



Syllabus Si6

Direction des Études

Cadrage

Ce document présente les éléments de contenus du semestre 6 @Epita. La formation est semestrialisée, chaque semestre contient à un nombre d'UEs (Unité d'Enseignement) qui contiennent chacune à leurs tours des ECUEs (Élément Constitutif d'une UE). Une UE est porteuse d'un nombre d'ECTS (European Credit Transfer System) qui reflète le temps de travail étudiant total sur l'UE. La correspondance admise est de 25--30 de travail étudiant par ECTS. Un semestre est valorisé de 30 ECTS, donc un total maximal de travail étudiant de 900h.

La liste des UE et ECUEs de tronc commun @Epita est listée en aval. Chaque UE a des objectifs d'apprentissages indiqués en tête de descriptif d'UE. Chacune participe à l'acquisition d'indicateurs de développement du référentiel compétences école, également indiqués dans le descriptif de chaque UE. Dans le descriptif d'une ECUE on retrouve

- un résumé des contenus abordés dans l'ECUE
- les AAVs (Acquis d'Apprentissages Visés) de l'ECUE
- le plan de cours de l'ECUE
- les pré-requis de l'ECUE
- les modalités de validation et le calcul de la note de l'ECUE
- la bibliographie associée au cours.
- le coefficient permettant de calculer la moyenne pondérée sur l'UE dont fait partie l'ECUE.

On trouve par ailleurs quelques éléments relatifs au cadrage temporel de l'ECUE et du référent de celle-ci. Parmi les éléments décrivant la ventilation des heures associées à l'ECUE

- VHT : Volume Horaire Total d'implication des étudiants sur l'ECUE; cela correspond au temps total estimé d'investissement des étudiants.
- FF : volume horaire de Face-à-Face.
- EQ-CM : nombre d'heures d'Equivalent Cours Magistraux, cela correspond au nombre d'heures d'enseignement magistral à effectué en équivalent des vidéos asynchrones mises à dispositions des étudiants.
- RMD : nombre d'heures de ReMédiation, les heures de face-à-face pédagogique au cours desquels un enseignant ou une enseignante apporte
 - des éclaircissements sur un contenu étudié préalablement en asynchrone
 - du contexte à un travail d'étude à venir ou déjà effectué
- TDP : les heures de TD, de TP ou d'activités interactives qui attendent une production de la part des étudiants.

Chaque UE a un coordonnateur attitré, chaque ECUE son référent cours. Ils sont à disposition des différents intervenants école pour répondre à leurs questions

Table des matières

[21_IS6TC_I12] Informatique industrielle 2.....	7
[TI] Technologie des interfaces	8
[IDVOC] Initiation au DevOps	9
[SEDE] Sécurité du Développement	10
[NET2] Éléments de réseaux 2	11
 [21_IS6TC_APR2] Ateliers et projets informatiques.....	12
[CPP] Atelier CPP	13
[JAVA] Atelier JAVA	15
[JS] Atelier JS	17
[TC] Projet TIGER	18
[JWS] Projet JAVA WEB SERVICES	20
[PING] Projet de fin d'ING1	21
 [21_IS6TC_MAT2] Mathématiques 2.....	23
[PBS1] Probabilités et statistiques 1	24
[CHIFR] Chiffrement et codes correcteurs	26
[CAMA] Calcul matriciel	27
 [21_IS6TC_IF2] Informatique fondamentale 2.....	29
[LOF0] Logique formelle	30
[TYLA] Typologie des langages de programmation	32
[THEG] Théorie des graphes	34
[MOB2] Modélisation objet 2	36

[21_IS6TC_IRT1] Innovation, recherche et approfondissement technique. .	38
[ERO1] Éléments de recherche opérationnelle 1	39
[TIFT] Électif technique	41
[TIFIR] Électif Exposition et initiation à la recherche	42
 [21_IS6TC_MSH2] Management et sciences humaines	 43
[ODP2C0] Développement des soft skills de collaboration	44
[LV] Langue vivante	46
[COMM2] Communication écrite	48
[CPT] Conception	50
[ETHIX] Éthique	52
 [21_IS6TC_CCE2] Culture et connaissances pour l'entreprise	 53
[MSST] MOOC Sécurité et santé au travail	54
[REPP] Projet professionnel	56
[MARK] Marketing	57
[GEPRO0] Gestion projet	59

[21_IS6TC_II2] Informatique industrielle 2

Semestre	S6
Crédits	4
VHT	60
CODE	21_IS6TC_II2
Coord.	Antoine Dumeige

Modalités de validation de l'UE

Validation de l'UE par une moyenne supérieure à 10 et une note supérieure à 6 dans chaque ECUE.

[TI] Technologie des interfaces

Niveau	L3
Semestre	SI6
UE	21_IS6TC_II2
Coeff UE	1/4
Référent	Christophe Eymard

EQ-CM	2
RMD	0
TDP	6
FF	8
VHT	8

Résumé et contexte

Cette activité a pour objectif d'introduire les interfaces graphiques de type "client lourd".

Acquis d'apprentissages visés (AAVs)

Au terme de cette activité, il est attendu d'un étudiant qu'il soit capable :

- De décrire les patrons de conception usuel d'une interface graphique (MVC & co, FRP)
- D'interagir avec le DOM efficacement
- De développer une interface minimaliste sur un service fourni par l'enseignant

Plan cours

- Rôle du client dans le développement logiciel.
- Design-patterns FRP, MVC, MVP et MVVM.
- Exemple synthétique de mise en œuvre / Travaux pratiques.

Format des activités

- a) Cours / Capsules vidéos
- b) Deux TP de trois heures, utilisant le framework ReactJS

Pré-requis

Connaissance fonctionnelle du langage JAVA

Évaluation des AAVs

L'activité est validée via une grille d'évaluation critériée sur l'interface réalisée pour le projet de synthèse PING.

Calcul de note finale

Une règle de trois sera appliquée sur la pondération de la grille pour ramener les notes à 20.

Références et bibliographie

TBA.

[IDV0] Initiation au DevOps

Niveau	L3	EQ-CM	6
Semestre	SI6	RMD	0
UE	21_IS6TC_II2	TDP	6
Coeff UE	1/4	FF	12
Référent	Cyril Duval	VHT	24

Résumé et contexte

Le module a pour but d'initier les étudiants aux méthodes et outils DevOps. L'enseignement à EPITA se concentre principalement sur les aspects de dev, et très peu sur l'ops. Ce cours est une ouverture vers une nouvelle branche, tout en fournissant des connaissances sur des outils très utilisés en industrie.

Acquis d'apprentissages visés (AAVs)

- YAML
- Docker:
 - fonctionnement
 - notion de container
 - run/exec/build
 - mount
 - network
- Gitlab-ci
- docker-compose

Plan cours

TBA.

Format des activités

6h de CM, 6H de TP avec projet individuel évalué et commencé pendant les TP.

Pré-requis

- Notions de bash
- Notions d'UNIX

Évaluation des AAVs

Projet individuel unique.

Calcul de note finale

100% sur le projet individuel.

Références et bibliographie

TBA.

[SEDE] Sécurité du Développement

Niveau	L3	EQ-CM	6
Semestre	SI6	RMD	0
UE	21_IS6TC_II2	TDP	6
Coeff UE	1/4	FF	12
Référent	Marc Espie	VHT	24

Résumé et contexte

Ce cours a pour but de permettre aux étudiants de connaître le cycle de développement d'un logiciel vis à vis de la sécurité de ce dernier. La manière dont il doit être conçu et les points de vigilance qui leur permettra d'adopter les bonnes pratiques.

Acquis d'apprentissages visés (AAVs)

Bonnes pratiques de développement dans des objectifs de sécurité

Connaissance des problèmes de sécurité les plus communs

Mesures palliatives de base dans les systèmes modernes

Plan cours

- cycle de développement logiciel, le point de vue de la sécurité
- trous classiques: buffer overflow, injection SQL, XSS...
- mesures palliatives des OS modernes
- bonnes pratiques

Format des activités

CM de 6h

Atelier de 6h "inversé" : les étudiants vont auditer leur propre code de projets précédents

Pré-requis

Validation de la piscine (CNIX)

Évaluation des AAVs

QCM moodle.

Calcul de note finale

100% QCM

Références et bibliographie

- Building secure software (Viega, McGraw, ISBN 0-201-72152-X)
- OpenBSD talks (<https://www.openbsd.org/papers>)

[NET2] Éléments de réseaux 2

Niveau	L3	EQ-CM	12
Semestre	SI6	RMD	0
UE	21_IS6TC_II2	TDP	6
Coeff UE	1/4	FF	18
Référent	Marc Schmitt	VHT	18

Résumé et contexte

Ce cours est la suite directe du cours NET1. Il se focalise sur les protocoles applicatifs et les paradigmes classiques du réseau en industrie.

Acquis d'apprentissages visés (AAVs)

À l'issue du cours, l'élève est censé maîtriser les protocoles applicatifs du modèle TCP/IP. Cela se traduit par les éléments suivants :

- Maîtrise des protocoles communément rencontrés sur le réseau
- L'élève possède les éléments conceptuels nécessaires pour appréhender un nouveau protocole
- L'élève sait architecturer un réseau basique d'entreprise
- L'élève sait décrire les axes d'évolution d'Internet

Plan cours

- Certificats X.509 et web moderne
- IPv6
- Couche physique
- Systèmes distribués
- Routage global
- Firewalling

Format des activités

Le cours s'organisera en 6 séances de 2h, plus un TP machine commun avec le cours IDVO. Chaque séance abordera l'un des points du plan cours.

Pré-requis

NET1

Évaluation des AAVs

- Projet commun avec IDVO
- Courte rédaction se rapportant à l'un des thèmes abordés en cours

Calcul de note finale

Projet (50%) + rédaction (50%)

Références et bibliographie

Les références et bibliographie seront mises en ligne [sur Moodle](#).

[21_IS6TC_APR2] Ateliers et projets informatiques 2

Semestre	S6
Crédits	10
VHT	414
CODE	21_IS6TC_APR2
Coord.	Antoine Dumeige

Modalités de validation de l'UE

Validation de l'UE par une moyenne supérieure à 10 et une note supérieure à 6 dans chaque ECUE.

[CPP] Atelier CPP

Niveau	L3
Semestre	SI6
UE	21_IS6TC_APR2
Coeff UE	1/19
Référent	Thierry Géraud

EQ-CM	0
RMD	21
TDP	20
FF	41
VHT	64

Résumé et contexte

Le langage C++ est un standard industriel pour obtenir des logiciels performants. Il est riche de très nombreuses notions, qui sont difficiles à appréhender dans leur globalité. Il est multi-paradigme, relativement de bas-niveau tout en proposant des abstractions de haut-niveau, mais possédant une syntaxe compliquée. Le fait de connaître les notions et la syntaxe du langage C, ainsi que le paradigme des langages à objets avec héritage comme Java, permet d'appréhender plus facilement le langage C++.

Acquis d'apprentissages visés (AAVs)

Dans le cadre d'un exercice de programmation en C++, il est attendu d'un étudiant au terme de l'activité :

- Qu'il analyse le problème et choisisse le(s) paradigme(s) de développement approprié(s) pour y répondre (procédural, orienté-objet, fonctionnel, programmation générique) ;
- Qu'il modélise un programme adapté et mette en oeuvre les paradigmes choisis dans le développement d'un programme en C++ moderne ;
- Qu'il applique les bonnes pratiques du développement C++, avec pour objectif l'écriture de programmes sûrs et performants ;
- Qu'il utilise les ressources de l'écosystème C++ à bon escient.

Plan cours

Les notions abordées sont :

- La compilation et l'écosystème C++
- Le typage C++: les types de bases, pointeurs, références et const-ness
- Prog. Objet : L'encapsulation 1 : Lier données et fonctions
- Prog. Objet : L'encapsulation 2 : restriction d'accès
- Prog. Objet : La vie d'un objet
- Prog. Objet : Définition d'opérateurs
- Gestion des ressources dynamique : Allocation dynamique
- Gestion des ressources dynamique : Gestion dynamique avec pointeurs intelligents (1/2)
- Gestion des ressources dynamique : Gestion et déplacement des ressources
- Prog. Objet : Hiérarchie de classes
- Prog. Objet : Typing, RTTI 1, Constructeurs et Destructeurs
- Prog. Objet : Encapsulation et RTTI 2
- Le types de polymorphisme
- Le polymorphisme paramétrique
- Généricité contrainte
- Les conteneurs STL in a nutshell
- Fonctions et lambdas: applications à la STL
- Exceptions
- Bonus: Constexpr et metaprogramming
- Design patterns et C++-20 modules

Format des activités

L'atelier est composé de :

- 6 heures de cours sous le format de capsule vidéos (de durées comprises entre 10 et 20 min)
- 21 heures de remédiation permettant une interaction directe concernant les notions abordées dans les capsules
- 20 heures de travaux pratiques et exercices dispensées par l'équipe assistante YAKA

Pré-requis

Les étudiants doivent avoir des bases solides de programmation en langage C. Une connaissance pratique de l'écosystème UNIX est également supposée. La connaissance d'un langage orienté-objet comme Java est un avantage.

Évaluation des AAVs

- Les séances de remédiation commencent par un tour rapide des planches correspondant aux capsules du jour ; des questionnaires Wooclap sont insérés pour avoir un retour rapide sur le niveau de compréhension des différentes notions par les étudiants.
- QCM Moodle : chaque notion abordée par les capsules vidéo donnera lieu à une évaluation sous Moodle. Les QCMs vérifient l'acquisition théorique des notions.
- Exercices du soir : des exercices à rendre tous les jours qui vérifient l'acquisition pratique des notions du jour et des jours précédents.
- Projet final en mode rush.

Les exercices sont moulinettés avec un retour immédiat vers les étudiants. Le projet est également évalué par une moulinette.

Calcul de note finale

La note finale est une combinaison des notes obtenues par les différentes évaluations :

- QCM: 20%
- Exercices du soir : 35%
- Projet final : 45%

Références et bibliographie

Livres de référence :

- [1]: Stroustrup, B. (2018). A Tour of C++. "Addison-Wesley Professional."
[2]: Meyers, S. (2014). Effective modern C++: 42 specific ways to improve your use of C++ 11 and C++ 14. " O'Reilly Media, Inc."

Vidéos d'introduction sur des notions spécifiques du C++ :

- [3]: [CPPCon Back to the Basics Tracks](<https://cppcon2021.sched.com/?searchstring=Back+to+Basics>)

[JAVA] Atelier JAVA

Niveau	L3
Semestre	S16
UE	21_IS6TC_APR2
Coeff UE	1/9
Référent	Antoine Dumeige

EQ-CM	0
RMD	21
TDP	20
FF	41
VHT	72

Résumé et contexte

Atelier d'apprentissage des bases de l'écosystème Java, du placement industriel à la syntaxe, en s'arrêtant rapidement sur le fonctionnement d'un ramasse-miettes.

Acquis d'apprentissages visés (AAVs)

Au terme de l'atelier, il est attendu d'un étudiant qu'il ai acquis une connaissance avancée du langage Java, en ayant compris les principes fondamentaux (Passage par valeur de référence, ramasse-miette, place prépondérante des interfaces, expressions lambda) et en situant correctement l'outil Java dans l'écosystème des langages de programmation.

L'étudiant devra savoir créer un projet, le compiler et l'exécuter dans l'IDE de son choix.

Il devra être capable d'écrire une série de test pour spécifier en amont et valider en aval son travail.

Il devra enfin être en maîtrise des outils de traitement par flux (Streams), des expressions lambda et des constructions de programmation asynchrone proposées par le langage et la bibliothèque standard l'accompagnant.

Enfin, lors du projet de validation l'étudiant devra être capable de concevoir, planifier et réaliser un projet respectant un protocole d'interconnexion réseau simple pour produire un automate.

Plan cours

- J1: placement industriel, objectifs, core language et héritage
- J2: Programmation générique, interfaces, classes anonymes et expressions lambda.
- J3: Programmation asynchrone et ramasse-miettes
- J4: TDs

Format des activités

- FAQ et CM le matin
- TD d'exercices l'après-midi
- Permanence et rendu le soir
- Rush à partir de jeudi soir

Pré-requis

- Cours modélisation Orientée Objet (MOB1)
- Projets Pratiques S5

Évaluation des AAVs

Chaque exercice couvre directement un ou plusieurs AAV, le rush recoupe tous les AAV.

Sauf cas contraire explicitement stipulé dans le sujet, les exercices doivent être réalisés avec 100% de tests validés pour être considérés validés.

Le rush propose une série d'achievements, la réalisation de 15 d'entre eux permettra à l'étudiant de valider le rush. En sus des 15 requis pour validation, une catégorie d'achievements spécifiques appelés "Saga" sera disponible, leur validation attribuera un point bonus à l'atelier, jusqu'à un maximum de cinq.

Calcul de note finale

L'atelier est composé de 4 éléments à valider indépendamment : 4 exercices du soir et un rush.

L'exercice mytracker (j4) passe en bonus.

Plus l'étudiant valide d'éléments, plus sa note est élevée :

Nb Éléments Validés	Note obtenue	
---	---	---
0	0	
1	1	
2	2	
3	10	
4	20	
5	20 + un mug dédié par le chef assistant	

En sus, comme précisé dans le point de notation du rush, des points bonus peuvent venir s'ajouter, sans que la note globale puisse toutefois dépasser 20.

Références et bibliographie

<https://docs.oracle.com/en/java/>

<https://docs.oracle.com/en/java/javase/17/>

[JS] Atelier JS

Niveau	L3
Semestre	SI6
UE	21_IS6TC_APR2
Coeff UE	1/9
Référent	Christophe Eymard

EQ-CM	3
RMD	0
TDP	32
FF	35
VHT	71

Résumé et contexte

TBD.

Acquis d'apprentissages visés (AAVs)

Dans le cadre d'un projet javascript traditionnel, l'étudiant :

- identifie les différents éléments d'un projet
- exécute le projet correctement
- comprend et rédige du code javascript fonctionnel

Plan cours

TBD

Format des activités

- Cours magistraux asynchrones sous format vidéo.
- Projet : projet individuel évalué à 20 heures de travail, aidé par les assistants.
- Accompagnement et question/réponse à distance via salon de discussion vocaux privés.

Pré-requis

Avoir validé l'atelier Java.

Évaluation des AAVs

- correction automatique des travaux pratiques, du point de vue de l'aspect fonctionnel *et* de la forme du projet
- correction automatique du projet individuel

Calcul de note finale

TP / 8 ou 7 à valider pour le mercredi soir
Rush / 12 ou 13

Références et bibliographie

- * MDN (Mozilla Developer Network)
- * stackoverflow
- * cheatsheet (support fourni)

[TC] Projet TIGER

Niveau	L3
Semestre	SI6
UE	21_IS6TC_APR2
Coeff UE	3/9
Référent	Etienne Renault

EQ-CM	24
RMD	0
TDP	0
FF	24
VHT	98

Résumé et contexte

Le projet TIGER vise à enseigner aux étudiants des notions d'informatique ayant trait aux langages de programmation, à travers l'écriture d'un compilateur pour le langage "jouet" Tiger.

Ecrire un compilateur n'est pas l'unique objectif du projet : il s'agit également d'un excellent support pour l'enseignement de concepts et de savoir faire fondamentaux en informatique :

- découverte de paradigmes de langages de programmation
- apprentissage du C++ et de design patterns
- manipulation d'outils de développement
- gestion d'un gros projet sur le long terme au sein d'une équipe de plusieurs étudiants.

Acquis d'apprentissages visés (AAVs)

À la fin de ce cours les étudiants:

- maîtriseront la théorie de la chaîne de compilation de bout en bout. Cela implique d'être en mesure de comprendre le rôle et l'importance de chaque partie du compilateur, d'en distinguer les parties fondamentale et celle qui relève de l'optimisation, et enfin de voir comment le langage d'entrée impacte la conception du compilateur. Les étudiants maîtriseront la réalisation de la partie frontale d'un compilateur jusque LLVM.
- sauront produire du code C++ de qualité et l'insérer dans un code existant. Cela implique d'être en mesure d'écrire des classes simples et de comprendre que les interfaces vont jouer un rôle dans la suite du projet, d'appliquer des patrons de conception lorsque cela est nécessaire, d'apprendre à factoriser du code (soit via le design soit via la programmation générique), et d'utiliser les derniers concepts du C++.
- maîtriseront la gestion d'un projet long en équipes de 4. Cela implique d'identifier et reconnaître les jalons et d'effectuer la répartition des tâches. Cela nécessite également de savoir mettre en place une bonne méthodologie de gestion de projet via la réalisation de tests et d'outils adaptés.

Les étudiants qui effectueront l'électif "backend" maîtriseront la réalisation d'un compilateur dans son ensemble.

Plan cours

Le projet est divisé en une partie cours, qui détaille les notions théoriques associées à la construction d'un compilateur, et une partie projet, s'étalant sur 7 semaines et divisé en tranches pour lesquelles du "code à trous" est fourni. Ces tranches se répartissent de la manière suivante:

- TC1: Construction d'un analyseur syntaxique et lexical
- TC2: Construction de l'arbre de syntaxe abstraite
- TC3: Calcul des liaisons
- TC4: Vérification des types
- TCL: Traduction vers la représentation intermédiaire de LLVM
- TC5 (électif Tiger-backend): Traduction vers une représentation intermédiaire "maison"
- TC6 (électif Tiger-backend): Canonisation de la représentation intermédiaire
- TC7 (électif Tiger-backend): Sélection des instructions
- TC8 (électif Tiger-backend): Analyse de la vivacité
- TC9 (électif Tiger-backend): Allocation des registres

Format des activités

24h de CM et 7 semaines de Projet

Pré-requis

Notions de C++

Évaluation des AAVs

L'enseignement est évalué en deux parties :

- la partie théorique (cours) sera évaluée au travers d'un examen Moodle
- la partie technique (projet) sera évaluée automatiquement à la fin de chaque tranche. Afin de permettre à chacun d'aller à son rythme les règles de calcul pour le projet sont les suivantes (TCX_Y se lit l'évaluation à la tranche Y de la tranche X). On y voit que les options (object-support, TC-A...) ne font que rapporter des points supplémentaires.

$TC1 = \max (TC1_1, TC1_2, TC1_3, TC1_4, TC1_L)$

$TC2 = \max (TC2_2, TC2_3, TC2_4, TC2_L)$

$TC3 = \max (TC3_3, TC3_4, TC3_L)$

$TC4 = \max (TC4_4, TC4_L)$

$TCL = \max (TCL_L) + \max (\text{object-support}) + \max(TC-A) + \max(TC-B) + \max(TC-C) + \max(TC-I) + \max(TC-D).$

Calcul de note finale

Projet (80%) + CM (20%)

Références et bibliographie

Les planches de cours <https://www.lrde.epita.fr/~tiger/lecture-notes/slides/ccmp/>

Le sujet du projet: <https://assignments.lrde.epita.fr>

Tiger Reference Manual: https://assignments.lrde.epita.fr/reference_manual/reference_manual.html

Le dépôt pour l'accès aux tranches du projet: <https://gitlab.lrde.epita.fr/tiger/tc-base>

Le livre de référence d'Andrew W. Appel <https://www.cs.princeton.edu/~appel/modern/>

Le dragon book (pour les personnes voulant un ouvrage plus technique) [https://en.wikipedia.org/wiki/Compilers: Principles, Techniques, and Tools](https://en.wikipedia.org/wiki/Compilers:_Principles,_Techniques,_and_Tools)

[JWS] Projet JAVA WEB SERVICES

Niveau	L3
Semestre	SI6
UE	21_IS6TC_APR2
Coeff UE	1/9
Référent	Antoine Dumeige

EQ-CM	1
RMD	0
TDP	0
FF	1
VHT	40

Résumé et contexte

Ce projet a deux objectifs distincts :

- Mettre en œuvre les langages Java et SQL.
- S'initier à l'utilisation d'un framework de développement ;
- Initier l'étudiant aux architectures logicielles via le développement d'une architecture en couches.

Acquis d'apprentissages visés (AAVs)

Au terme de ce projet, il est attendu qu'un étudiant ait acquis la connaissance de ce qu'est une architecture en couche. Il est également attendu qu'il sache développer en conformité avec une norme de développement en respectant ladite architecture en couche, qu'il sache organiser correctement son code, qu'il respecte des normes de nommage précises et qu'il ait acquis une plus grande autonomie dans sa pratique des langages Java et SQL. Enfin, il est attendu que l'étudiant ait su analyser la problématique donnée pour concevoir un modèle objet à la fois synthétique et complet.

Plan cours

- Objectifs des architectures logicielles.
- Définition de l'architecture en couche.
- Exemple synthétique de mise en œuvre.

Format des activités

- Cours magistraux asynchrones sous format vidéo.
- Projet : projet individuel évalué à 20 heures de travail, aidé par les assistants.
- Accompagnement et question/réponse à distance via salon de discussion vocaux privés.

Pré-requis

Connaissance fonctionnelle du langage JAVA
Connaître le langage SQL

Évaluation des AAVs

L'évaluation est réalisée via la soumission d'un projet qui sera confronté à une batterie de tests fonctionnels et de conformité technique.

Calcul de note finale

La batterie de tests est organisée en catégories (initialisation, réponse simple, architecture...).
La validation de chaque catégorie permet le gain de points, jusqu'à un maximum de 20.

Références et bibliographie

* Quarkus, Creating your first application: <https://quarkus.io/guides/getting-started>

[PING] Projet de fin d'ING1

Niveau	L3
Semestre	SI6
UE	21_IS6TC_APR2
Coeff UE	2/9
Référents	Bashar Dudin Antoine Dumeige

EQ-CM	1
RMD	0
TDP	2.5
FF	3.5
VHT	60

Résumé et contexte

PING est un projet de fin d'année, c'est une répétition générale au cours de laquelle vous allez être amenés à mobiliser le minima des compétences du développeur informatique. Vous allez devoir travailler en groupe efficacement, pour produire du code normé et répondant à l'expression de besoins clients. Dans le cas présent on attend de vous de créer un IDE en tandem avec votre coach. Il y a un coach par groupe. Le ou la coach agit en tant que client, utilisateur final ou PO suivant le positionnement qu'il ou elle choisit. Il sera de votre responsabilité d'assimiler les modalités de travail que le positionnement de votre coach vont imposer.

Acquis d'apprentissages visés (AAVs)

Au terme de l'atelier, il est attendu d'un étudiant qu'il ai acquis une connaissance avancée du langage Java, en ayant compris les principes fondamentaux (Passage par valeur de référence, ramasse-miette, place prépondérante des interfaces, expressions lambda) et en situant correctement l'outil Java dans l'écosystème des langages de programmation. L'étudiant devra savoir créer un projet, le compiler et l'exécuter dans l'IDE de son choix. Il devra être capable d'écrire une série de test pour spécifier en amont et valider en aval son travail. Il devra enfin être en maîtrise des outils de traitement par flux (Streams), des expressions lambda et des constructions de programmation asynchrone proposées par le langage et la bibliothèque standard l'accompagnant. Enfin, lors du projet de validation l'étudiant devra être capable de concevoir, planifier et réaliser un projet respectant un protocole d'interconnexion réseau simple pour produire un automate.

Plan cours

Le projet se déroule en deux temps: Un premier au cours duquel vous prenez possession du sujet et vous avancez sur les spécifications techniques. Cette partie de votre travail sera moulinetté et les spécifications minimales les mêmes pour tout le monde. Vos repos de rendus techniques seront disponibles, comme d'habitude sur l'intra des assistants.

Un second qui vous amènera à créer le front de votre IDE. Ce travail se fera en osmose avec votre coach et donnera lieu à une livraison respectant ses attentes. Ces temps sont une inversion de la démarche que vous devriez avoir lorsque vous développez un produit. C'est un choix que l'on fait ici pour deux raisons : on souhaite d'une part vérifier que vous savez poser une API correctement, on aimerait de l'autre vous montrer dans quelle mesure deux fronts posés sur un même back peuvent être différents. Le front c'est un job.

La séance de restitution de votre projet aura lieu en accord avec votre coach.

Format des activités

Projet en groupe ponctué de séance avec votre coach.

Pour interagir avec votre coach il vous faudra être proactifs et prendre rendez-vous suffisamment tôt au regard des disponibilités de ceux-ci. Les coaches ne viendront pas vers vous. Il est important de noter que votre coach est une

ressource limitée, à partir de la semaine de la date d'entrée des coachs en jeu, vous avez le droit à 1h d'échange avec votre coach la première semaine, 30 minutes d'échange hebdomadaire avec votre coach à partir de la deuxième semaine. Ces disponibilités sont les seules que vous aurez, votre enjeu sera de tirer le meilleur parti de ce temps à votre disposition.

Évaluation des AAVs

Projet.

Calcul de note finale

Se fera sur la base d'une grille critérée.

Références et bibliographie

<https://docs.oracle.com/en/java/> <https://docs.oracle.com/en/java/javase/17/>

[21_IS6TC_MAT2] Mathématiques 2

Semestre	S6
Crédits	4
VHT	102
CODE	21_IS6TC_MAT2
Coord.	Bashar Dudin

Modalités de validation de l'UE

Validation de l'UE par une moyenne supérieure à 10 et une note supérieure à 6 dans chaque ECUE.

[PBS1] Probabilités et statistiques 1

Niveau	L3
Semestre	SI6
UE	21_IS6TC_MAT2
Coeff UE	3/6
Référent	Guillaume Euvrard

EQ-CM	4
RMD	10
TDP	16
FF	24
VHT	42

Résumé et contexte

TBA.

Acquis d'apprentissages visés (AAVs)

Fonction de répartition et densité:

- déterminer si une fonction donnée est une densité ou non
- calculer une probabilité à partir d'une densité ou fonction de répartition
- connaître des lois de référence
- interpréter géométriquement: savoir se ramener sans calcul à des lois de référence
- déterminer la densité d'une transformée simple d'une variable aléatoire

Calcul de moments:

- calculer une espérance et une variance à partir d'une densité ou fonction de répartition
- interprétation géométrique: savoir deviner sans calcul un interval où se trouvent les moments
- se ramener à des lois de référence

Vecteur aléatoire

- densité d'un couple de variables: calculer des probabilités
- déterminer les densités marginales à partir de la densité jointe
- décider si deux variables aléatoires sont indépendantes ou non
- déterminer une covariance et une corrélation. I
- interpréter en terme de dépendance
- déterminer graphiquement si indépendance ou non, et le signe de la corrélation

Suite de variables aléatoires

- évaluer s'il y a convergence, et de quel type
- théorème central limite: déterminer un interval de prédiction

Analyse de données

- lire un jeu de données dans un logiciel statistique (R)
- estimer l'espérance et la variance,
- notion de test d'hypothèse: déterminer un intervalle de confiance sur l'espérance
- évaluer si deux échantillons ont même espérance
- évaluer et interpréter un coefficient de corrélation

Plan cours

1. Variable aléatoire

- Passer du discret au continu: tribue borélienne, répartition, densité
- Densité d'une variable transformée
- Notions d'intervalle de prédiction et d'intervalle de confiance

2. Moments d'une variable aléatoire

- Espérance et variance
- Espérance et variance d'une fonction d'une variable
- Inégalité de Chebyshev

3. Couple de variables aléatoires

- Densité jointe, densités marginales
- Indépendance
- Moments, covariance et corrélation

4. Suite de variables aléatoires

- Différents types de convergence
- Loi des grands nombres
- Théorème Central Limite

5. Travaux Pratiques

- Analyse de jeux de données

Format des activités

1. Cours et TD

- RMD: 4 séances de 2h
- TD: 5 séances de 2h

2. Travaux Pratiques en salle machine

CM: 1 séance de 2h (logiciel R)

TP: 2 séances de 2h

Pré-requis

Probabilités discrètes

Intégration

Fonction de deux variables réelles

Évaluation des AAVs

FRM: Evaluations formatives: exercices sur Moodle répartis au cours du temps.

EXM: Examen Moodle

TP: Travaux Pratiques

Calcul de note finale

$(6 \cdot \text{EXM} + 2 \cdot \text{TP}) / 8$

Références et bibliographie

TBA.

[CHIFR] Chiffrement et codes correcteurs

Niveau	L3
Semestre	SI6
UE	21_IS6TC_MAT2
Coeff UE	1/6
Référent	Sandra Marcello

EQ-CM	6
RMD	8
TDP	6
FF	14
VHT	24

Résumé et contexte

Ce cours est une introduction aux mathématiques de la cryptographie et aux codes correcteurs d'erreurs linéaires. Un rappel des éléments d'arithmétique modulaire sera effectué.

Le cours se concentrera sur différents aspects de la cryptographie asymétrique : schéma de chiffrement à clef publique, schéma de signature.

Des problèmes difficiles sous-jacents permettant d'assurer la sécurité théorique des schémas présentés seront explicités.

Un algorithme permettant d'attaquer sur les problèmes présentés sera présenté.

Les éléments nécessaires à la cryptographie sur des courbes elliptiques seront présentés.

Les codes correcteurs linéaires seront introduits afin de montrer comment ceux-ci peuvent être utilisés en cryptographie pour construire un schéma de chiffrement dont le problème reste difficile même avec l'avènement d'un ordinateur quantique.

Acquis d'apprentissages visés (AAVs)

TBD.

Plan cours

Arithmétique Modulaire

Cryptographie symétrique et asymétrique

Schéma de chiffrement RSA, El Gamal.

Protocole de Diffie-Hellman.

Schéma de chiffrement à base de courbe elliptique.

Introduction aux codes correcteurs d'erreurs linéaires.

Présentation d'une adaptation du schéma de chiffrement de McEliece.

Présentation du problème difficile sous-jacent au schéma de chiffrement de McEliece : indistingabilité des codes de Goppa.

Format des activités

CM + TD

Évaluation des AAVs

40% DM et 60% DST

Calcul de note finale

Moyenne de tous les contrôles continus (même coef pour tous les CC)

Références et bibliographie

D. Huffman V. Pless Fundamentals of Error-Correcting Codes Springer

D. Knuth : The art of programming Addison-Wesley Professional

N. Koblitz, Algebraic Aspects of Cryptography (Springer 1997)

X Marsault, Compression et cryptage en Informatique (Hermès 1992) pp 1 à 79

J. Menezes, P.C van Oorschot, P.C vanstone Handbook of applied cryptography

G. Robin, Algorithmique et Cryptographie (Ellipses 1991) pour des détails mathématiques

B. Schneier, Cryptographie Appliquée (Thomson Publ.1997) pour une approche approfondie de la cryptographie

D.R Stinson Cryptography : theory and practice CRC Press

J.H. Van Lint Introduction to Coding Theory (Springer 1982),

G. Zemor Cours de cryptographie Cassini

[CAMA] Calcul matriciel

Niveau	L3	EQ-CM	12
Semestre	SI6	RMD	12
UE	21_IS6TC_MAT2	TDP	12
Coeff UE	2/6	FF	24
Référent	Olivier Ricou	VHT	36

Résumé et contexte

Revoir et découvrir l'utilisation des matrices pour résoudre différents problèmes en mettant les mains dans le code. La programmation est vectorielle et s'appuie sur les bibliothèques Numpy et Scipy

Acquis d'apprentissages visés (AAVs)

Il est attendu qu'un élève sache après ce cours

- programmer de façon vectorielle en Python en s'appuyant sur Numpy
- utiliser les méthodes itératives et les astuces permettant d'améliorer la convergence
- traduire une présentation mathématique sur le sujet et en un programme efficace

Plan cours

- Transformations isométriques, changement de repère
- Vecteurs propres et valeurs propres, analyse en composantes principales (ACP)
- Résolution d'un système matriciel
 - * résolution directe : pivot de Gauss, décomposition LU
 - * impact du conditionnement de la matrice
 - * résolution à l'aide de méthodes itératives : Jacobi, Gauss-Seidel, surrelaxation
- Problèmes d'optimisation
 - * méthode du gradient en général et pour résoudre un système matriciel
 - * méthode du gradient conjugué
- Systèmes matriciels non linéaire
- Décomposition en valeurs singulières (SVD)

Tous les cours s'appuient sur Python et Numpy tant pour présenter les concepts dans les cours magistraux que pour les TP.

Format des activités

Cours asynchrones avec des temps de remédiation (12h) en CM
12h de TP

Pré-requis

Savoir ce qu'est une matrice, un vecteur, le produit scalaire, le produit matrice vecteur

Évaluation des AAVs

Evaluations continues via Moodle.

Calcul de note finale

Moyenne de tous les contrôles continus (même coef pour tous les CC)

Références et bibliographie

<https://python3.mooc.lrde.epita.fr/tree/lesson5%20Scipy/matrices> pour les maths

<https://python3.mooc.lrde.epita.fr/tree/lesson4%20Numpy> pour Numpy

<https://python3.mooc.lrde.epita.fr/notebooks/lesson5%20Scipy/sp01%20Calcul%20matriciel.ipynb> pour la partie algèbre linéaire de Scipy

[21_IS6TC_IF2] Informatique fondamentale 2

Semestre	S6
Crédits	4
VHT	127
CODE	21_IS6TC_IF2
Coord.	Adrien Pommellet

Modalités de validation de l'UE

Validation de l'UE par une moyenne supérieure à 10 et une note supérieure à 6 dans chaque ECUE.

[LOF0] Logique formelle

Niveau	L3
Semestre	SI6
UE	21_IS6TC_IF2
Coeff UE	2/8
Référent	Adrien Pommellet

EQ-CM	6
RMD	12
TDP	0
FF	12
VHT	18

Résumé et contexte

Ce cours présente principalement deux notions théoriques avec des applications concrètes en informatique :

- Les systèmes de preuves, qui introduisent la notion de vérité comme construction obtenue à partir d'une série de déductions ; des logiciels de preuve comme COQ ou des langages certifiés comme l'Atelier B en dépendent.
- Le lambda calcul, formalisme de programmation différent des machines de Turing mais tout aussi puissant, que l'on peut typer et qui sert de base à la programmation fonctionnelle.

Ces deux notions sont reliées par l'isomorphisme de Curry-Howard. Un programme est donc d'une certaine manière une forme de preuve mathématique.

Acquis d'apprentissages visés (AAVs)

A l'issue de ce cours, les étudiants sont capables de :

- Formuler un théorème sous forme de formule de la logique propositionnelle.
- Vérifier la correction d'un système de preuve, et de rappeler le concept de complétude
- Analyser une formule et de la prouver grâce aux systèmes à la Hilbert et à la déduction naturelle, en écrivant une arbre de preuve (pour les cas simples) ou complétant un arbre plus complexe (pour des cas compliqués).
- De prouver des réductions et des conversions de lambda expression.
- De jouer avec les entiers et Booléens de Church pour décrire des opérations classiques de programmation.
- De prouver le type simple d'un lambda terme.
- Appliquer l'isomorphisme de Curry-Howard pour lier déduction naturelle intuitionniste et lambda calcul étendu.

Plan cours

Ce cours est découpé en six séances constituées d'une heure de vidéo asynchrone puis d'une heure de remédiation :

1. Logique propositionnelle et sa sémantique.
2. Preuves, systèmes à la Hilbert.
3. Déduction naturelle.
4. Lambda calcul et sémantique.
5. Stratégies de réduction du lambda calcul.
6. Typage du lambda calcul, isomorphisme de Curry-Howard.

Format des activités

- 6 * 1h = 6h de vidéos, accompagné d'un travail personnel de 6 * 1h30 = 9h à chaque séance.
- 6 * 1h = 6h de remédiation.

Pré-requis

- Capacité à comprendre et écrire des preuves par récurrence
- Capacité à lire des formules mathématiques et à exprimer des propriétés sous forme de formule.
- Il est souhaitable d'avoir suivi FOLO.

Évaluation des AAVs

Un contrôle continu d'une demi-heure, et un examen de 1 heure.

Calcul de note finale

$1/3 \text{ [Note CC]} + 2/3 \text{ [Note Examen]} = \text{[Note totale]}$

Références et bibliographie

<https://www.lrde.epita.fr/~adrien/lofo.html>

[TYLA] Typologie des langages de programmation

Niveau	L3	EQ-CM	12
Semestre	SI6	RMD	0
UE	21_IS6TC_IF2	TDP	0
Coeff UE	1/8	FF	12
Référent	Etienne Renault	VHT	12

Résumé et contexte

L'outil principal du programmeur est le langage de programmation. Ce cours cherche à démontrer que les langages ne sont pas égaux, et que certains sont plus adaptés que d'autres. Il cherche à montrer les erreurs du passé, et à souligner qu'elles sont souvent reproduites. Il insiste sur le lien étroit, mais presque invisible, qui existe entre les technologies de compilation existantes, et la définition d'un langage, pour mettre l'accent sur l'évolution des langages de programmation, et le besoin de promouvoir de nouvelles technologies. Enfin, puisque le choix n'est pas toujours disponible, il cherche à promouvoir l'esprit de langages riches dans la lettre du langage imposé.

Acquis d'apprentissages visés (AAVs)

TBD.

Plan cours

01. Introduction Générale
02. Histoire des machines et des langages de programmation
03. Les sous-programmes
04. Stratégies d'évaluation et de retour
05. Un tour des langages object : Simula, Smalltalk, Eiffel
06. La programmation orienté contrat
07. Héritage et Généricité
08. Aperçu des langages "logique"
09. Le parallélisme dans les langages (si le temps le permet
10. Tour d'horizon d'un langage "moderne"

Format des activités

- Cours dispensé sous la forme de CM.
- Exercices d'applications notés (à faire chez soi en travaillant le cours).

Pré-requis

Programmation impérative et programmation objet.

Évaluation des AAVs

- Exercices notés à faire chez soi.
- Examen final de 90 minutes.

Calcul de note finale

30% [Travaux à faire à la maison] + 70% [Examen final]

Références et bibliographie

- "Out of their Minds", Denis Shasha and Cathy Lazere
- "Comparative programming languages", Leslie B. Wilson et Robert G. Clark
- "Concepts of programming Languages", Robert W. Sebesta
- "Conception et programmation orientées objet", B. Meyer
- "HOPL conference: History of programming Languages"

[THEG] Théorie des graphes

Niveau	L3	EQ-CM	12
Semestre	SI6	RMD	12
UE	21_IS6TC_IF2	TDP	0
Coeff UE	3/8	FF	12
Référent	Alexandre Duret-Lutz	VHT	30

Résumé et contexte

L'objectif du cours est quadruple:

- introduire le vocabulaire de la théorie des graphes et d'illustrer son vaste champ d'applications
- prolonger le cours de complexité des algorithmes en l'appliquant aux algorithmes de graphes, ainsi qu'en montrant de nouveaux algorithmes de programmation dynamique
- relier les graphes à la théorie des groupes (en particulier, semi-anneaux)
- préparer au cours de recherche opérationnelle qui sera sanctionné par un projet mêlant des problématiques de graphe.

Acquis d'apprentissages visés (AAVs)

TBD.

Plan cours

- Introduction à la théorie des graphes, à l'aide de plein d'exemples:
 - Démonstration de la formule des Alcanes
 - Formule de Cayley
 - Sous-arbres couvrants et circuits électroniques
 - Isomorphisme de graphe comme Zero Knowledge Proofs
- Briques algorithmiques de bases (DFS/BFS/...)
- Représentation de graphes
- Circuit Eulériens
- Exploration de l'espace d'états d'un Rubik's cube
- Distances
 - avec poids positifs (Dijkstra)
 - avec poids négatifs (Bellman-Ford, programmation dynamique)
 - n sources / n destinations (Floyd-Warshall)
- vision algébrique
 - semi-anneaux, puissance de matrices
 - résolution de problème de dénombrement
- couplages
 - applications
 - recherche de chemins améliorants
 - couplages pondérés
- composantes connexes et fortement connexes

Format des activités

12h de CM et des exercices de programmation en ligne hebdomadaires sur Moodle.

Pré-requis

- Python (Les exercices seront des algos de graphe à coder en Python)
- CPXA (cours de Complexité des algorithmes)
- Notions de théorie des groupes

Évaluation des AAVs

Exercices individuels à faire via Moodle

Calcul de note finale

100% contrôle-continu (Exercices en ligne hebdomadaires)

Références et bibliographie

Introduction à l'algorithmique. Cormen et al. Edition Dunod

[MOB2] Modélisation objet 2

Niveau	L3	EQ-CM	12
Semestre	SI6	RMD	12
UE	21_IS6TC_IF2	TDP	0
Coeff UE	2/8	FF	12
Référent	Didier Verna	VHT	24

Résumé et contexte

MOB2 présente une approche orientée objet alternative aux systèmes classiques décrits dans MOB1. Bien que toujours basée sur la notion de classe, cette approche possède un certain nombre de traits distinctifs : typage dynamique, multi-méthodes, et protocole méta-objet en sont les principaux. Notre but ici est à la fois de montrer qu'il n'existe pas qu'une seule vision du paradigme objet, mais aussi que nombre de faiblesses des systèmes traditionnels disparaissent dans cette approche plus expressive.

Acquis d'apprentissages visés (AAVs)

Voir clés {CG.02} {CG.03} {CS1_B.1.K3}.

Plan cours

1. CLOS - the Common Lisp Object System : introduction, classes et objets, polymorphisme, héritage.
2. CLOS Avancé : protocoles standard, méta-classes, redéfinitions.
3. Étude de cas 1 - programmation des méthodes binaires: non-problèmes, explicitation du concept niveau utilisateur, niveau implémentation.
4. Étude de cas 2 - Design Patterns: introduction, exemple du visiteur en C++, en Lisp, visiteurs à état, conclusion.

Format des activités

Visionnage asynchrone de vidéos (12h) puis remédiation (12h).

Pré-requis

MOB1 et PFON

Évaluation des AAVs

Contrôle continu en conditions d'examen à raison d'un par chapitre de cours.

Calcul de note finale

Moyenne des CC.

Références et bibliographie

- Linda G. DeMichiel and Richard P. Gabriel. The Common Lisp Object System : An Overview. ECOOP, 1987
- Sonja E. Keene. Object-Oriented Programming in Common Lisp : a Programmer's Guide to CLOS. Addison-Wesley, 1989
- Daniel G. Bobrow, Linda G. DeMichiel, Richard P. Gabriel, Sonya E. Keene, Gregor Kiczales and David A. Moon. Common Lisp Object System specification. ACM SIGPLAN Notices, 1988.
- Didier Verna. Binary Methods Programming : the CLOS Perspective. JUCS Vol.14.20, 2008.
- Bruce, K. B., Cardelli, L., Castagna, G., Eifrig, J., Smith, S. F., Trifonov, V., Leavens, G. T., and Pierce, B. C. (1995). On binary methods. Theory and Practice of Object Systems, 1(3):221–242.
- Castagna, G. (1995). Covariance and contravariance: conflict without a cause. ACM Transactions on Programming Languages and Systems, 17(3):431–447.

[21_IS6TC_IRT1] Innovation, recherche et approfondissement technique

Semestre	S6
Crédits	3
VHT	78
CODE	21_IS6TC_IRT1
Coord.	Bashar Dudin

Modalités de validation de l'UE

Validation de l'UE par une moyenne supérieure à 10 et une note supérieure à 6 dans chaque ECUE.

[ER01] Éléments de recherche opérationnelle 1

Niveau	L3
Semestre	SI6
UE	21_IS6TC_IRT1
Coeff UE	2/4
Référent	Bashar Dudin

EQ-CM	12
RMD	6
TDP	6
FF	12
VHT	36

Résumé et contexte

Ce cours est une première exposition aux problématiques de recherche opérationnelle. Cet enseignement se déroule en apprentissage par problèmes (APP). Il s'agit de mettre les étudiants face à un problème qui guidera vos apprentissages par les besoins qu'il crée. Ce problème se doit d'être à la fois suffisamment complexe pour effectuer un travail de fond et relativement accessible, du moins par tranches. Cet APP se déroule par équipe de 4/5 personnes. Le problème à aborder dans cette activité est celui d'optimiser le parcours des déneigeurs dans une ville. Il donnera lieu à un résumé et un exposé du travail fourni.

Le cours vise à apporter :

- une première exposition aux problématiques de recherche opérationnelle
- une mise en pratique de la démarche scientifique : Analyser et formaliser - Faire un état de l'art - Modéliser - Valider - Remettre en question
- une expérience de chaîne de développement type Implémenter - Tester - Optimiser dans le cadre d'une problématique de R&D
- un questionnement quant aux prises de décisions en rapport avec l'orientation à prendre dans le développement du livrable.

Acquis d'apprentissages visés (AAVs)

En fin d'APP, dans le contexte d'un problème de RO déterministe simple, on attend de vous que vous soyez collaborativement en mesure:

- d'implémenter une boucle de rétroaction : Analyser et formaliser -> Faire un état de l'art -> Modéliser -> Valider -> Remettre en question
- d'implémenter parallèlement en place une chaîne de développement type Implémenter -> Tester -> Optimiser dans le cadre d'une problématique de R&D
- d'analyser les points de bascules des boucles précédentes à chaque itération (les prises de décisions en rapport avec l'orientation à prendre dans le développement du livrable).

Par ailleurs on souhaite que vous soyez, face à un problème de recherche opérationnelle simple, individuellement en mesure :

- de formaliser un problème d'optimisation simple sous forme standard
- de reconnaître les problèmes de recherche opérationnelle standards
- d'expliquer quelques algorithmes apportant une solution aux problèmes précédent et d'en connaître la complexité en temps.

Plan cours

L'activité est une activité en apprentissage par problèmes. Le problème posé aux étudiants est le suivant : La mairie de Montréal souhaite optimiser les trajets de ces équipes de déblaiement suite aux épisodes neigeux. Votre équipe participe à la réalisation d'une solution qui vise à optimiser le trajet des engins qui participent aux opérations de déblaiement.

Trois séances de remédiations et trois séances de mise en pratique vont accompagner cette activité, elles seront découpées comme suit :

Comment s'attaquer à un problème de RO ? RMD1

Problèmes de flots : RMD2 / TD1

Problèmes de transport optimal : RMD3 / TD2

Problèmes de couplages : TD3

Format des activités

- a) Confrontation à un problème authentique à traiter
- b) Mise à disposition de temps de ressources (coaching) pour échange et suivi
- c) Conférences en séances plénières traitant de problématiques de RO générales
- d) Séances de travaux dirigés

Pré-requis

- Une première exposition à l'algorithmique des graphes (THEG)
- Une première expérience du travail collectif sur un projet informatique.

Évaluation des AAVs

Rendu de projet (code, synthèse, vidéo) et soutenance en groupe.

Calcul de note finale

$[NOTE] = 75\%[NOTE \text{ PROJET}] + 25\%[NOTE \text{ SOUTENANCE}]$

Références et bibliographie

- Voir le conteneur Moodle du cours.

[TIFT] **Electif technique**

Voir document annexe.

[TIFIR] **Electif Exposition et initiation à la recherche**

Voir document annexe.

[21_IS6TC_MSH2] Management et sciences humaines 2

Semestre	S6
Crédits	3
VHT	80
CODE	21_IS6TC_MSH2
Coord.	Hélène Vaury

Objectifs de l'UE

Approfondir les capacités d'interaction dans un milieu professionnel multiculturel ainsi que la dimension éthique d'un ingénieur.

Compétences

- Agir en ingénieur responsable conscient de ses actions et de l'impact de ses réalisations (techniques) sur son environnement.
- Respecter les opinions de ses partenaires professionnels et être ouvert dans les confrontations qui peuvent en découler.
- Prendre en compte l'expérience client et ses besoins.
- Échanger par oral et par écrit dans une langue étrangère.
- Savoir communiquer et interagir à l'écrit de façon claire et précise en respectant les usages et coutumes du milieu dans lequel on évolue.

Modalités de validation de l'UE

Validation de l'UE par une moyenne supérieure à 10 et une note supérieure à 6 dans chaque ECUE.

[ODP2C0] Développement des soft skills de collaboration

Niveau	L3
Semestre	SI6
UE	21_IS6TC_MSH2
Coeff UE	1/5
Référents	Hélène Vaury François-Xavier Dufour Anne Dewilde

EQ-CM	12
RMD	0
TDP	0
FF	12
VHT	16

Résumé et contexte

TBD.

Acquis d'apprentissages visés (AAVs)

TBD.

Plan cours

La gestion de Conflit

- 1) Conscience de son impact sur l'autre
- 2) Communication non-violente

L'assertivité et l'affirmation de Soi

- 1) Les fondamentaux de l'affirmation de soi
Prendre conscience de son profil (Auto-diagnostic des comportements)
Les comportements refuges
Les mécanismes des situations de tension
Les clés de la communication asservie
Mise en pratique : Recevoir et exprimer un compliment
- 2) Utiliser les outils de l'assertivité dans son quotidien
Affirmer une position
Formuler une demande
Dire non
Exprimer une critique constructive
Recevoir une critique justifiée
Recevoir une critique injustifiée
Réagir aux comportements parasites

Eclairage de la process com

- 1) Mieux se connaître grâce à la Process com
Avez-vous pris conscience de vos points forts ? Connaissez-vous vos besoins psychologiques ? Avez-vous identifié vos mécanismes de stress ?
La Process communication est un modèle de connaissance de soi et de compréhension de ses modes de fonctionnement qui permet de mieux veiller à son écologie personnelle.
Contenu du cours : partie théorique et exercice interactif en binôme
- 2) Mieux communiquer grâce la Process com
La Process com est un outil qui permet de comprendre que nous avons tous des modes de perception du monde différents. En apprenant à identifier les types de personnalité chez nos interlocuteurs, nous pouvons alors chercher à améliorer nos communications.
Contenu du cours : partie théorique et exercice en groupe

Format des activités

TBD.

Pré-requis

TBD.

Évaluation des AAVs

Partiel final

Calcul de note finale

Pariel = 100% de la note.

Références et bibliographie

TBD.

[LV] Langue vivante

Niveau	L3
Semestre	SI6
UE	21_IS6TC_MSH2
Coeff UE	1/5
Référents	Steve Frank Max Testemale

EQ-CM	0
RMD	0
TDP	12
FF	12
VHT	16

Résumé et contexte

Ce cours vise à augmenter la capacité des étudiants de comprendre un échange ou discours oral quotidien, et des textes courts. Dans tous ces cas, les étudiants doivent montrer qu'ils maîtrisent du vocabulaire provenant aussi bien de la vie courante que des domaines professionnels, ce dernier grâce au travail sur le TOEIC. L'aspect compréhension orale (listening comprehension) du test aura une place importante.

Nous poursuivrons la révision de grammaire. Par le biais d'un exposé, individuel ou en groupe, les étudiants vont travailler leurs compétences de prise de parole en public. Une sélection de sujets leur seront proposée par les professeurs. TOEIC officiel en fin de semestre.

Acquis d'apprentissages visés (AAVs)

A l'issue de ce cours, les étudiants sont capables de :

- Démontrer l'augmentation de leur niveau dans une langue étrangère.

Pour l'anglais :

- Bien comprendre l'anglais courant, parlé et écrit, à la vitesse normale.
- Identifier les points essentiels d'un contenu oral ou écrit.
- Démontrer l'augmentation du niveau de compréhension orale d'anglais.

Plan cours

- Si TOEIC INF à 800

- Révision de grammaire et de vocabulaire professionnel
- Compréhension orale
- Sujets d'actualités (par les exposés)

- Si TOEIC sup à 800

Apprentissage d'une nouvelle langue étrangère sur la plateforme Rosetta stone

Format des activités

Formats des activités pédagogiques :

- Travaux dirigés si TOEIC INF à 800
- Plateforme d'apprentissage si TOEIC sup à 800

Pré-requis

Anglais niveau B2-C1. Rappel du niveau C1 : un vocabulaire riche. L'élève manie la langue correctement et peut produire un texte clair et détaillé avec un minimum de fautes.

Évaluation des AAVs

Anglais :

- Examens Moodle d'une heure en milieu et fin d'enseignement visant à évaluer la compréhension orale. Les documents ne sont pas autorisés
- Exposé en anglais (suivi)

Rosetta Stone :

- Évaluation continue et évaluation finale

Calcul de note finale

Anglais :

Epreuve oral 1 : 30% sur Moodle (plateforme interne en ligne)

Epreuve écrite 2 : 40% sur Moodle (plateforme interne en ligne)

Suivi / contrôle continu : 30%

Rosetta Stone :

1h de travail = 1 point

Evaluation finale (sur 5 points)

Références et bibliographie

<https://login.rosettastone.com/>

Understanding and Using English Grammar by Betty Azar

La Bible officielle - Test TOEIC

[COMM2] Communication écrite

Niveau	L3	EQ-CM	10
Semestre	SI6	RMD	0
UE	21_IS6TC_MSH2	TDP	2
Coeff UE	1/5	FF	12
Référent	Hélène Vaury	VHT	16

Résumé et contexte

Ce module de communication écrite va permettre de mieux réaliser des notes, comptes rendus.

Ces modules vous enseigneront la méthode imparable pour être lus jusqu'au bout, pour captiver.

Mais ils vous apprendront aussi les procédés pour toucher tel lectorat ou tel autre.

Ils vous inculqueront le ton, la ponctuation et tous ces termes du langage courant qu'il faut bannir à l'écrit.

Enfin, vous apprendrez à structurer vos propos, tâche qui se révèle être la pierre angulaire de toute bonne rédaction, entre autres choses...

Acquis d'apprentissages visés (AAVs)

Produire des écrits en situation professionnelle

Clarifier et adopter les messages écrits à l'interlocuteur

Hiérarchiser et structurer les écrits

Délivrer un message synthétique sans en dénaturer le sens

Rédiger des écrits impactants

Plan cours

- L'orthographe dans la communication écrite
- La grammaire dans la communication écrite
- Le choix du vocable dans la communication écrite
- La recherche d'informations dans la communication écrite
- Les chiffres dans une communication écrite
- Le ton dans une communication écrite
- La ponctuation dans une communication écrite
- Le but de la rédaction
- Le choix de l'angle dans la communication écrite
- Le bon choix de mots pour le bon lecteur
- Le plan dans la communication écrite
- La forme dans la communication écrite
- Écrire un mail et/ou un courrier professionnel
- Le respect des sources dans une communication écrite
- A bannir dans une communication écrite
- A faire absolument après rédaction d'une communication écrite
- Avantages et inconvénients d'une communication écrite

Format des activités

Vidéo

Travaux pratiques de mise en application

Pré-requis

[AUCUN]

Évaluation des AAVs

Deux TP de mise en pratique.

Calcul de note finale

50% premier TP 50% deuxième TP

Références et bibliographie

ionisx.com

[CPT] Conception

Niveau	L3
Semestre	SI6
UE	21_IS6TC_MSH2
Coeff UE	1/5
Référent	Juliette Chansard

EQ-CM	6
RMD	0
TDP	6
FF	12
VHT	18

Résumé et contexte

En logique, une conception est un ensemble de contenu de pensée, qui, lorsqu'elle est appliquée à un objet, peut former une proposition. En linguistique, le concept représente le signifié, c'est-à-dire le sens du mot, tandis que le mot lui-même constitue son signifiant. Le concept est un terme abstrait qui se distingue donc de la chose désignée par ce concept. En anglais on traduit le mot conception par "design". "Le design ce n'est pas seulement ce à quoi ça ressemble, le design est comment ça marche" : Steve Jobs.

L'objectif sera d'expérimenter des outils utiles à l'identification du besoin jusqu'à la proposition d'une solution répondant en tout ou partie à ces besoins.

Nous traiterons un outil par thématique dans le cadre d'un exercice ciblé, en groupe, d'une durée de 60 minutes. Un livrable précis sera demandé à l'issue de celui-ci.

Ce cours permettra aux élèves d'ING1 d'aborder leur projet PING plus sereinement avec une première initiation aux approches centrées utilisateurs.

Acquis d'apprentissages visés (AAVs)

A la fin de la session de conception, les élèves d'ING1 seront capables de répondre à une problématique:

- d'émettre des hypothèses en apprenant à chercher puis analyser des informations, les partager, être capable de s'aligner sur consensus, à recueillir des données qualifiées et les analyser afin d'identifier une problématique ;
- d'élaborer un modèle à partir d'une réflexion organisée, d'abord divergente puis convergente menant à la scénarisation d'une expérience immersive et aboutissant à un produit minimum viable ;
- de tester ce modèle grâce à des scénarios de test précis et évaluables.

Plan cours

Thématique 1 : Stratégie d'acculturation

TP Atelier en groupe // sur une thématique donnée :

CM : intérêt et principe d'une bonne stratégie d'acculturation

Objectifs

- chercher et analyser des informations
- partager
- aligner
- enrichir ses connaissances

Thématique 2 : Les interviews

TP Atelier en groupe type jeu de rôle

CM : technique pour réaliser une interview centrée sur une thématique

Objectifs :

- recueillir des datas qualifiées
- analyser les datas
- identifier une problématique

Thématique 3 : Idéation

TP Atelier en groupe : comprendre et intégrer la démarche de l'état de l'art à partir d'une problématique donnée

Objectifs :

- réfléchir individuellement
- restituer et discuter ensemble (divergence)
- choisir et organiser (convergence)

Thématique 4 : Narration Interactive

TP Atelier en groupe : rédiger une narration interactive sur une thématique donnée

CM : Historique de la narration et technique de conception

Objectifs

- apprendre à créer un storytelling
- rythmer son expérience
- rendre l'expérience immersive

Thématique 5 : Test [scénario et grille d'évaluation]

TP Atelier en groupe : réalisation de test et analyse des feedbacks

CM les différents types de test et les techniques de conception

Objectifs :

- créer un scénario test
- mettre en place un plan de test
- analyser les feedbacks

Thématique 6 : Prototypage

CM : Comment prototyper rapidement un MVP sans code

TD : Initiation à un outils de prototypage no code

Objectifs :

- manipuler un outil de prototypage no code
- créer un MVP

Format des activités

6 thématiques traitées, réparties en 6h de TP, 5h de CM et 1h de TD.

Pré-requis

[AUCUN]

Évaluation des AAVs

Un livrable sera demandé à la fin de chaque TP thématique [6 livrables au total].

Calcul de note finale

Grille d'évaluation de compétence en cours de création. Celle-ci viendra s'ajouter à celle du PING.

Références et bibliographie

[https://fr.wikipedia.org/wiki/Biais_cognitif#/media/Fichier:The_Cognitive_Bias_Codex_\(French\)_-_John_Manoogian_III_\(jm3\).svg](https://fr.wikipedia.org/wiki/Biais_cognitif#/media/Fichier:The_Cognitive_Bias_Codex_(French)_-_John_Manoogian_III_(jm3).svg)

<https://podcast.designmasterclass.fr/episode/1/user-research-recruter-animer-co-construire>

<https://medium.com/@quentin.kuntzmann/tuxt-2-cerveau-et-gamification-7dbb1ff273d4>

Méthodes de design UX : 30 méthodes fondamentales pour concevoir des expériences optimales Par Carine Lallemand, Guillaume Gronier

<https://www.nngroup.com/>



[ETHIX] Ethique

Niveau	L3	EQ-CM	8
Semestre	SI6	RMD	0
UE	21_IS6TC_MSH2	TDP	4
Coeff UE	1/5	FF	12
Référent	Alice Louis	VHT	16

Résumé et contexte

TBD.

Acquis d'apprentissages visés (AAVs)

A l'issue de ce cours, les étudiants sont capables de :

- Distinguer les différents éléments de l'éthique
- Sensibiliser les équipes aux enjeux sociétaux des TIC

Plan cours

I- Théorie = Amphi

Définitions, notions voisines

Chartes, règlementations

Ethique by design, éthique des usages.

II- Pratique = TD

Les nouveaux métiers

Cas d'usages des algorithmes de l'IA

Ethique appliquée aux autres technologies émergentes.

Format des activités

Cours et TD.

Pré-requis

[AUCUN]

Évaluation des AAVs

Travail de groupe et participation.

Calcul de note finale

80% travail de groupe 20% participation.

Références et bibliographie

CIGREF, « Une éthique du numérique », Entreprise 2020, le 19/09/19, disponible sur : <http://www.entreprise2020.fr/une-ethique-du-numerique/>

Une bibliographie complète sera donnée en début de cours

[21_IS5TC_CCE2] Culture et connaissances pour l'entreprise 2

Semestre	S6
Crédits	2
VHT	54
CODE	21_IS6TC_CCE2
Coord.	Hélène Vaury

Objectifs de l'UE

Aborder les différents items qui régissent la vie d'une entreprise.

Compétences

- Être conscient des impératifs de la santé et sécurité au travail
- S'insérer dans un contexte de gestion de projet
- Formaliser son projet professionnel
- Comprendre et identifier les interactions avec le marketing selon l'entreprise et les différentes activités de celle-ci

Modalités de validation de l'UE

Validation de l'UE par une moyenne supérieure à 10 et une note supérieure à 6 dans chaque ECUE.

[MSST] MOOC Sécurité et santé au travail

Niveau	L3	EQ-CM	0
Semestre	SI6	RMD	0
UE	21_IS6TC_CCE2	TDP	0
Coeff UE	1/4	FF	0
Référent	Hélène Vaury	VHT	8

Résumé et contexte

Ce MOOC, enrichi de témoignages vidéos de professionnels (dirigeants, ergonome d'un service de santé au travail, contrôleur de sécurité), permet d'acquérir des connaissances de base en santé et sécurité au travail et de sensibiliser sur la prévention des risques professionnels.

Acquis d'apprentissages visés (AAVs)

A l'issue de ce cours, les étudiants sont capables de :

- Identifier les principes généraux de prévention
- Identifier les liens entre travail et santé
- Identifier les principales méthodes d'analyse des risques professionnels.

Plan cours

Le parcours "Acquérir des bases en prévention des risques professionnels" se compose de 4 modules indépendants :

- Module "S'initier à la prévention des risques professionnels" : Valeurs et enjeux, principes généraux de prévention, acteurs internes et externes, définitions des accidents du travail et des maladies professionnelles.
- Module "Comprendre les liens entre travail et santé" : Composantes d'une situation de travail, travail prescrit et activité de travail, déterminants de l'activité, variabilité et aléas, santé au travail.
- Module "Comprendre l'accident de travail" : Réactions immédiates, pluricausalité, faits, enquête, mécanisme de l'accident, arbre des causes, analyse et prévention.
- Module "Participer à l'évaluation des risques professionnels" : Pourquoi évaluer les risques, s'engager dans une démarche, identifier les risques, analyser les risques, caractériser les risques, rechercher des mesures de prévention, planifier les actions.

Un mail sera envoyé avec une invitation à se connecter à la plateforme e-learning de l'INRS

Format des activités

Plateforme e-learning de l'INRS

La durée de chaque module est estimée à 2h.

La consultation est libre et s'effectue en fonction de son rythme (dans la limite de 90 jours maximum déclenchée lors de la première connexion). Hors prérequis où les quatre modules doivent être validés, il est possible de suivre ces modules dans l'ordre de son choix.

Les modules d'autoformation sont composés d'un contenu interactif en ligne : fictions animées, écrans d'information, exercices auto-évaluatifs, témoignages de professionnels.

Pré-requis

[AUCUN]

Évaluation des AAVs

Un test de validation à la fin de chaque module.

Un score de 0 à 100% de réussite pour chaque module

Calcul de note finale

Moyenne de tous les tests rapportée sur 20.

Références et bibliographie

<https://www.inrs.fr/>

[REPP] Projet professionnel

Niveau	L3	EQ-CM	6
Semestre	SI6	RMD	0
UE	21_IS6TC_CCE2	TDP	0
Coeff UE	1/4	FF	6
Référent	Laurent Trébulle	VHT	16

Résumé et contexte

Ce module, qui fait suite à l'atelier « semaine professionnelle », a pour objet la rédaction du projet professionnel individuel des étudiants et l'élaboration du chemin optimal (stages, choix de majeure, etc.) pour l'atteindre.

Acquis d'apprentissages visés (AAVs)

A l'issue de ce module, les étudiants sont capables de :

- Formaliser leur projet professionnel ;
- Ordonner leurs priorités par rapport à leurs aspirations ;
- Identifier le type d'entreprise dans lesquelles ils pourraient s'accomplir ;
- Concevoir le rétroplanning pour y parvenir en repérant les étapes clés ;
- Développer leur présence sur les réseaux sociaux professionnels en utilisant une expression fluide et un vocabulaire professionnel.

Plan cours

1. Travail sur soi : cette partie d'autoévaluation a pour but de permettre à l'étudiant de cerner sa personnalité, ses aspirations ainsi que ses compétences et d'analyser ses expériences.
2. Travail sur le secteur d'activité, sur les métiers et sur les entreprises : cette partie a pour but d'identifier des métiers dans lesquels l'étudiant peut se projeter et d'identifier des entreprises susceptibles de proposer ce type de métiers.
3. Organiser la mise en œuvre du projet professionnel : cette partie a pour but l'identification, par l'étudiant, des actions à entreprendre pour obtenir un stage d'une des entreprises identifiées.
4. Utiliser les réseaux sociaux professionnels : cette partie consiste à avoir un profil à jour sur LinkedIn, à gérer son réseau et à savoir repérer et contacter des personnes utiles pour atteindre son objectif.

Format des activités

Conférences.

Évaluation des AAVs

A l'issue de cette semaine de conférences, l'étudiant rédige son Projet Professionnel qui sera corrigé par un enseignant. Il est ensuite encouragé à le mettre à jour et à le faire évoluer à l'issue de son premier stage. Rapport en français

Calcul de note finale

Rapport : 90% + participation : 10%

Références et bibliographie

Plan de rédaction du rapport disponible sur l'intranet des RE.

[MARK] Marketing

Niveau	L3
Semestre	S16
UE	21_IS6TC_MSH2
Coeff UE	1/5
Référent	Jean-Luc Viard-Gaudin

EQ-CM	12
RMD	0
TDP	0
FF	12
VHT	12

Résumé et contexte

"Avoir une compréhension de base du marketing permet de libérer son potentiel d'ingénieur développeur". Plus de sensibilité aux besoins de l'utilisateur, plus de créativité pour générer des idées, plus de rigueur dans la capacité de raisonner dans des situations inattendues.

Acquis d'apprentissages visés (AAVs)

Identifier les différents rôles du marketing

Déterminer les outils du marketing ainsi que les canaux à utiliser selon les différents cas en marketing

Mettre en œuvre un plan d'action marketing

Acquérir le bon vocabulaire en matière stratégique, en fonction de l'entreprise, des présentations de ses activités

Plan cours

1) Comprendre les fondamentaux et découvrir les faces cachées du marketing

Son importance, sa place, son rôle et son omniprésence dans la vie des entreprises et de la société en général

Les différentes composantes du marketing

Comprendre ses audiences et ses cibles

Les enjeux du marketing: son influence, ses limites

Les exemples des best of/ des échecs

Études de cas réels

Les liens entre l'ingénierie informatique et le marketing

Exercices ludiques de marketing en classe

2) La communication-marketing

Qu'est-ce que la communication ? Principes et théories

Comment communiquer ? Le cadre stratégique, les outils.

La réalité et l'omniprésence de la communication dans le monde

Les praticiens de la communication

Pourquoi un bon développeur doit savoir bien communiquer.

Étude de cas de communication

3) Le marketing, un levier pour développer sa créativité

Les théories et fondamentaux des techniques de créativité

Les applications de techniques de la créativité

De stratégies de marketing créatives

Des exemples de campagne "créativement" brillantes

Exercice de créativité en classe

4) Les terrains de jeu du marketing et de la communication, les outils et techniques

Les disciplines: la publicité, la promotion, le marketing direct, le sponsoring, le merchandising, les RP, le packaging, le digital, le social media, les applications, le CRM, le Search, etc.

Les canaux du marketing et les media

L'UX Design, nouvelle frontière entre marketing et développement informatique

Exercices de développement d'une campagne: choix des tactiques, des médias

Format des activités

Cours magistraux

Pré-requis

[AUCUN]

Évaluation des AAVs

Contrôle final.

Calcul de note finale

100% note finale.

Références et bibliographie

Marketing pour ingénieurs, Alain Goudey et Gaël Bonnin Mercator, 12 éd. Jacques Lendrevie, Julien Lévy et De Baynast

Principes de marketing, Philip Kotler, Gary Armstrong, Emmanuelle Le Nagard-Assayag et Thierry Lardinois

[GEPR00] Gestion de projet

Niveau	L3
Semestre	SI6
UE	21_IS6TC_MSH2
Coeff UE	1/4
Référents	Jean-Jacques Guelin Pierre Roussenq

EQ-CM	8
RMD	0
TDP	3
FF	11
VHT	16

Résumé et contexte

- Fournir les bases méthodologiques de la gestion de projet
- Comprendre le cycle de vie d'un projet et les outils et techniques à mettre en œuvre à chaque étape
- Voir les similitudes, les différences, les avantages et inconvénients des méthodologies de gestion de projet classiques (prédictives) et la méthode AGILE
- Être capable d'appliquer ces techniques dans leur prochaines activités (PFE, stages...)

Acquis d'apprentissages visés (AAVs)

A l'issue de ce cours, les étudiants sont capables individuellement dans le cadre du développement d'un projet en équipe:

- Comprendre le contexte dans lequel il intervient
- Connaître les différentes terminologies et méthodologies. Utiliser le vocabulaire et des concepts adéquats
- Cadrer un projet, Organiser le travail, les ressources et planning d'un projet et Piloter le déroulement d'un projet
- Connaître le cycle de vie d'une application
- Connaître les rôles et les responsabilités de chacun au sein d'un projet
- Identifier des risques
- Connaître le contenu d'un Project Charter
- Connaître les méthodes pour collecter et formaliser les exigences du produit à construire ; planifier un projet : la construction d'un Gantt ; le découpage d'un Product backlog en Sprint et Release
- Connaître les méthodes pour collecter et formaliser les exigences du produit à construire

Plan cours

MODULE 1 (2H) : Fondamentaux de la Gestion de projet ; initialisation d'un projet ; Cadrage Périmètre

- Qu'est-ce qu'un projet, les objectifs de la gestion de projet, méthode prédictive vs Agile
- Phase d'initialisation : Comprendre le sens du projet : Réaliser un « Project Charter »
- Définition du Périmètre : Partie 1 : les exigences fonctionnelles du produit attendu.

MODULE 2 (2H) : Compléter le Périmètre ; Planifier le projet dans la Méthode Classique (Prédictive)

- Définition du Périmètre : Partie 2 : les exigences non fonctionnelles (technique, qualité, projet)
- Parcourir le processus et les méthodes pour passer du cahier des charges au Planning de référence du projet : WBS, découpage en activité, séquençement, évaluation de charges et de durées
- Construire un Gantt

TD #1 : (1h30) Construire un planning sur un projet de développement.

- Construire une arborescence de tâches (WBS) couvrant le périmètre d'un projet
- Détailler tout ou partie de cette WBS en activités ; estimer les charges et les durées.
- Construire le GANTT associé avec MS Project ou GanttProject à défaut.

MODULE 3 (2H) : Planification d'un projet de développement avec la Méthode Agile ;

- Ordonner, pondérer un Product Backlog : Découpage en Release puis en Sprints.
- Sprint Planning : Etablir un Sprint Backlog et un ScrumBoard.
- Exécuter un Sprint: Daily Scrum, Sprint Review, Sprint Retrospective, Product Backlog Grooming

TD #2 : (1h30) Construire un planning d'un projet de développement en mode AGILE.

- Construire une arborescence de tâches (WBS) en identifiant les parties traitables en AGILE et celles qui relèvent de tâches plus traditionnelles couvrant le périmètre d'un projet.
- Construire le plan d'une release dans un Product Backlog : périmétrer, pondérer les User Stories, découper en Sprints)

- Construire une Timeline projet incluant la partie Agile, et les autres tâches.
- Planifier un Sprint.

MODULE 4 (2H) : Communication ; Piloter l'exécution d'un projet en mode classique ; Gestion des risques Planifier la communication du projet

- Méthode Classique : Pilotage du projet, gestion des travaux et du temps ; gestion des changements.
- Gestion des risques dans le projet (identification, Evaluation, Réponses), gestion continue.

Format des activités

- a) Cours magistraux
- b) Illustration sur un cas concret en mode fil rouge.
- c) Réalisation de 2 TDs pour mise en application de la planification projet

Pré-requis

[AUCUN]

Évaluation des AAVs

QCM et présence en TD

Calcul de note finale

80% pour le QCM 20% présence participation aux TD

Références et bibliographie

PMBOK du Project Management institute <https://www.pmi.org/pmbok-guide-standards>

SCRUM guide <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-French.pdf>

Karl Wiegers and Joy Beaty "Software requirement Third ed."

Alistair Cockburn « Rédiger des cas d'utilisation efficaces »

Une méthode agile pour vos projet <https://www.editions-eni.fr/livre/scrum-une-methode-agile-pour-vos-projets-3e->

