CSCI 2132: Software Development

Software Development Life Cycle

Norbert Zeh

Faculty of Computer Science
Dalhousie University

Winter 2019

Software Development Life Cycle (SDLC)

General term that describes structure imposed on the development of a software product
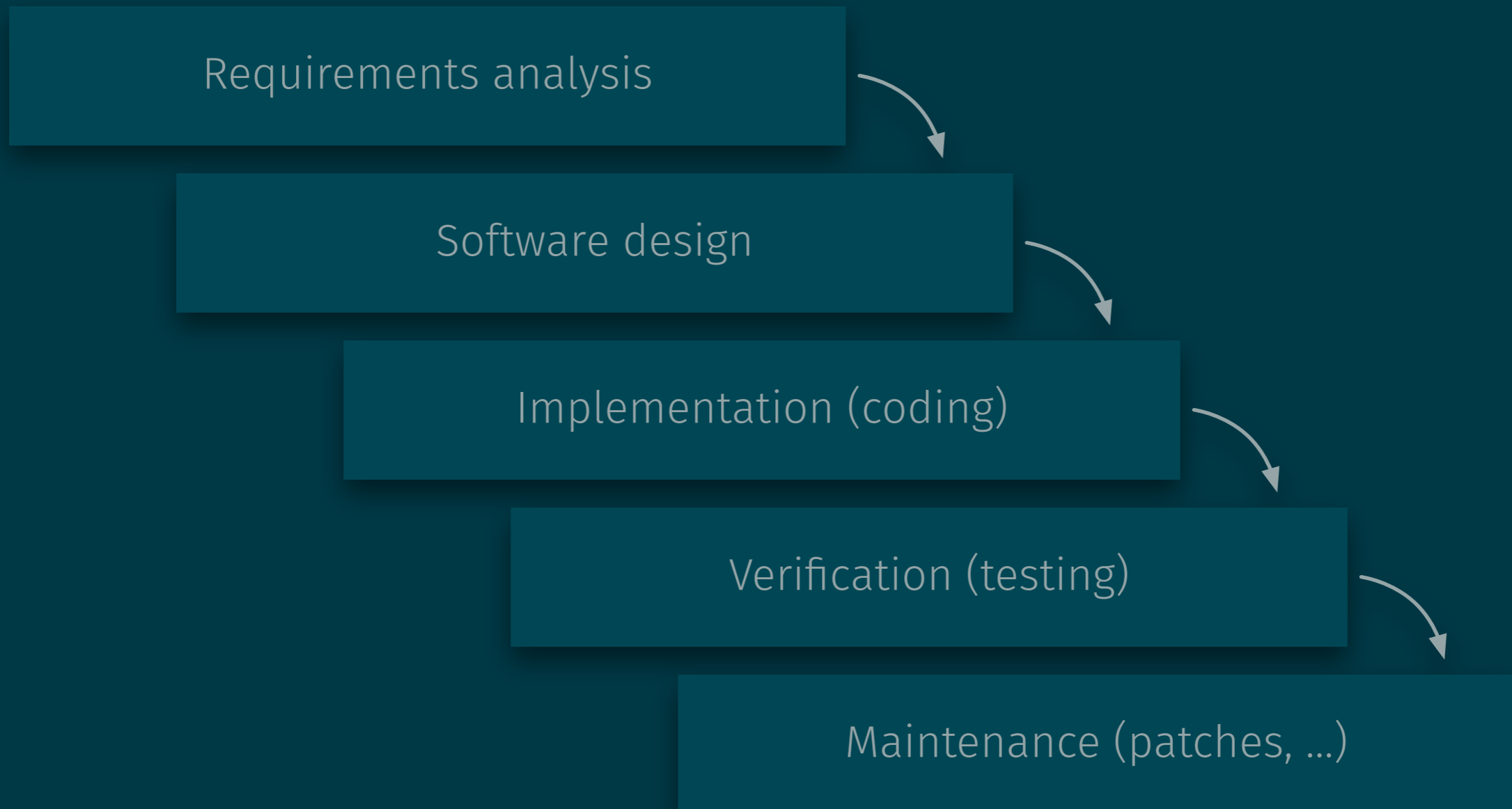
Purpose:

- Manage project cost
- Reduce risk of missing a deadline
- Ensure product quality
- Prevent "scope creep"

Many models to organize the SDLC have been proposed.

Waterfall Model

Sequential design process

Requirements analysis

Software design

Implementation (coding)

Verification (testing)

Maintenance (patches, ...)
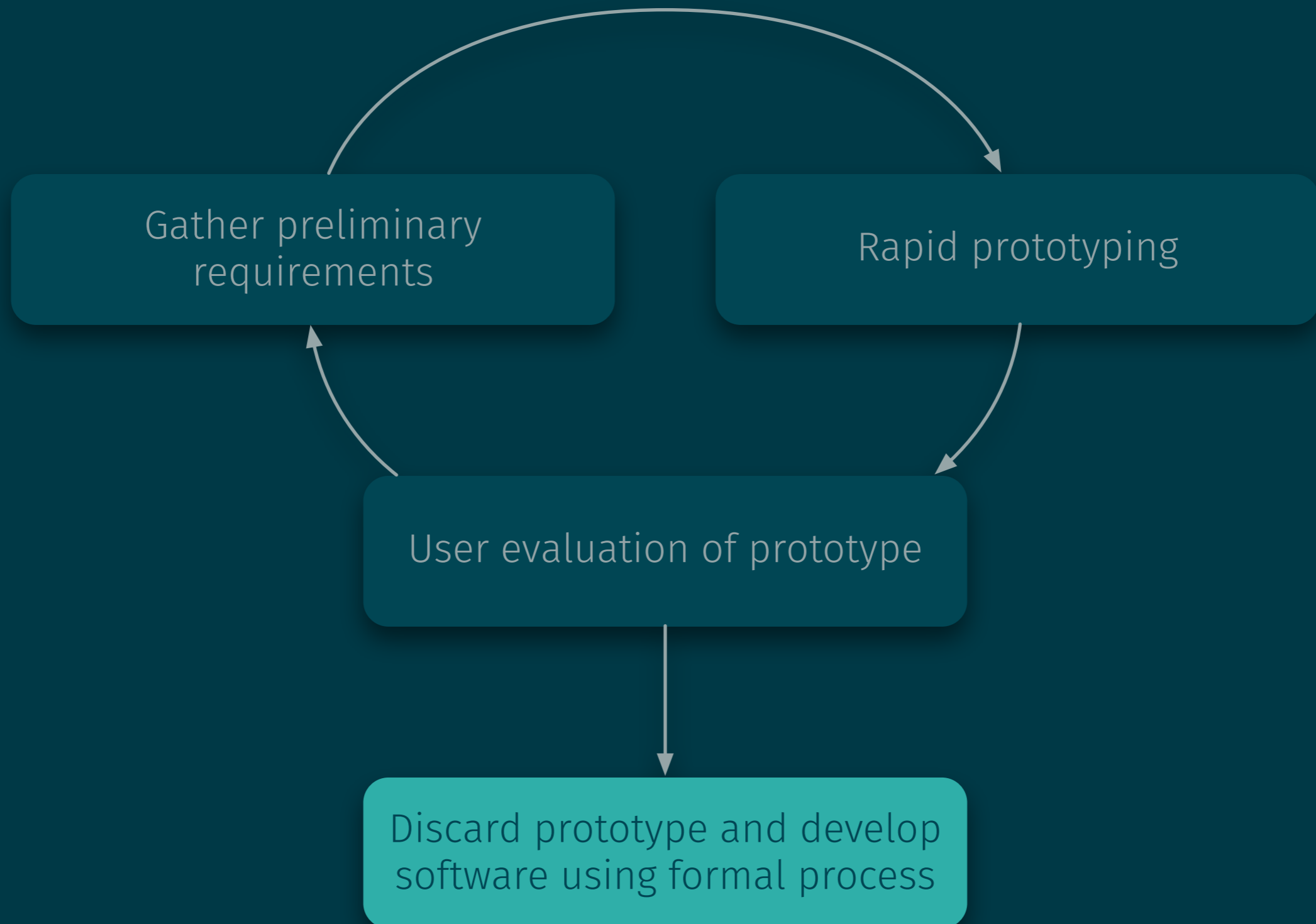
Pros and Cons of Waterfall Model

Pros:

- Natural and easy to understand
- Widely used
- Reinforces the notion of "design before coding"
- Clear milestones

Cons:

- Often not practical (too rigid)
- Clients may change the requirements
- Designers may not be aware of implementation challenges (even if they are the same person)

Rapid Prototyping Model

Gather preliminary requirements

Rapid prototyping

User evaluation of prototype

Discard prototype and develop software using formal process

Pros and Cons of Rapid Prototyping

Pros:

- Ensures that software product meets client's requirements

- Reduce time and cost if client requests changes during the process

Cons:

- Adequate and appropriate user involvement may not always be possible

- Cost of prototype development

- Developer attachment to prototype

More About Models

There are many models.

Studied in Software Engineering course (CSCI 3130)

Choose appropriate model depending on the particular software to be developed and client constraints.