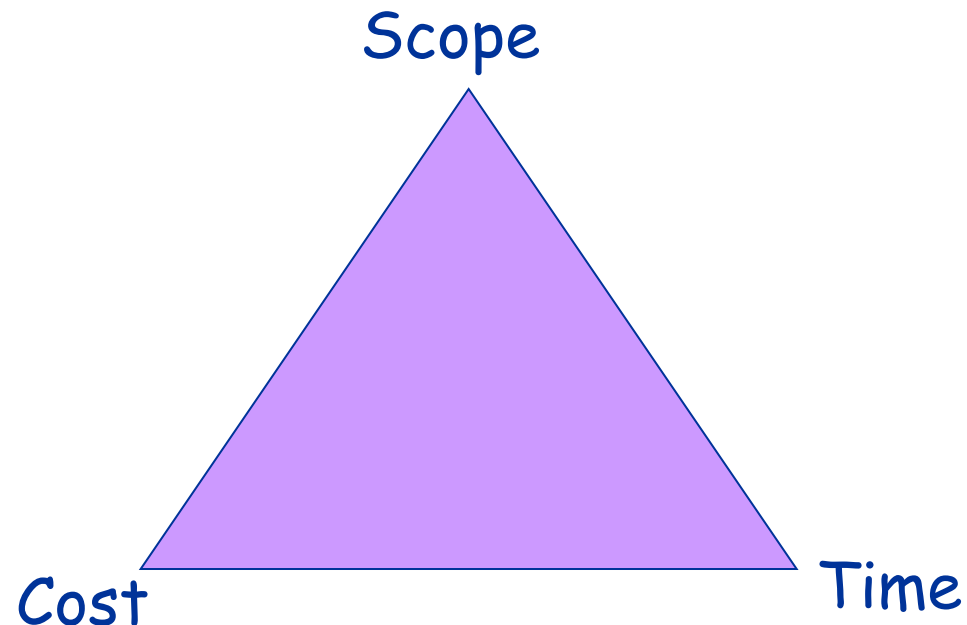
What helps Projects Succeed? (2)

- 6. Agile process
- 7. Project management expertise
- 8. Resources
- 9. Execution
- 10. Tools and infrastructure

Software Project Management

Software Project Management

- Software Project Management: the application of knowledge, skills, tools, and techniques to project activities to **meet project requirements**.
- Project Management Triple Constraints:

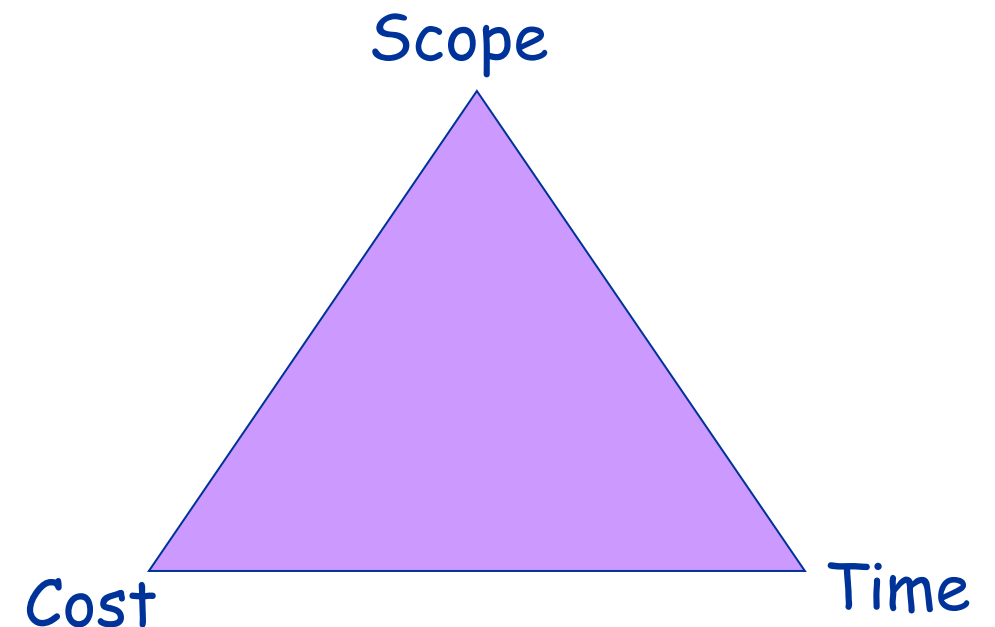


Software Project Management (2)

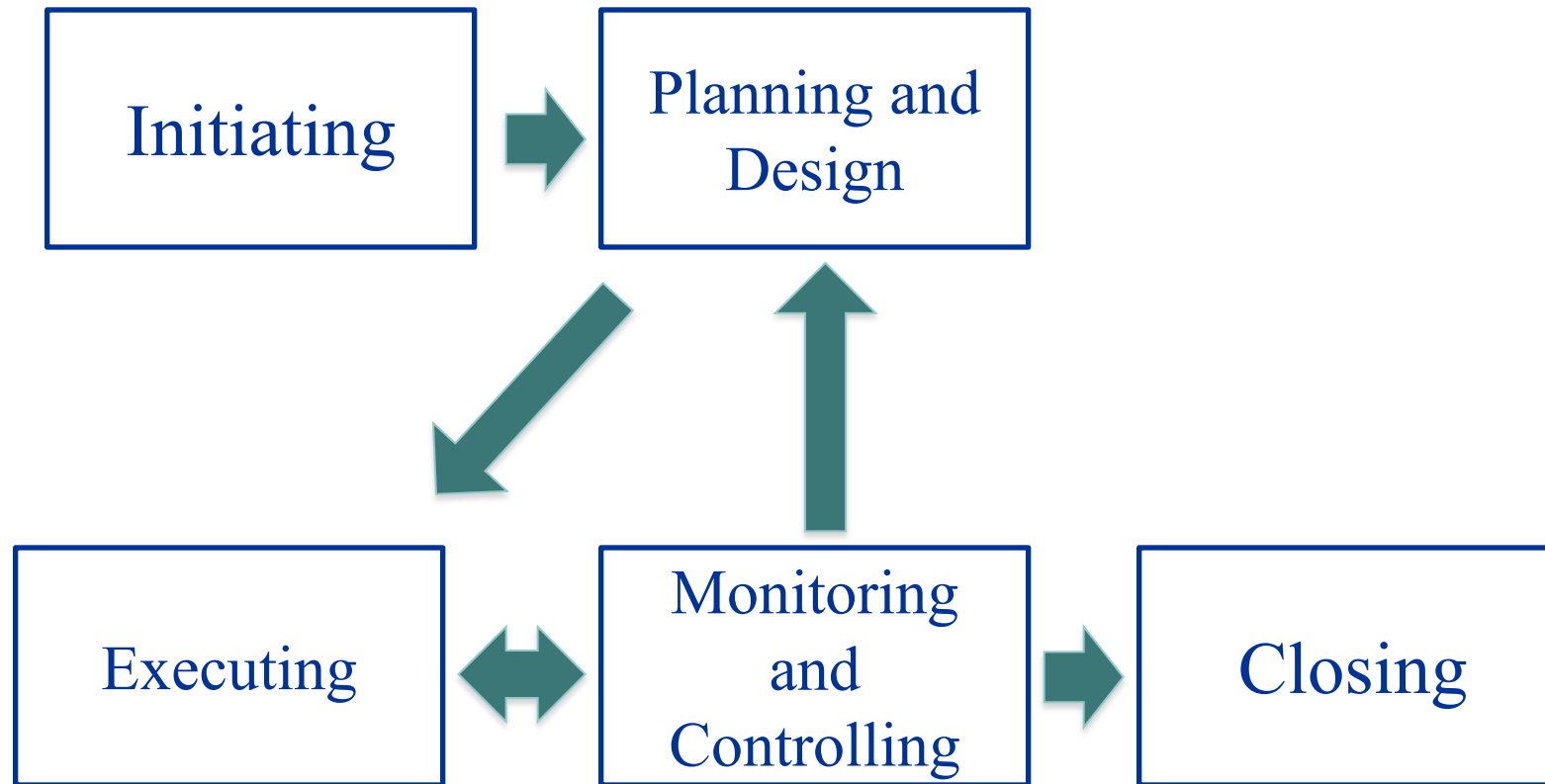
Q: What is a project?

A: “A temporary endeavor undertaken to create a unique *product, service or result*”

Where are time, cost, scope?



Software Project Management Phases



Important Goals

The success criteria for project management vary from project to project, but for most projects, important goals are:

- to deliver the software to the customer at the agreed time;
- to keep overall costs within budget;
- to deliver software that meets the customer's expectation;
- to maintain a coherent and well-functioning development team;

Factors that Affect Project Management

Some of the most important factors that affect how software projects are managed are:

- Company size;
- Software customers;
- Software size;
- Software type;
- Organizational culture;
- Software development model;

Factors that Affect Project Management (2)

- Company size
 - Small company
 - Informal management and team communication;
 - Do not need formal policies;
 - Do not need management organization;
 - Large company
 - Management hierarchies;
 - Formal reporting and budgeting;
 - Approval processes must be followed;

Factors that Affect Project Management (3)

- Software customers
 - Internal customers
 - Informal communication
 - No need to fit in with the customer's ways of working
 - External customers
 - Formal communication channel
 - Try to fit in with the customer's ways of working
 - Government agency
 - Formal communication channel
 - Must operate according to the agency's policies and procedures

Factors that Affect Project Management (4)

- Software size
 - Small systems
 - Small team;
 - Large systems
 - Multiple development teams;
 - Teams may be geographically distributed and in different companies;
 - Project manager has to **coordinate** the activities of these teams and **arrange** for them to communicate with each other;

Factors that Affect Project Management (5)

- Software type
 - Consumer product
 - Formal records of project management decisions are unnecessary.
 - Safety-critical system
 - Project management decisions should be record and justifies as these may affect the safety of the system.

Factors that Affect Project Management (5)

- Organizational culture
 - Different companies may have different project management culture;
 - CompanyA: supporting and encouraging individuals;
 - CompanyB: group focused;
 - CompanyC: take risks;
 - CompanyD: risk averse;

Factors that Affect Project Management (6)

- Software development model
 - Agile
 - “lightweight” management
 - Waterfall
 - More formal processes
 - Project management monitoring

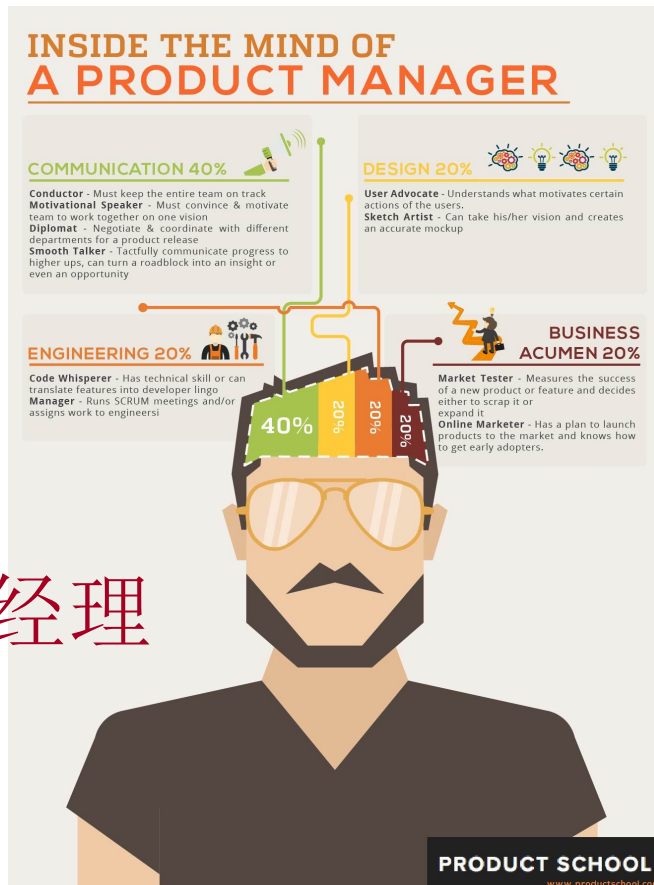
The Role of a Project Manager

- Project managers coordinate the efforts of project teams, functional groups, suppliers, and operations staff supporting the projects to ensure that products and processes are implemented to maximize benefits.
- Project managers are responsible for more than the delivery of project results; they are change agents responsible for the success of products and processes developed by those projects.

Skills for Project Managers

- The Project Management Body of Knowledge.
- Application area knowledge, standards, and regulations.
- Project environment knowledge.
- General management knowledge and skills.
- Soft skills or human relations skills.

Product Manager vs. Project Manager



产品经理



项目经理

Product Manager vs. Project Manager (2)

- Product Manager

- A product manager's role is strategic, much like a CEO but for the product

做正确的事情

- Project Manager

- A project manager's role, on the other hand, is more tactical, focusing primarily on the execution side.

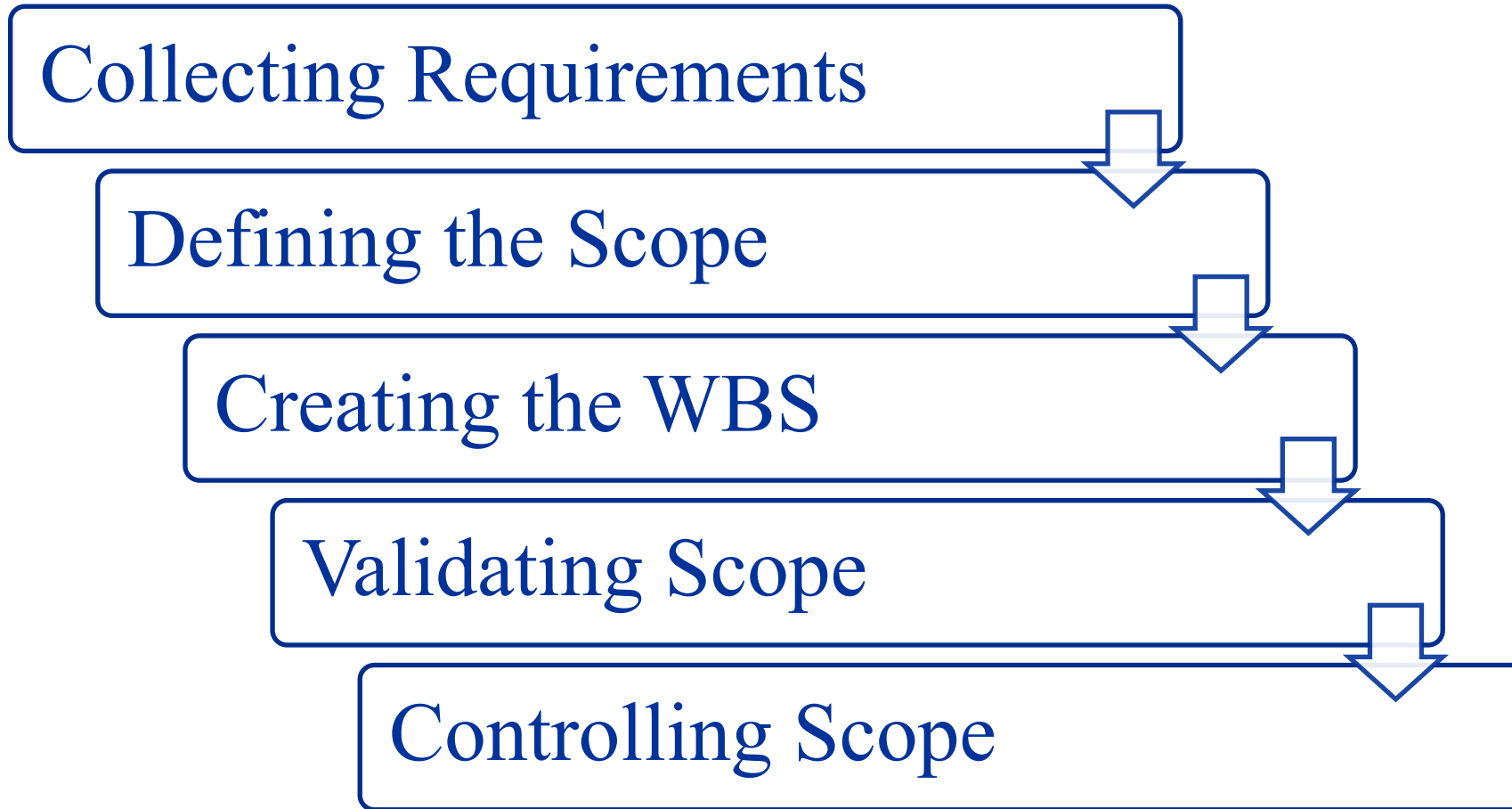
把事情做对

Project Scope Management

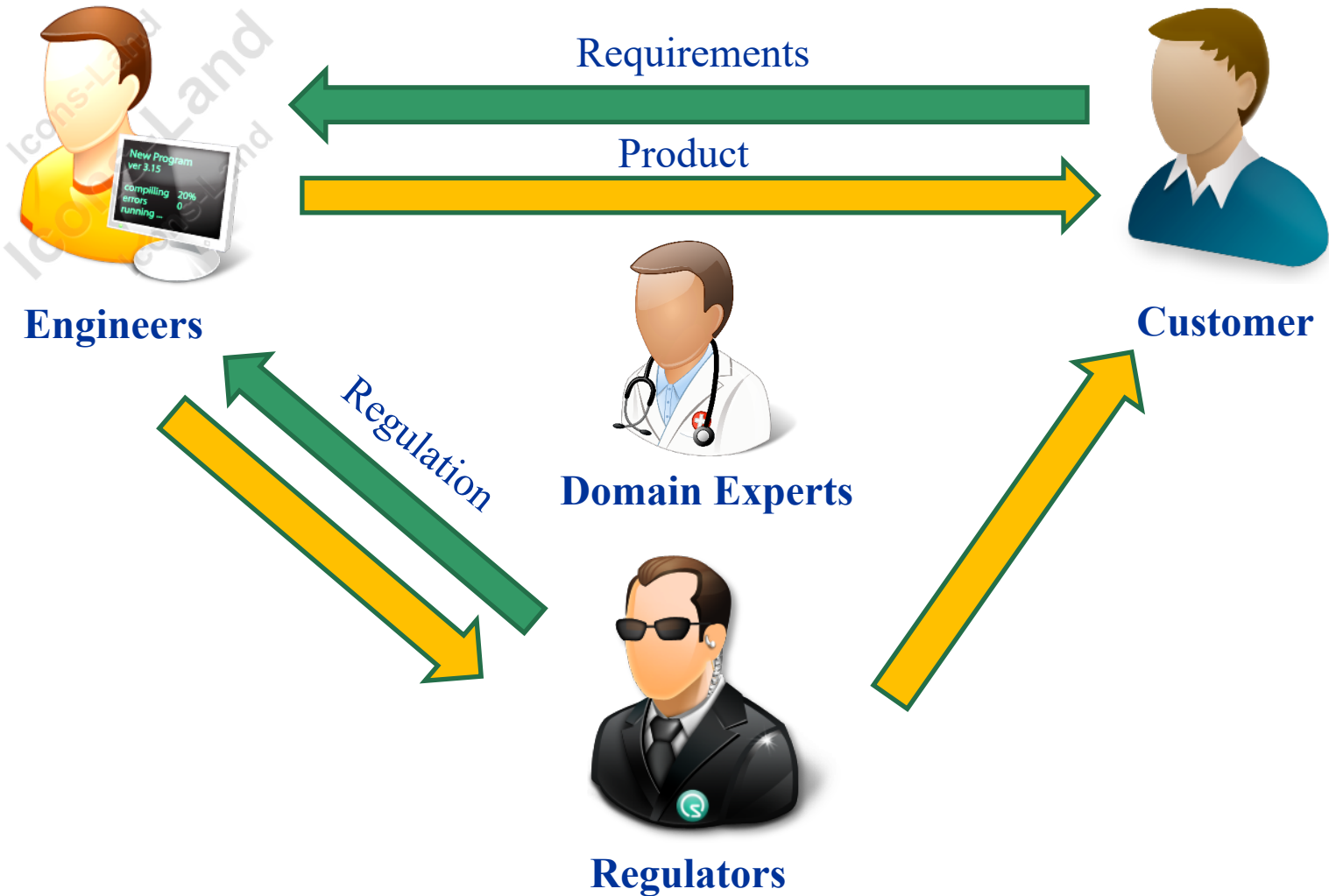
Project Scope Management

- **Scope** refers to all the work involved in creating the products of the project and the **processes used** to create them. It defines what is or is not to be done.
- **Deliverables** are products produced as part of a project, such as hardware or software, planning documents, or meeting minutes.
- Project **stakeholders** must agree
 - What the deliverables of project are;
 - How they should be produced.

Workflow of Project Scope Management



Collecting the Requirement



Defining the Scope

- This step provides a detailed definition of the work required for the project.
- Good scope definition is very important to project success because
 - Improve the accuracy of time, cost, and resource estimates;
 - Define a baseline for performance measurement and project control;
 - Aids in communicating clear work responsibilities.

Defining the Scope (2)



Defining the Scope (3)

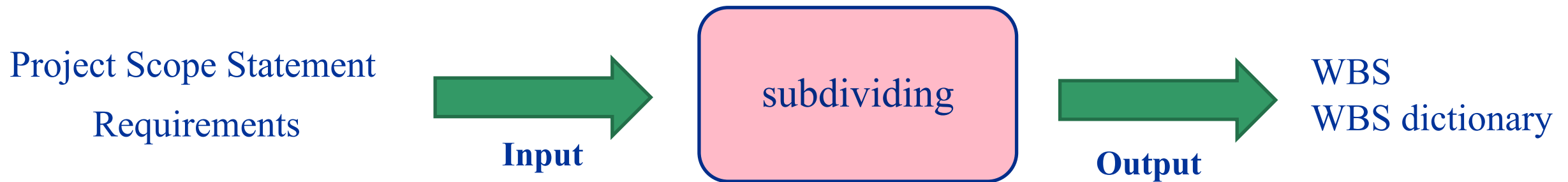
- To develop an effective project scope statement, you should:
- Outline project scope description
 - Objectives, feature, specifications, details about the project
- Select major deliverables
 - Document, software, source code, data, ...
- Identify key milestones
 - A key milestone indicates when stakeholders can expect a particular deliverable to be completed.

Defining the Scope (4)

- Identify major constraints
 - Personnel, resources, schedule, or other requirements
- List scope exclusions
 - The list consists of deliverables that a project sponsor **may assume are included** in the scope of the project **but are not**, in fact, included.

Creating the WBS

- Work Breakdown Structure (WBS)
- WBS: the process of **subdividing** project deliverables and project work into smaller, more manageable components.



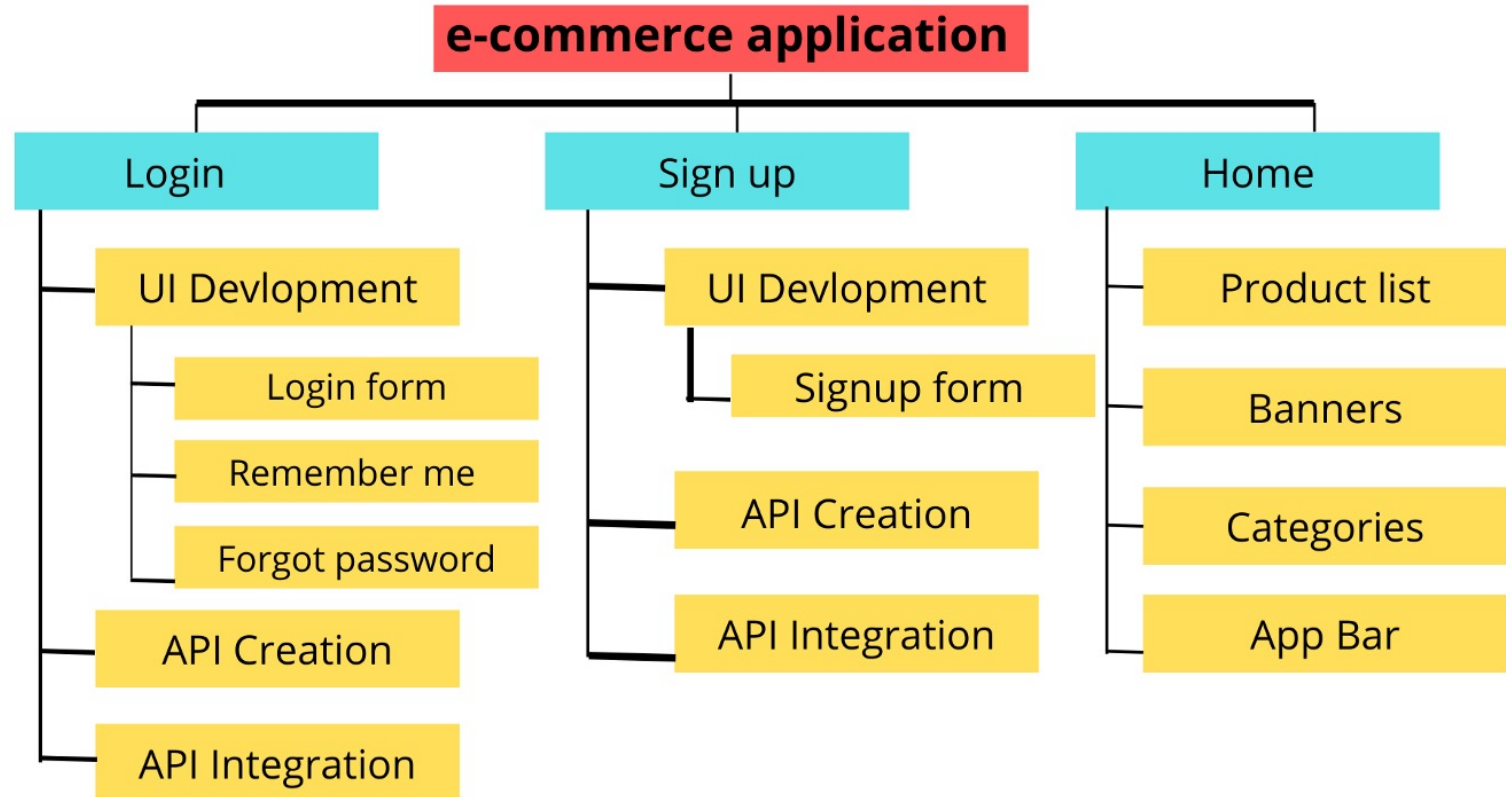
Creating the WBS: Example



Design an E-commerce app



WBS

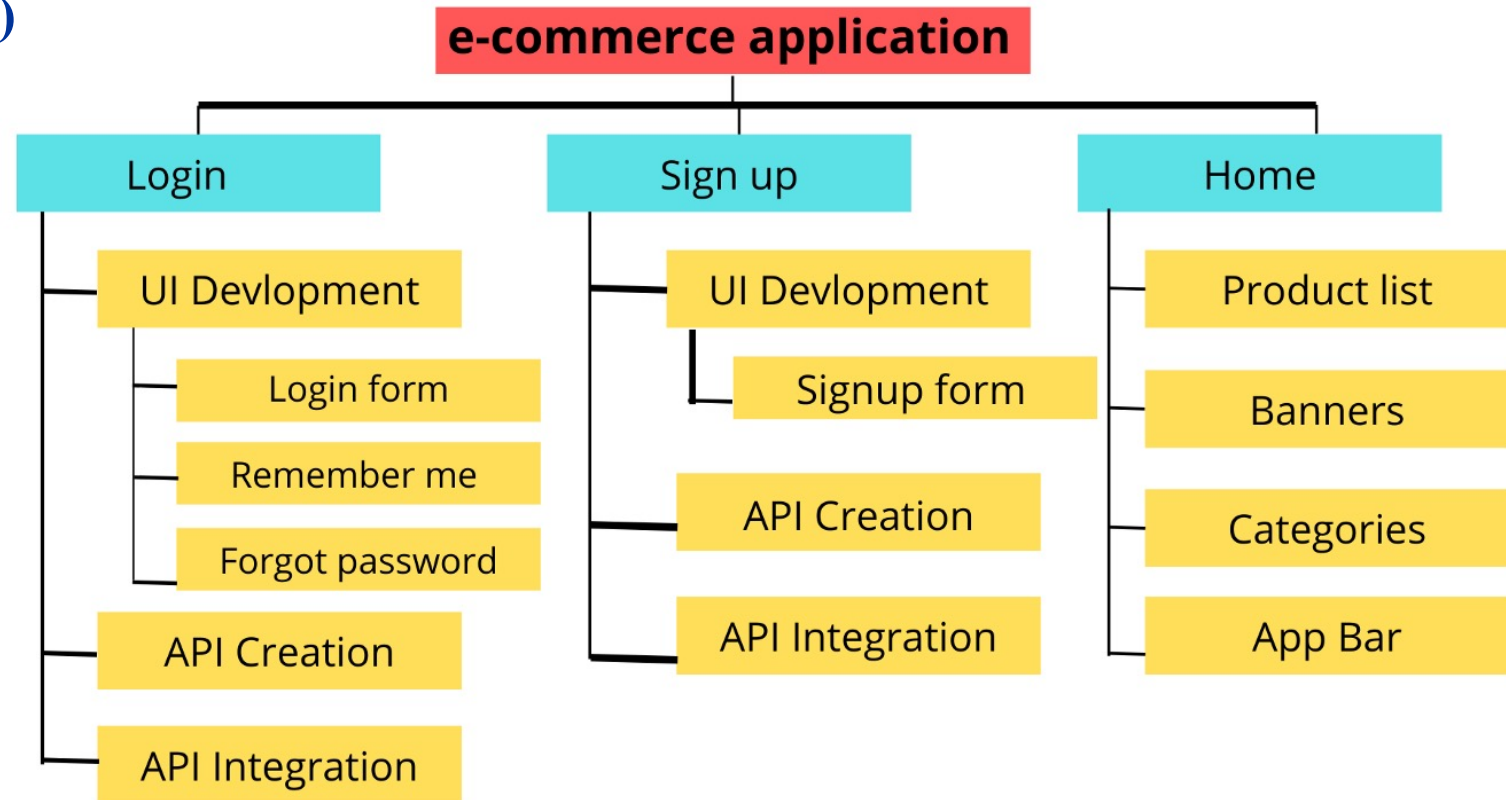


Creating the WBS: Example (2)

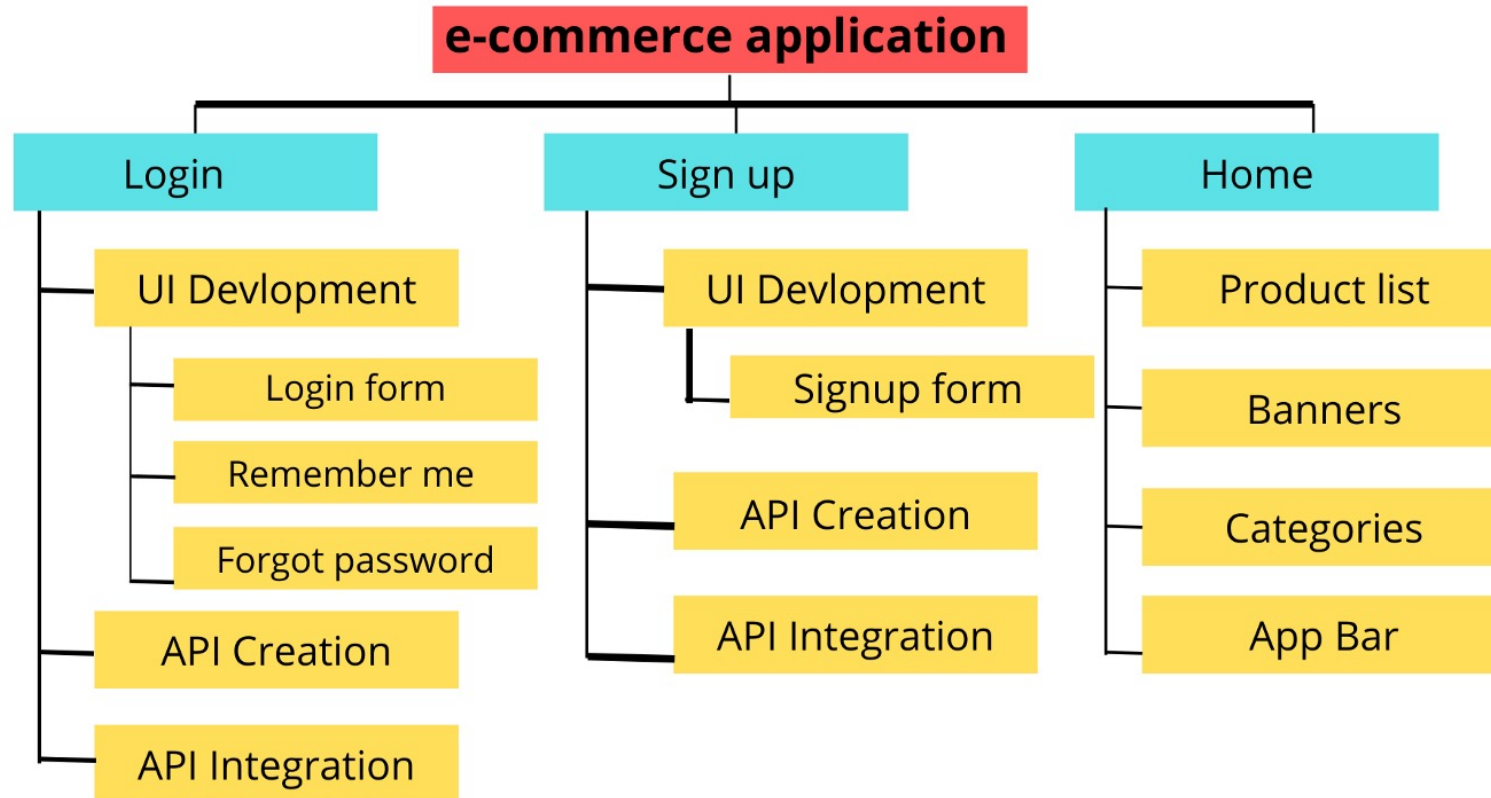
Level 1 (Entire Project)

Level 2 WBS

Level 3 WBS



Defining Scope vs WBS



Defining Scope vs WBS

Defining Scope

Login

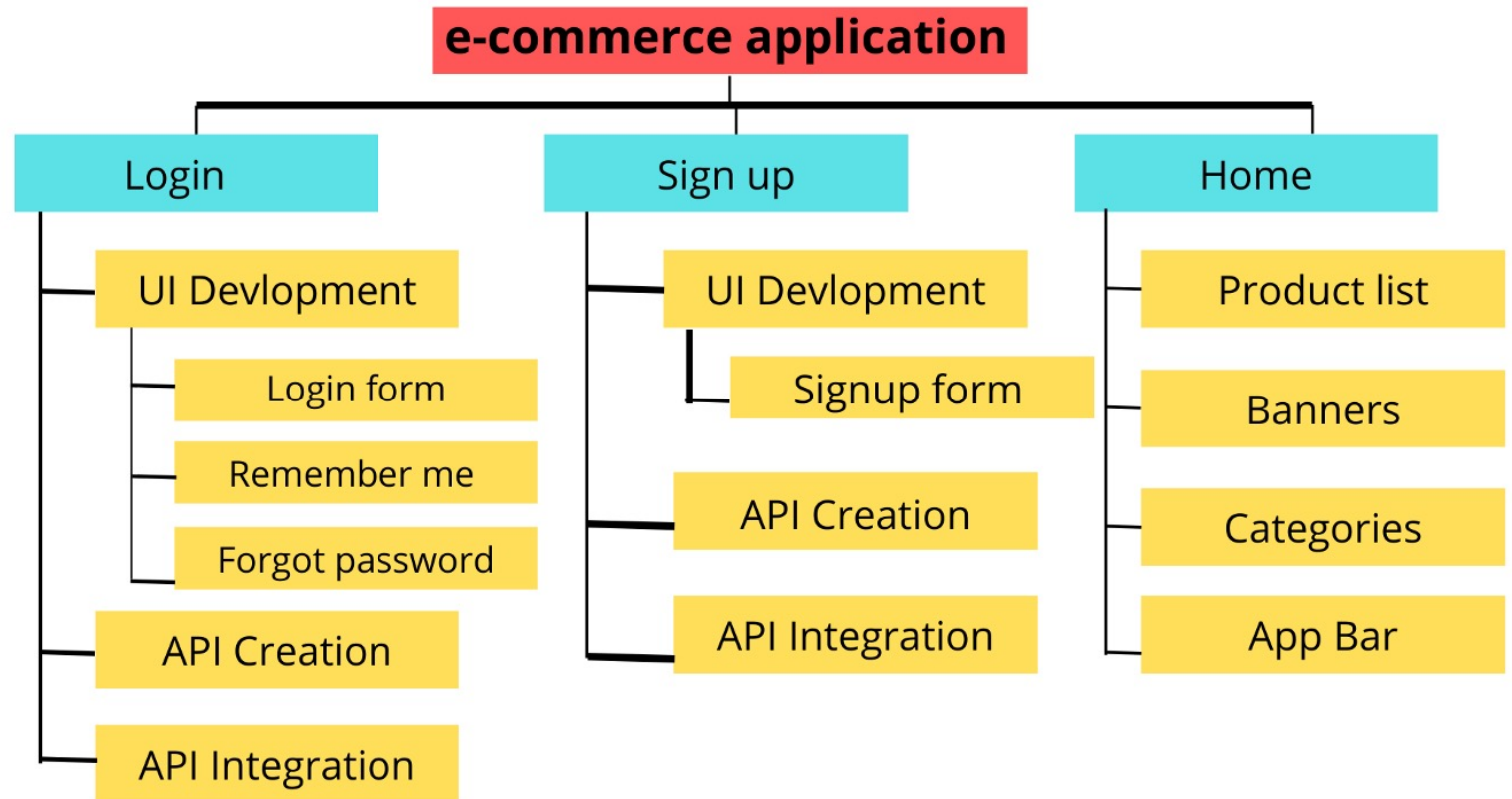
Sign up

Home

Product List

(collecting requirements)

WBS

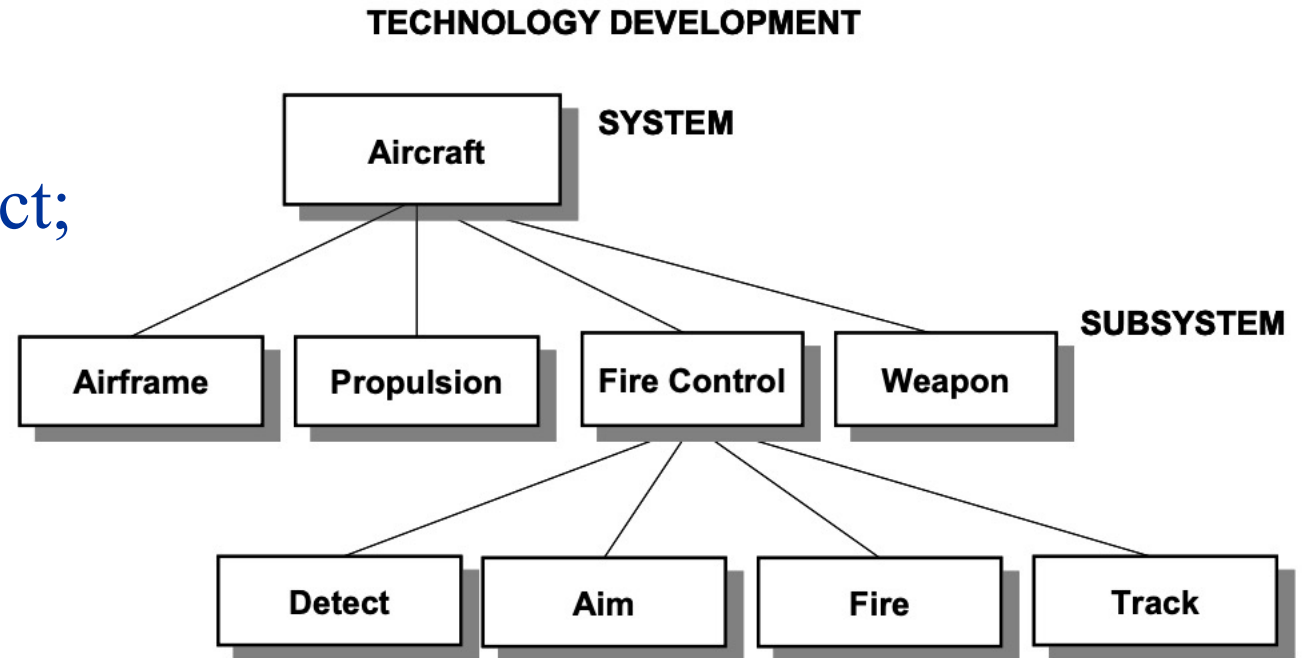


Approaches to develop WBS

- There are several approaches to develop a work breakdown structure:
 1. Using guidelines
 2. The analogy approach
 3. The top-down approach
 4. The bottom-up approach
 5. The mind-mapping approach

Using Guidelines

- If **guidelines** exist for developing a WBS, it is very important to **follow them**.
- Example:
 - USA DOD Handbook;
 - WBS Templates in MS Project;



The Analogy Approach

- In the analogy approach, you use a **similar project's WBS** as a **starting point**.
- Example
 - Boeing, provides an example of using an analogy approach when creating WBSs.
 - When creating a WBS for a new aircraft design, it started by using **74 predefined subsystems** for building fighter aircraft based on past experience.

The Top-down Approach

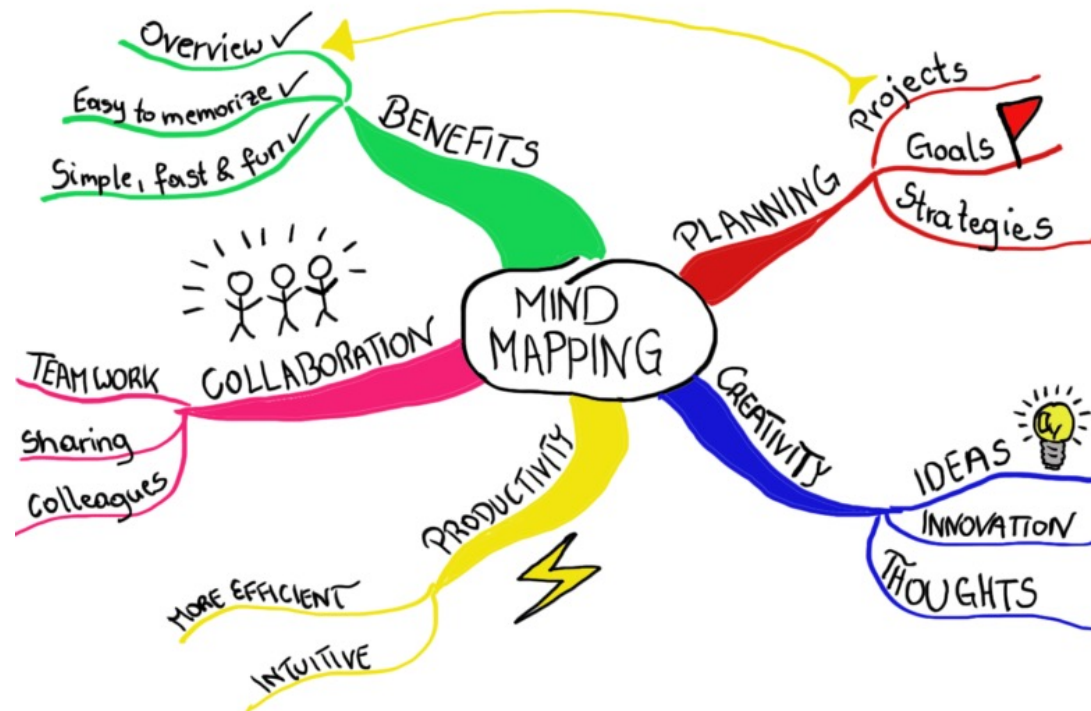
- To use the top-down approach, start with the largest items of the project and break **them into subordinate items**.
- This process involves refining the work into greater and greater levels of detail.

The Bottom-up Approach

- In the bottom-up approach, team members first identify as many specific tasks related to the project as possible.
- They then aggregate the specific tasks and organize them into summary activities, or higher levels in the WBS.

The Mind-mapping Approach

- Mind mapping is a technique that uses branches radiating from a core idea to structure thoughts and ideas.



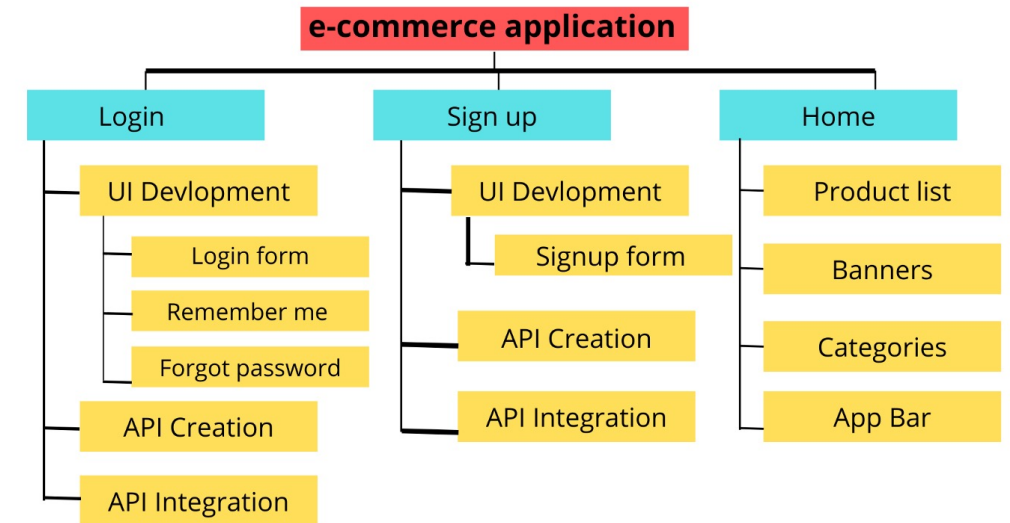
Recap



Design an E-commerce app

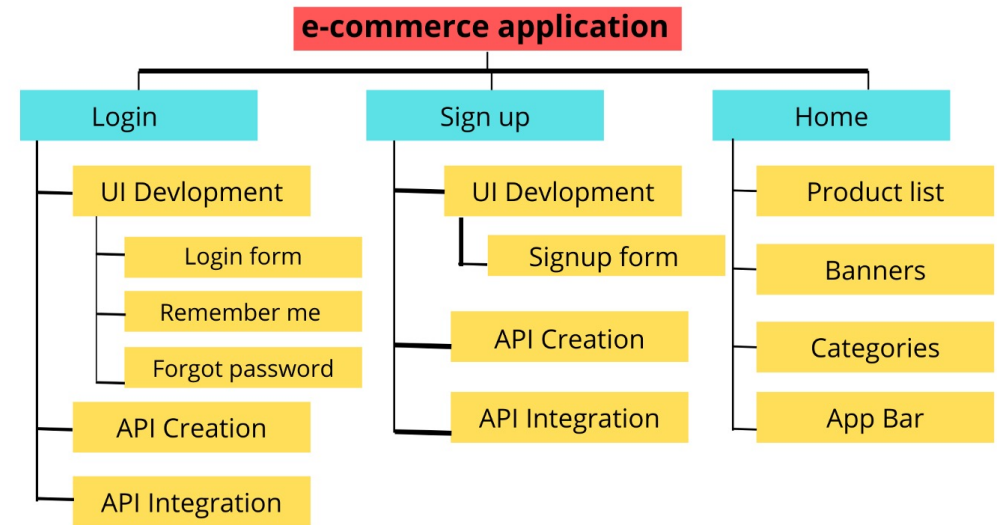


Using Guidelines
Analogy Approach
Mind-mapping Approach
Top-down Approach
Bottom-up Approach



WBS Dictionary

- Many of the items listed on the WBSs are rather vague.
- For example, what exactly does “UI Development” mean?
- A **WBS dictionary** is a document that provides detailed information about each WBS item.



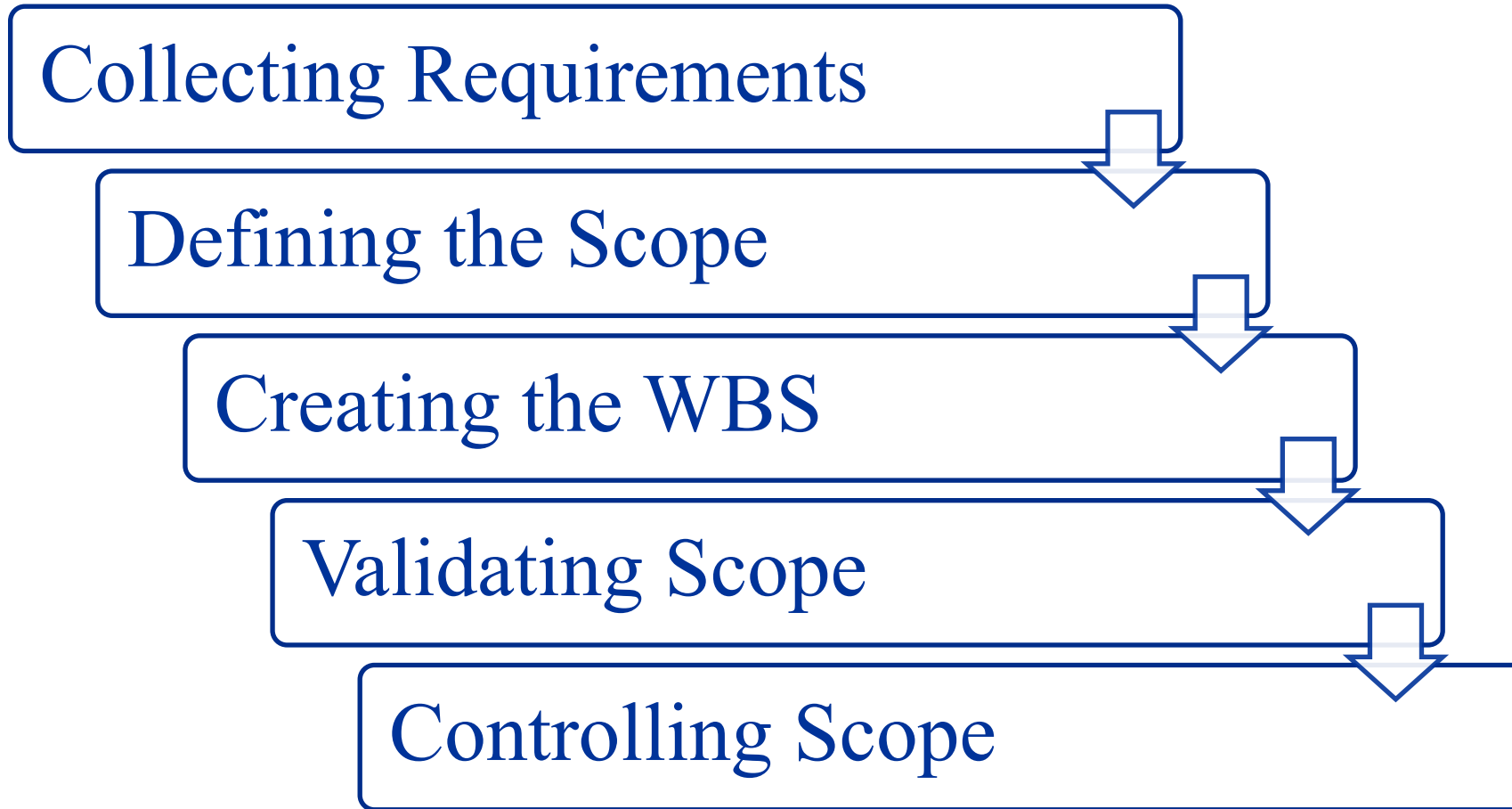
Validating Scope

- Scope validation involves formal acceptance of the completed project deliverables.
- This acceptance is often achieved by
 - a customer inspection;
 - a customer sign-off on key deliverables.

Controlling Scope

- Change is inevitable on projects, especially changes to the scope of IT projects.
- Users often are not sure
 - How they want screens to look;
 - What functionality they will need;
 - ...
- You cannot do a good job of controlling scope if you do not **first do a good job of** collecting requirements, defining scope, and validating scope.

Recap: Project Scope Management



Software Development Life Cycle (SDLC)

