

1IN3000 - Coding Weeks

Responsables : **Paolo BALLARINI** , **Marc-Antoine WEISSER**

Département de rattachement : **DÉPARTEMENT INFORMATIQUE**

Langues d'enseignement : **ANGLAIS , FRANCAIS**

Campus où le cours est proposé : **CAMPUS DE PARIS - SACLAY , CAMPUS DE METZ , CAMPUS DE RENNES**

Nombre d'heures d'études élèves (HEE) : **80**

Nombre d'heures présentielles d'enseignement (HPE) : **48**

Année académique : **2024-2025**

Niveau avancé : **non**

Présentation, objectifs généraux du cours :

Cette activité a pour objectif, au travers de la réalisation de projets de développement informatique, de permettre aux étudiants de consolider leurs connaissances en programmation et développement d'applications informatiques d'une part, et d'autre part de s'initier aux pratiques et aux méthodologies du développement informatique à plusieurs et en mode projet. Il s'agira aussi au travers de cette activité de s'initier et d'acquérir, par la pratique essentiellement les fondamentaux de la culture et de la qualité du code informatique («software craftsmanship »).

Période(s) du cours (n° de séquence ou hors séquence) :

Semaine intercalaire à l'issue de la SG1

Prérequis :

- Systèmes d'Information et programmation (SIP)

Plan détaillé du cours (contenu) :

Cette activité est organisée en deux semaines consécutives de développement informatique en mode « bootcamp ».

Lors d'une première partie (3 jours), les étudiants travailleront par groupes de 4 à 5 et en binômes au développement d'un projet type organisé en différentes itérations progressives (sprints). Les binômes pourront éventuellement être renouvelés chaque jour (i.e. « pair programming, » une des méthodes de développement informatique considérée comme une bonne pratique de l'«extreme programming »). De courts apports méthodologiques et techniques seront aussi apportés sous forme de conférences et/ou mini-tutoriels (gestionnaires de versions, Test Driven Development et qualité du code, agilité) et mis en oeuvre lors du développement de ce projet type. Cette première partie donnera lieu à une évaluation prenant en compte un contrôle des connaissances des concepts introduits lors de cette première partie, un contrôle de présence et une note de participation.

Lors de la deuxième partie (5 jours), les étudiants travailleront par groupe de 4 à 5 sur un projet

thématique avec comme objectif de passer de l'idée au produit. Il s'agira de mettre en oeuvre les méthodologies et pratiques de développement informatiques acquises précédemment. Ce projet permettra aussi l'apprentissage du travail en groupe. Ce projet donnera lieu à une évaluation par groupe par un jury lors d'une soutenance.

Déroulement, organisation du cours :

Cette activité s'organise en deux semaines consécutives.

=====

Semaine 1 – Jour 1 : 1h30 amphi (présentation + apport de compétences) + travail en EL/TP en groupe

Semaine 1 – Jour 2 : travail en EL/TP en groupe

Semaine 1 – Jour 3 : travail en EL/TP en groupe

Semaine 1 – Jour 4 : travail sur le projet

=====

Semaine 2 – Jour 1 : travail sur le projet

Semaine 2 – Jour 2 : travail sur le projet

Semaine 2 – Jour 3 : travail sur le projet

Semaine 2 – Jour 4 : travail sur le projet - Soutenances finales la dernière demi-journée.

Organisation de l'évaluation :

- Les trois premiers acquis d'apprentissage seront évalués à la fois par le travail sur le projet type de la première partie et par le travail sur le projet plus libre de la partie 2 (soutenance - livraison et documentation du projet sur Gitlab ou github). Le quatrième acquis d'apprentissage sera plus spécifiquement évalué lors du projet de semaine 2.

Moyens :

- Equipe Enseignante : Paolo Ballarini (responsable), Marc-Antoine Weisser (responsable) Lina Yé, Gianluca Quercini, Wassila Ouerdane, Myriam Tami et une équipe de vacataires.
- Taille des groupes : de 25 à 50 élèves.
- Langages de programmation : majoritairement python et ses nombreux modules et selon les groupes : java, unity ...
- Outils logiciels : VisualStudioCode, git et selon les groupes Android Studio, Unity,...

Acquis d'apprentissage visés dans le cours :

A la fin de cet enseignement, l'élève sera capable de :

- Connaître et appliquer les bonnes pratiques de programmation et méthodes du développement informatique à l'échelle d'un projet simple : structurer un ensemble de programmes de manière claire et sensée, utiliser des bibliothèques ou des modules existants, construire un projet de façon modulaire, tester et permettre la lisibilité, la compréhension et l'utilisation du code source du projet (qualité logicielle).
- Savoir construire un projet informatique dans une démarche itérative (de l'idée au produit) et collaborative.
- Connaître et savoir utiliser les principaux outils du développement informatique.
- Travailler en équipe/en collaboration.

Description des compétences acquises à l'issue du cours

:

- This course will provide a first milestone in competency **C6: " Be operational, responsible and innovative in the digital world"**
- Knowing and applying good programming practices and methods of computer development at the scale of a simple project is covered in **C6.2: "Software Design" and C6.3: "Data Processing"**.
- Knowing how to build a computer project in an iterative (from idea to product) is part of **C6.1: "Solve a problem numerically"**
- Knowing and knowing how to use the main tools of software development is part of C6.2: "Software Design"
- Teamwork/collaboration is part of **C8: "Leading a project, a team"**.
- Skills **C1.1, C3 and C7** will also be mobilized.

Bibliographie :

- Langues de programmation
 - Python
 - Think Python 2nd Edition - Allen B. Downey - GreenTeaPress.
 - Python Cookbook, 3rd Edition - Recipes for Mastering Python- B. Jones - D.Beazley - O'Reilly
 - Test-Driven Development With Python: Obey the Testing Goat- Harry J.W. Percival- O'Reilly
 - Autres : RealPython : <https://realpython.com/>
 - Java
 - Core Java Volume I—Fundamentals, Eleventh Edition - Cay S. Horstmann. Prentice Hall
- Culture code
 - Clean Code - A Handbook of Agile Software Craftmanship- Robert C. Martin
 - Apprenticeship Patterns: Guidance for the Aspiring Software Craftsman- Dave Hoover et Adewale Oshineye - O'Reilly
 - OCTO - Culture Code - <https://www.octo.com/publications/culture-code/>