University of Sheffield

Part 2: Restructuring - Intro

Software Reengineering

(COM3523 / COM6523)

Dr Donghwan Shin

The University of Sheffield

Join at

slido.com

#1146 901

What is Software Reengineering?

Two pillars of software reengineering

Part 1

Understanding

Legacy Software

Part 2

Restructuring

Legacy Software

What you have learned so far …

Part 1

Understanding

Legacy Software

Static analysis

Dynamic analysis

Repository analysis

…

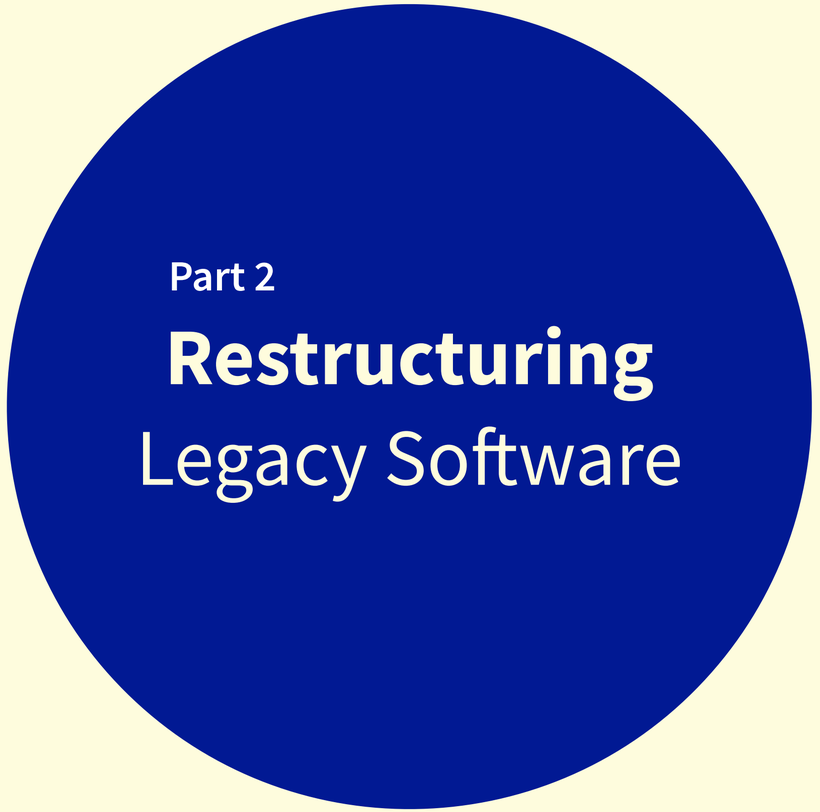
What you will learn from now on …

Testing

Responsibilities

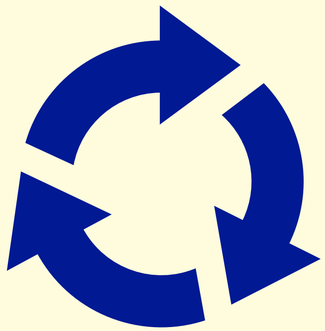
Code clones

Dead code



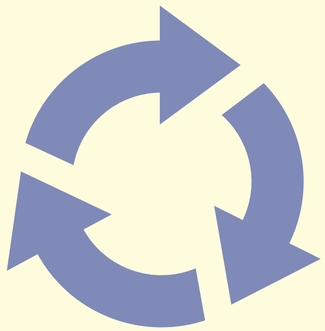
The four main topics in restructuring legacy software

The four main topics in restructuring legacy software

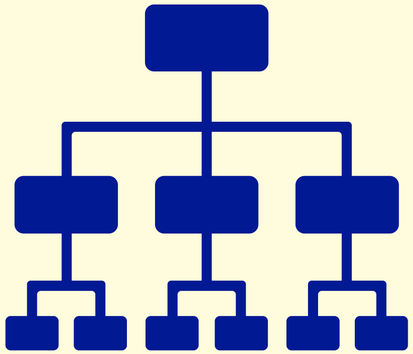


Testing for Reengineering

The four main topics in restructuring legacy software

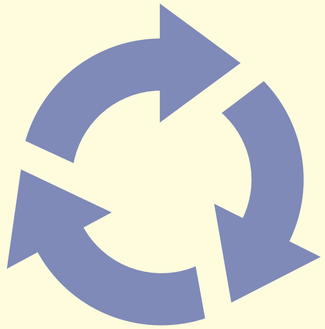


Testing for Reengineering

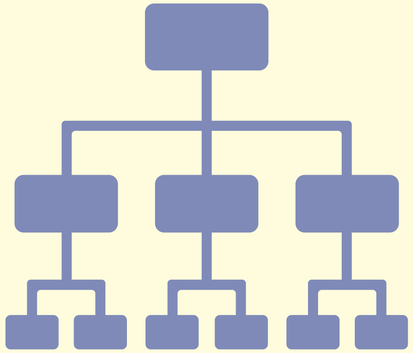


Redistribute Responsibilities

The four main topics in restructuring legacy software



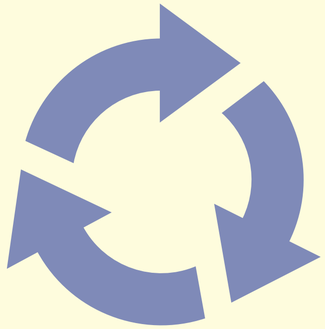
Testing for Reengineering



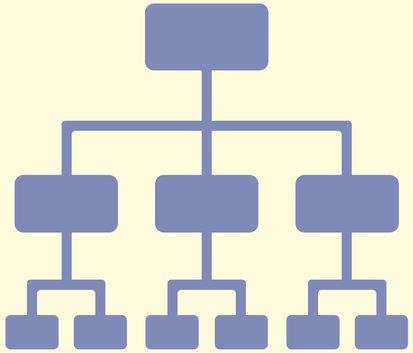
Redistribute Responsibilities

Manage Code Clones

The four main topics in restructuring legacy software



Testing for Reengineering



Redistribute Responsibilities

Manage Code Clones

Remove Dead Code

One more thing!

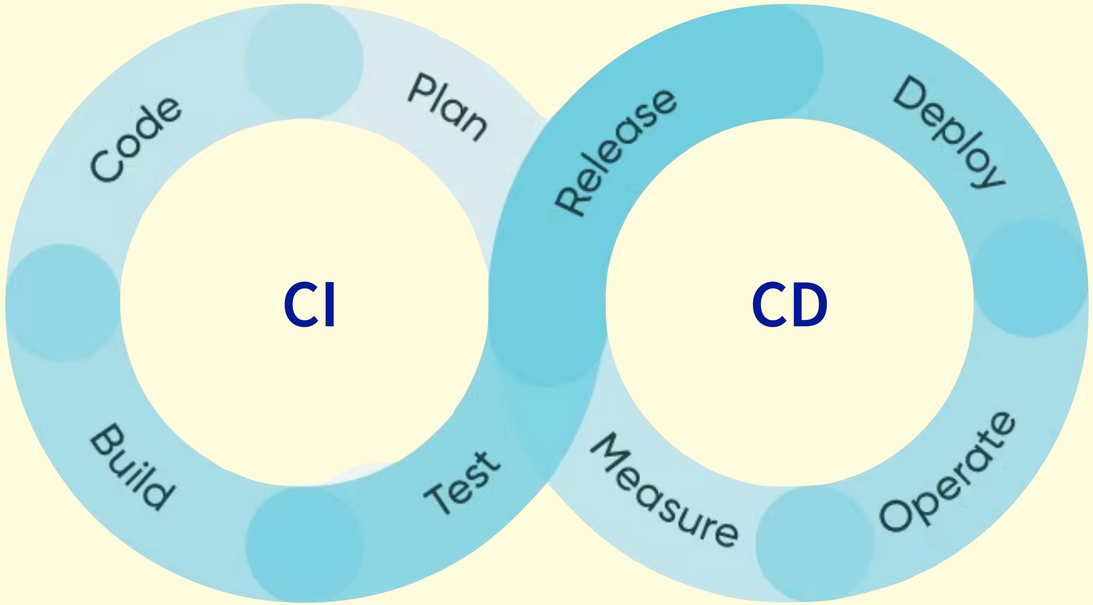


Image from https://www.servicenow.com/uk/products/devops/what-is-cicd.html

Learning objectives for Part 2

To learn frequent ***reengineering problems*** and ***solutions*** as "***patterns***".

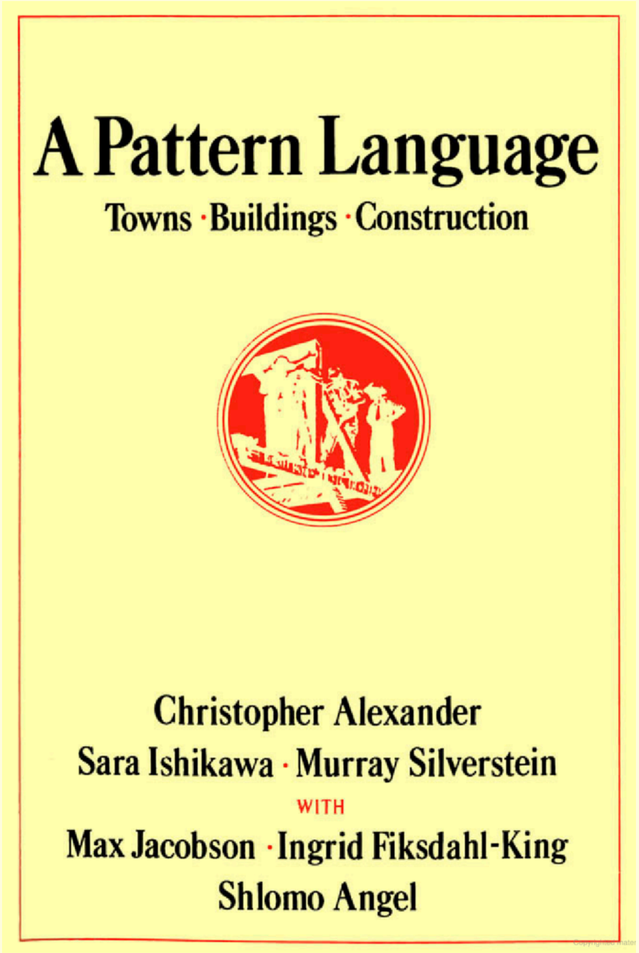
To understand the ***objectives*** and ***tradeoffs*** for individual reengineering patterns.

To ***restructure*** legacy software with educated decisions.

What are “patterns”?

What are “patterns”?

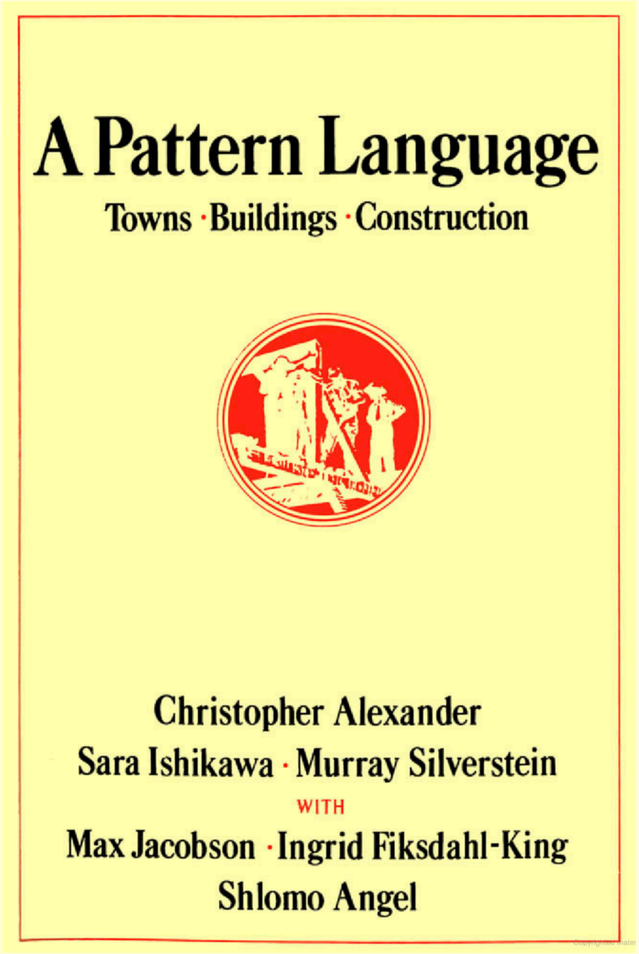
- Architect Christopher Alexander and his colleagues introduced ***patterns*** as a literary form in his book.



Alexander, Christopher. A pattern language: towns, buildings, construction. Oxford university press, 1977.

What are “patterns”?

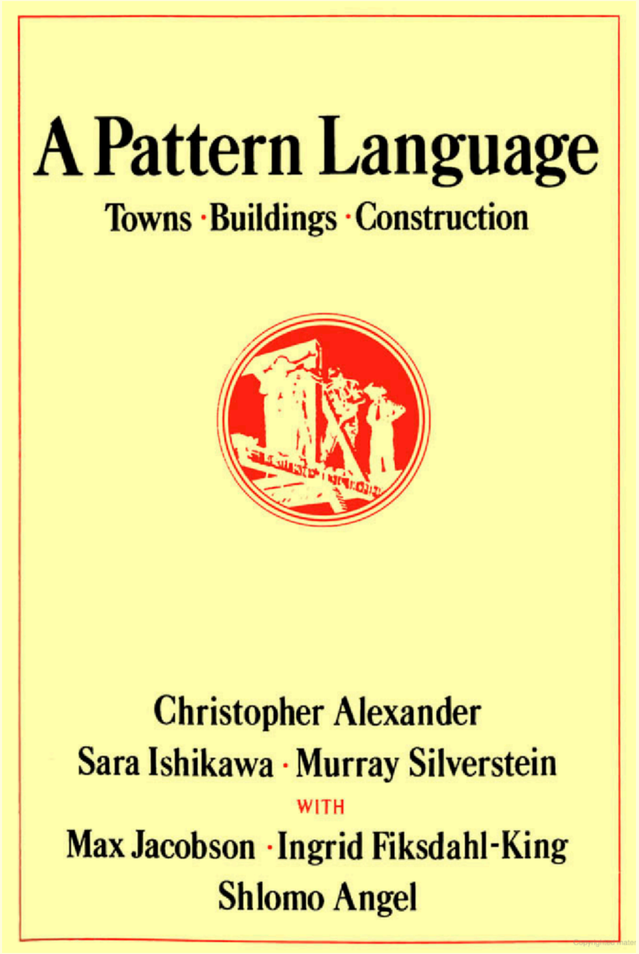
- Architect Christopher Alexander and his colleagues introduced ***patterns*** as a literary form in his book.
- They present a systematic method for architecting various **physical structures**, from rooms to buildings and towns.



Alexander, Christopher. A pattern language: towns, buildings, construction. Oxford university press, 1977.

What are “patterns”?

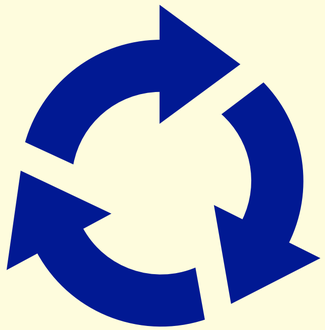
- Architect Christopher Alexander and his colleagues introduced ***patterns*** as a literary form in his book.
- They present a systematic method for architecting various **physical structures**, from rooms to buildings and towns.
- They introduced patterns to discuss ***issues*** and ***tradeoffs***.



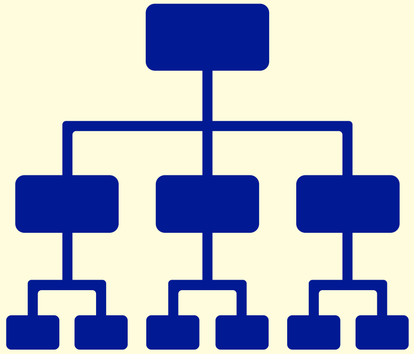
Alexander, Christopher. A pattern language: towns, buildings, construction. Oxford university press, 1977.

Reengineering patterns

- Let us discuss ***recurring issues*** and ***tradeoffs*** in reengineering legacy systems.
- Help in diagnosing ***problems***, identifying ***weaknesses***, and finding ***solutions***.



Testing for Reengineering



Redistribute Responsibilities

Manage Code Clones

Remove Dead Code

For each reengineering pattern, you will learn …

For each reengineering pattern, you will learn …

1. Problem

For each reengineering pattern, you will learn …

1. Problem
2. Solution

For each reengineering pattern, you will learn …

1. Problem
2. Solution
3. Related techniques

For each reengineering pattern, you will learn …

1. Problem
2. Solution
3. Related techniques
4. Examples

For each reengineering pattern, you will learn …

1. Problem
2. Solution
3. Related techniques
4. Examples
5. Tradeoffs (pros, cons, difficulties)

For each reengineering pattern, you will learn …

1. Problem
2. Solution
3. Related techniques
4. Examples
5. Tradeoffs (pros, cons, difficulties)
6. Advanced topics

How are reengineering patterns different from refactoring?

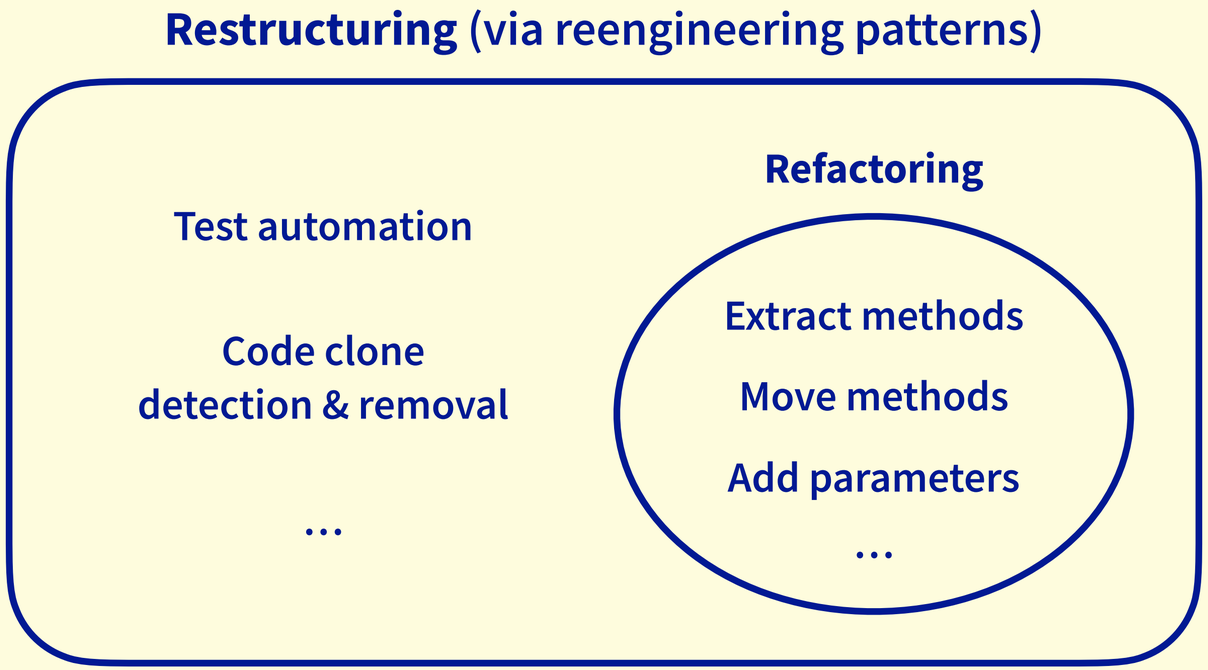
How are reengineering patterns different from refactoring?

Refactoring is a technique to resolve specific reengineering patterns.

How are reengineering patterns different from refactoring?

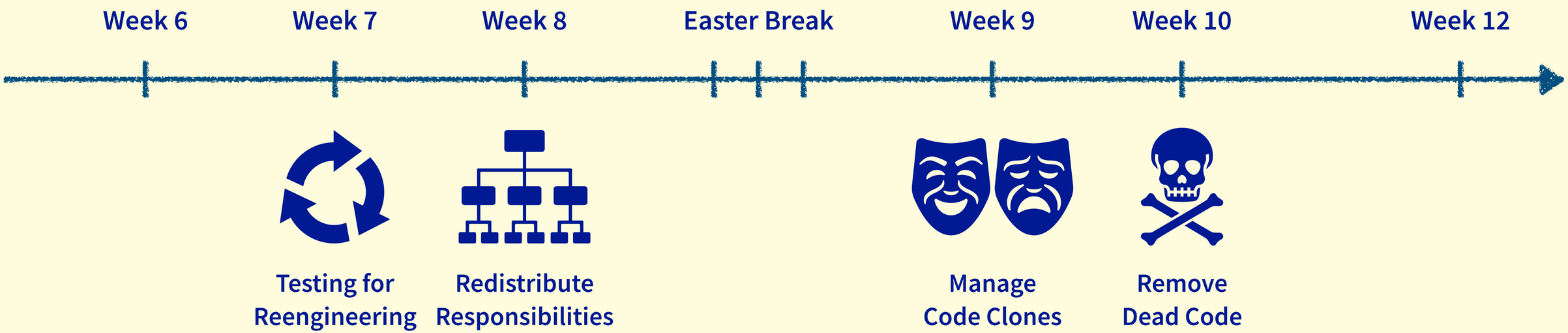
Refactoring is a technique to resolve specific reengineering patterns.

Reengineering patterns entail more than refactoring (e.g., test automation).

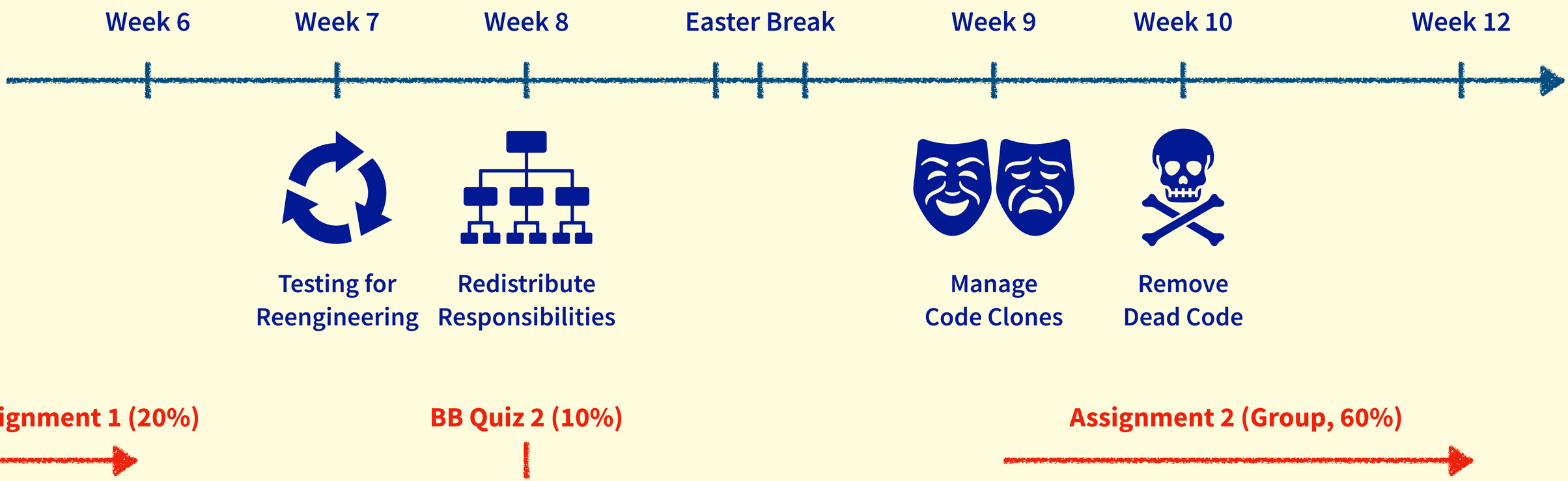


Our journey will be …

Our journey will be …



Our journey will be …



Weekly lab sessions

You will get hands-on experience with the corresponding lecture materials.

Each session will have a live demonstration (by an instructor) followed by lab exercises.

The answers for the lab sessions will be available by the end of the week.

Where can I get support?

Where can I get support?

- Sli.do

Where can I get support?

- Sli.do
- Your group

Where can I get support?

- Sli.do
- Your group
- Blackboard discussion forum

Where can I get support?

- Sli.do
- Your group
- Blackboard discussion forum
- Office hours (available on Blackboard - Staff Contacts)

Where can I get support?

- Sli.do
- Your group
- Blackboard discussion forum
- Office hours (available on Blackboard - Staff Contacts)
  - Please only send us emails directly as a last resort.

Reading material

Reading material

- The following book covers the main concepts of the restructuring part:
  - *“Object Oriented Reengineering Patterns”* by Serge Demeyer, Stéphane Ducasse, and Oscar Nierstratz.
  - The book is available as a **free** e-book here: http://scg.unibe.ch/download/oorp/

Reading material

- The following book covers the main concepts of the restructuring part:
  - *"Object Oriented Reengineering Patterns"* by Serge Demeyer, Stéphane Ducasse, and Oscar Nierstratz.
  - The book is available as a **free** e-book here: http://scg.unibe.ch/download/oorp/
- Additionally, each lecture will include relevant papers or chapters from books.
  - Check out the "**Reading List (click here)**" on Blackboard - all the links are there.

Any questions?

You can use the Sli.do Q&A if you prefer!