

# Génie logiciel / Software engineering



Composante  
Polytech  
Grenoble - INP,  
UGA

- > **Langue(s) d'enseignement:** Français
- > **Ouvert aux étudiants en échange:** Oui
- > **Code d'export Apogée:** KARI8M03

## Présentation

### Description

Objectifs : introduction au génie logiciel

Compétences visées :

- connaître, savoir utiliser les processus logiciels pour conduire un projet de développement logiciel
- connaître, savoir utiliser les outils pour la production de logiciel : gestion de version, gestion automatique des "builds", forges, ...
- resituer le test dans un cycle de développement par rapport aux exigences
- connaître et savoir concevoir des logiciels au moyen de la notation UML 2.0

\* Introduction au Génie Logiciel

\* Processus logiciels : itératif, cascade,

V, incrémental, spirale, agile

\* Activités du Génie Logiciel : gestion des exigences, spécification, conception, implantation, validation, intégration, déploiement, maintenance, évolution

\* Conception avec la notation UML 2.0

\* Outils collaboratifs de production de logiciels : gestion automatique des builds (Ant, Maven), gestions de version (SVN, Git), Suivi de bugs, Tests (BDD, ...), intégration continue, forges, analyse de code (métriques logicielles)

\* Économie du logiciel et de l'open-source

\* Cas d'étude avec UML 2.0 et SCRUM

This is an introductory lecture in Software Engineering.

- \* Introduction to Software Engineering (SE)
- \* SE Processes : iterative, waterfall, V, incremental, spiral, agile
- \* SE Activities : requirements, specification, design, implementation, validation, integration, deployment, maintenance, evolution
- \* Design with the UML 2.0 notation
- \* Collaborative tools for software production: build managers (Ant, Maven), version control (SVN, Git), bugs trackers, tests (BDD, ...), continuous integration, forges, code analysis (software metrics)
- \* Software and open-source economies
- \* Case studies with UML 2.0 and SCRUM

## Heures d'enseignement

Génie logiciel / Software engineering - CMTD

Cours magistral - Travaux dirigés

32h

## Pré-requis recommandés

Une expérience du développement logiciel en équipe.

An experience of software development in a team.

**Période :** Semestre 8

## Évaluation initiale / Session principale - Épreuves

Libellé	Nature de l'enseignement	Type d'évaluation	Nature de l'épreuve	Durée (en minutes)	Nombre d'épreuves	Coefficient de l'épreuve	Remarques
						25/100	

## Bibliographie

- \* Ian Sommerville, Software Engineering (9th Edition), Pearson Pub., 2011, <http://www.cs.st-andrews.ac.uk/~ifs/Books/SE9/>
- \* Armando Fox and David Patterson, Engineering Software as a Service: An Agile Approach Using Cloud Computing, <http://beta.saasbook.info/courses>
- \* Eric Ries, The Lean Startup, <http://theleanstartup.com/>

# Infos pratiques

---

## Lieu(x) ville

› Grenoble

---

## Campus

› Grenoble - Saint-Martin d'Hères