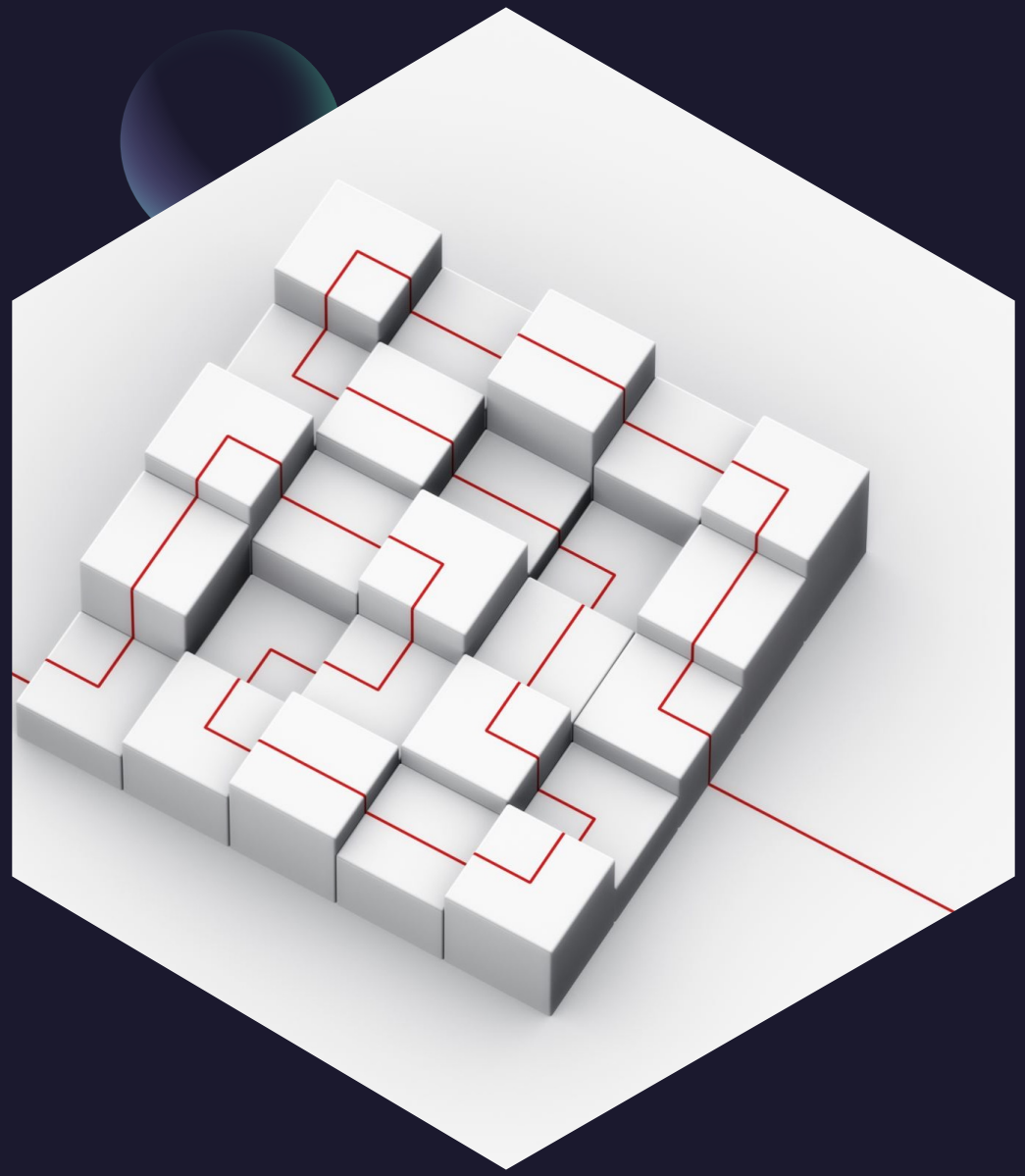
De la programmation par objets aux Structures de Données et Algorithmique

From OOP to Data Structures and Algorithmics



Mireille Blay-Fornarino

Université Nice Sophia Antipolis | POLYTECH NICE-SOPHIA

Membre de UNIVERSITÉ CÔTE D'AZUR

Introduction

Software Development Life Cycle (SDLC)

Tests	Design	Build
You will study the tests in PS5; you will have to use them systematically in all our developments. We will complete them by checking the complexity of the codes and their "quality."	You will gain design skills by working on the organization of your software, learning how to assess a specification.	You will learn to design and write code by mastering specific tools and techniques. It's the essential part of the acquisition targeted by this teaching.

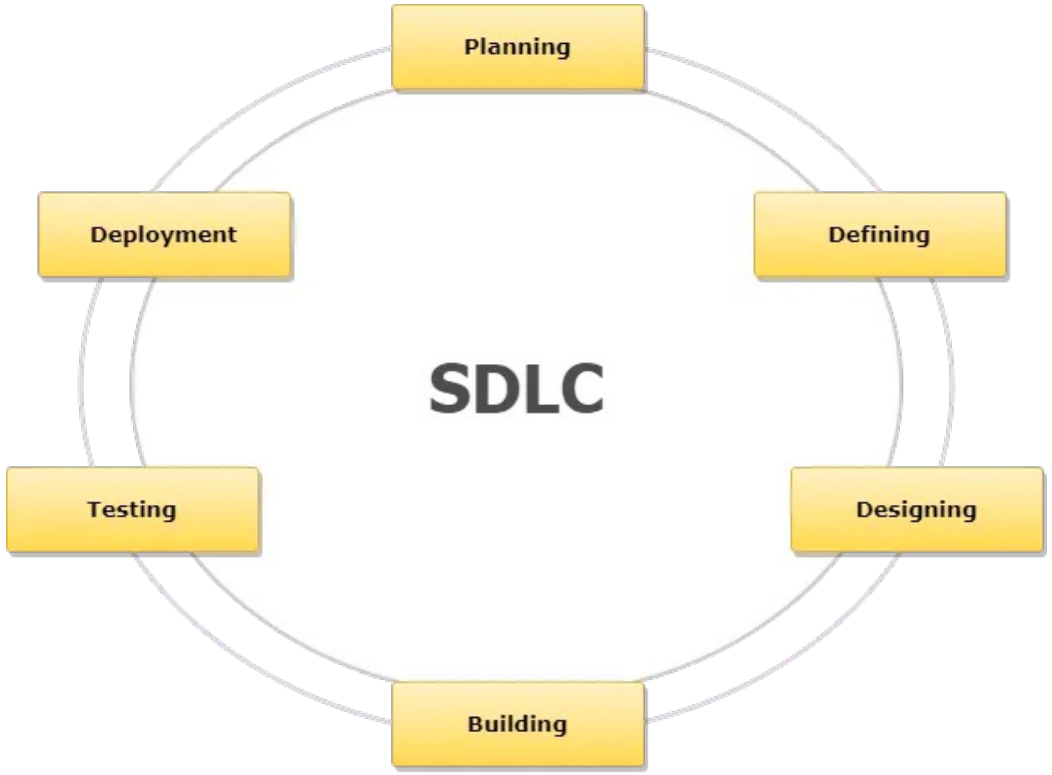


Figure: Software Development Life Cycle Stages

https://www.w3schools.in/sdlc/software-development-life-cycle-sdlc

Software design

« Software design (the process) is

- *the construction **of abstractions of data** and **computation** and*
- *the **organization of these abstractions** into a working software application*.

…

In practice, the design process is essentially one of **decision making**.

Should we use a list or a stack?

What services should this interface offer?

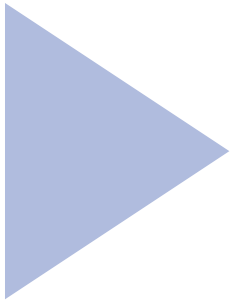
Where should this error be handled? »

Robillard, M. P. (2019) Introduction to Software Design with Java. Cham: Springer International Publishing. doi: 10.1007/978-3-030-24094-3

Goals

Be able **to make good design choices** as a developer (project manager, etc).

- Reason in terms of the general abstractions that come up in all non-trivial software (and many non-software) systems



Be able to **justify and communicate your design decisions**

- Key abstractions used almost every day in just about anything related to computing and software

Assumed Background

- Fundamentals of computer science and object oriented programming – Variables, conditionals, loops, methods, fundamentals of defining classes, arrays, …

Organisation du module

Eco-système

Semestre 1 :

- **De la POO aux Structures de Données et Algorithmique** : Basics of software design
- PS5 Project: Basics of Software Development Management

Semestre 2 :

- Algorithmiques et Structures de données : POO en action (S6)
- PS6 Project : Producing software for humans: from specifications to tests and vice versa, impacts on software architecture.

Environnement de travail

Java 17

IntelliJ et ses plugins

- sonarLint

A installer avant le TD
https://www.jetbrains.com/community/education/#students

Organisation (non figée)

- 1h de cours par semaine
- 2h30 de TDs
  - Un quizz en début de séance de 5 à 10 minutes
  - Parfois un TD noté en séance
- Un travail personnel est attendu pour terminer les TDs en plus de comprendre le cours évidemment.

Matériel de cours

- Tous les documents de cours et de Tds seront accessibles à partir de moodle* (parfois pour des renvois vers des documents/codes extérieurs)
- Il s'agit d'aides visuelles, mais pas d’une description complète
- **Livres**:
  - Robillard, M. P. (2019) Introduction to Software Design with Java.
    - Avancé
  - Cormen, Leiserson, Rivest, Stein
    - Bonne lecture, mais assez détaillée et complète

  Les références utilisées dans le cours vous sont données au fur et à mesure.

* Nommage des modules non mis à jour… problème administratif.

Evaluation

- Quizz, Tests et Rendus : Contrôles réalisés en cours ou TD et Rendus de devoir (60%)
- DS : Contrôle(s) réalisé(s) individuellement (40%)