Software development process

Jana Kostičová, 20.9.2023

- SDLC - systems development life cycle
- A set of activities necessary to develop a software
  - Order, repetition, inputs, outputs, ….

Phases of software development

1. *Business analysis*
2. Requirements
3. Architecture and design
4. Implementation
5. Verification and validation
6. Maintenance

➔ Each phase builds on the results of <u>all</u> previous phases
➔ Phases does not need to run sequentially, they may be combined or overlap
➔ Phases vs. roles

Phases of software development

1. *Business analysis*
   - **Why** the system should be developed (changed)
2. Requirements
   - **What** the (new / changed) system should do
3. Architecture and design
   - **How** the (new / changed) system should do it

Describing the system to be developed / changed

4. Implementation

Construction

5. Verification and validation
6. Maintenance

Post-construction phases

Business analysis

WHY the IT system should be developed / changed

BA identifies changes within the organisation which are necessary to achieve strategic goals of the organisation.

- Overlaps with requirements analysis
- Changes are realized through projects (both IT and non-IT)

Strategic goals
of the organisation
(2-5 years perspective)

↑ Necessary
to achieve

Changes
in strategy, organisation structure,
policies, processes, IT, …

- Changes are realized by projects
- Initial project documentation should include
  - Goal and purpose
  - Scope
  - Key stakeholders
  - Business case
  - …

Business analysis - example

Company: "Go Outdoors"
- Brick-and-mortar retail business
- 10 retail stores around SK & CZ

Strategy goals:
- GOAL 1: Implement a new online sales channel that generates a 50% increase in sales in the next 5 years
- GOAL 2: Open 3 new locations within the next 3 years
- …

Changes needed (GOAL 1):
- IT: Build ecommerce store (supplier)
- Organization:
  - Hire new people to marketing & sales dept, IT dept.
  - Create new SEO team
- Also changes in business processes

=> New project initiation: "Ecommerce store"

Business analysis

Requirements

Architecture & design

Implementation

Verification & validation

Maintenance

Requirements

WHAT the (new / changed) system should do

= Discovering, documenting, and maintaining a set of requirements for an IT system

Input
- Initial documentation of given IT project

Output
- Requirements specification
  - Text / models / diagrams / schemas /…

This phase is also known as "Requirements analysis", "Requirements engineering", "IT business analysis" or "Analysis" only

Business analysis

Requirements

Architecture & design

Implementation

Verification & validation

Maintenance

Requirements - example

Project "Ecommerce store" - example requirements:

- User shall be able to put one or more items into their shopping basket.
- User shall see price for each item.
- User shall be able to pay for items in shopping basket by credit or debit card.
- System shall be able to handle 10K sessions simultaneously.
- ….

Business analysis

Requirements

Architecture & design

Implementation

Verification & validation

Maintenance

Architecture and design

HOW the (new / changed) system should work

Architecture provides a high-level arrangement of components or modules. **Design** provides a detail arrangement within an individual component or module.

Input
- Initial documents of an IT project
- Requirements specification

Output
- Architecture models & documentation
- Design models & documentation

Architecture and design - examples

Project "Ecommerce store"

- System shall use client-server architecture
- System shall be implemented using MVC (model-view-controller) pattern
- System shall use the following logical data model - <u>E-R diagram attached</u>
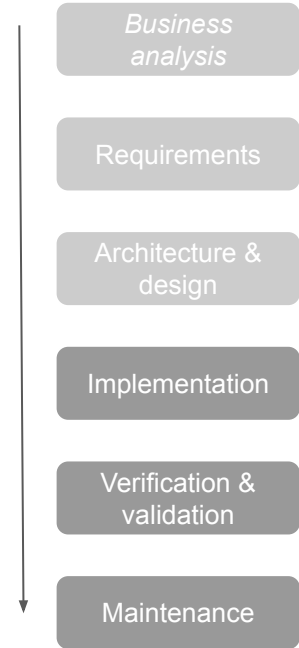- …

Construction & post-construction phases

- Implementation → Write the code

- Verification & validation → Test the system

  Not that simple, see PTS1

- Maintenance → Monitor & repair/upgrade the system

Business analysis

Requirements

Architecture & design

Implementation

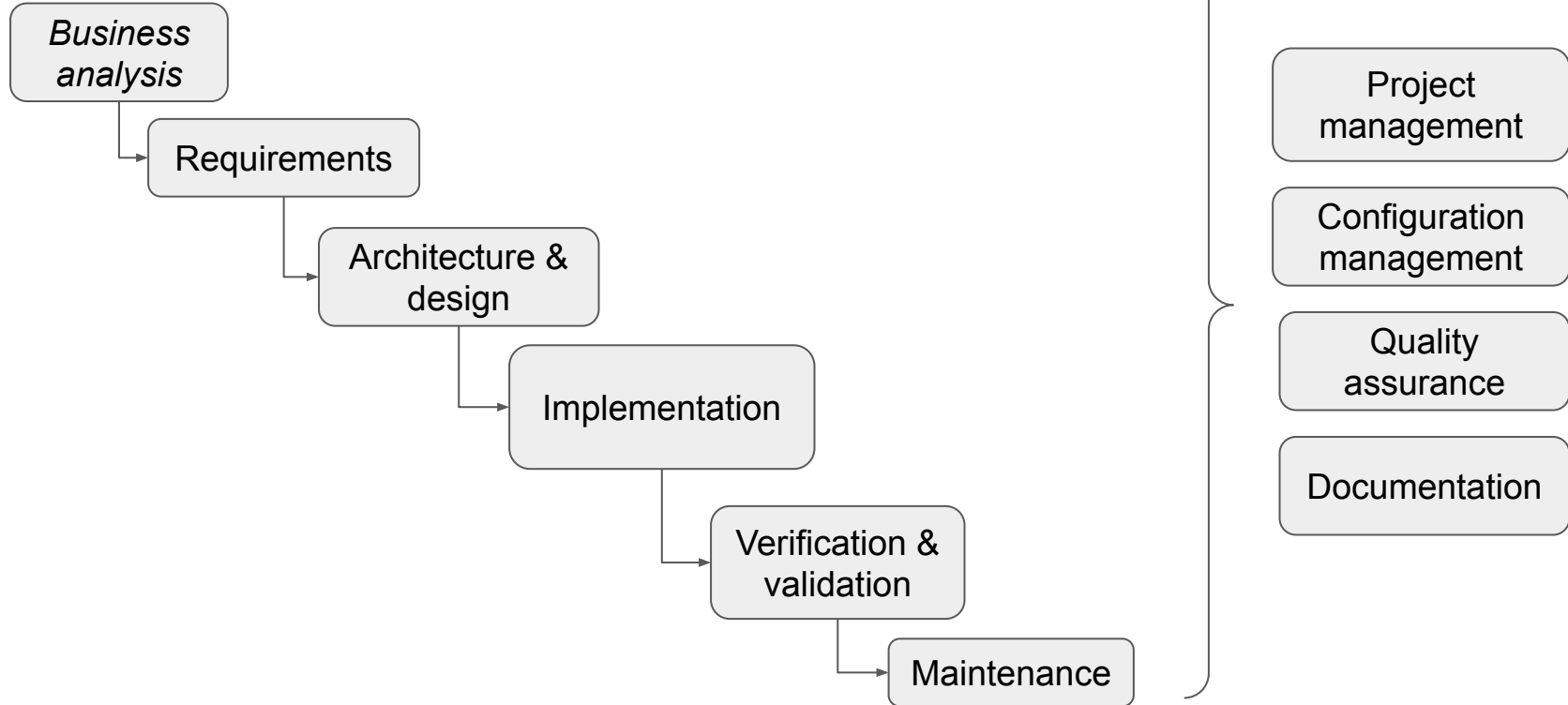Verification & validation

Maintenance

Supporting activities

- Project management
  - Planning (project plan, milestones, deadlines, resource allocation), risk management, project dependencies, facilitation, status reporting…
- Configuration management
  - Its purpose is to establish and maintain the integrity of all of the identified outputs of a project or process and make them available to concerned parties
- Quality assurance
  - Ensures that quality requirements will be fulfilled throughout the process
- Documentation
- …

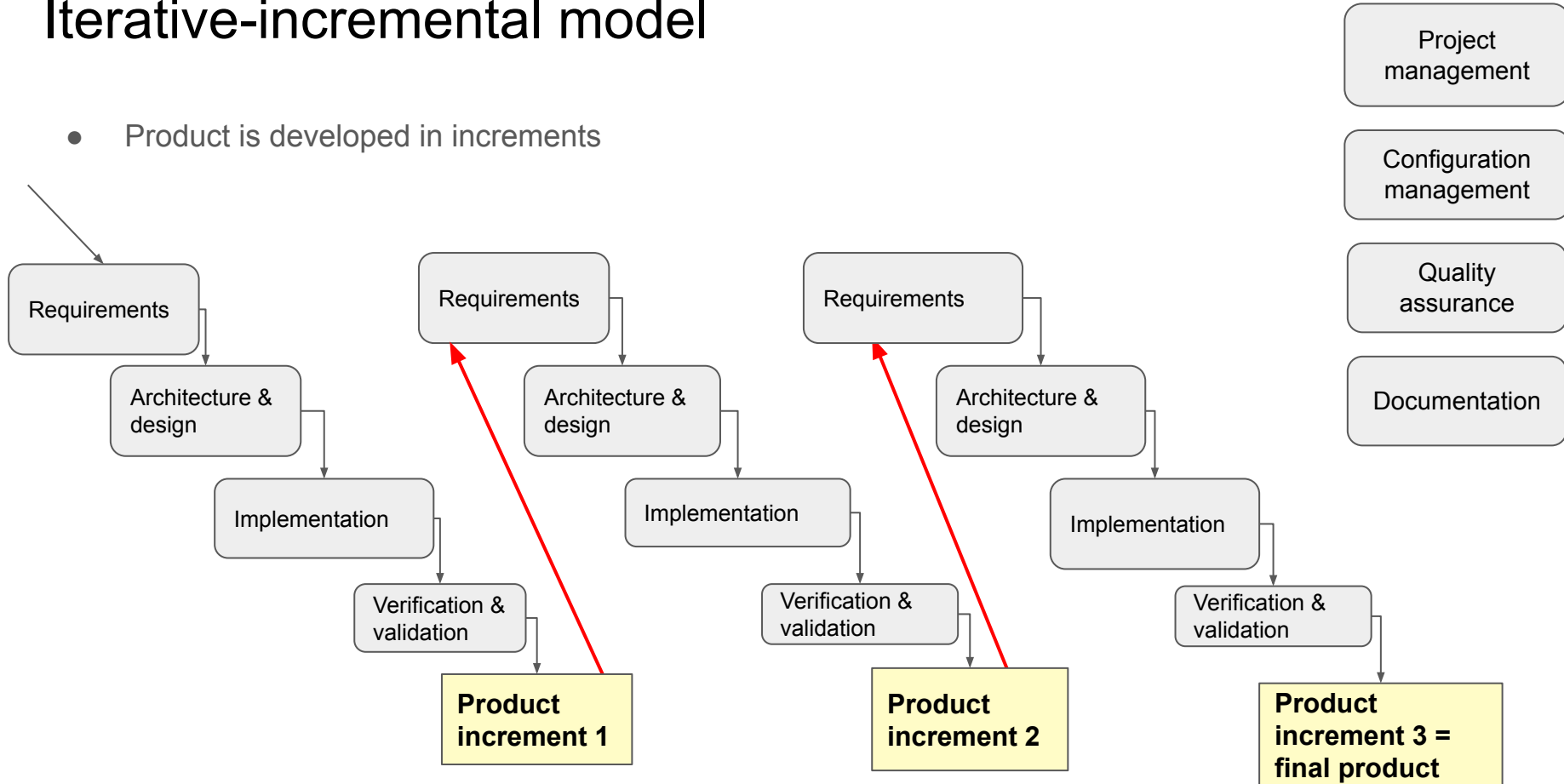# Models / methodologies / approaches

- Waterfall
- Iterative and incremental development
- Spiral development
- Rapid application development
- Extreme programming
- Agile
- Continuous integration
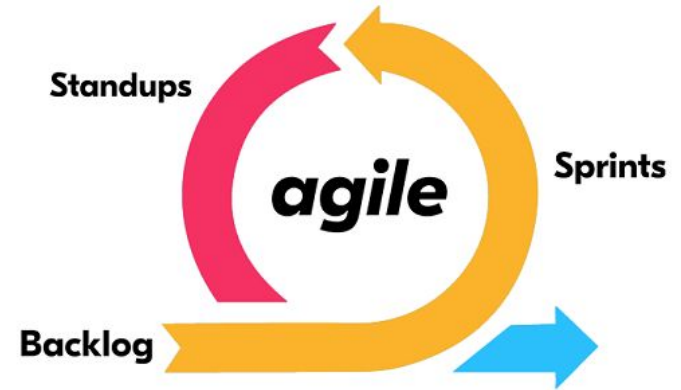- Lean management
- …

Waterfall model

Business analysis → Requirements → Architecture & design → Implementation → Verification & validation → Maintenance

Project management
Configuration management
Quality assurance
Documentation

Iterative-incremental model

- Product is developed in increments

Requirements

Architecture & design

Implementation

Verification & validation

Product increment 1

Requirements

Architecture & design

Implementation

Verification & validation

Product increment 2

Requirements

Architecture & design

Implementation

Verification & validation

Product increment 3 = final product

Project management

Configuration management

Quality assurance

Documentation

Agile methodologies

- Flexibility
- Collaboration
- Quality
- Continuous improvement

➔ Based on iterative-incremental model
➔ Solutions evolve through collaboration between self-organizing, cross-functional teams.
➔ ! Also agile approaches have their limitations.

Software development process - PTS1 vs PTS2

Phases

1. *Business analysis*
2. Requirements
3. ==Design== and architecture
4. ==Implementation==
5. ==Verification and validation==
6. Maintenance

Supporting activities

- Project management
- ==Configuration management==
- ==Quality assurance==
- Documentation

PTS1: ==yellow==
PTS2: everything else

References & further reading

- Ian Sommerville: Software Engineering 10th Edition
- Software Engineering Body of Knowledge (SWEBOK)
- Robert Lukoťka: Software development process, Software contracts