

Pré-requis pour les cours M2 E-Services

1 GLIHM

1.1 ODEVA, Sébastien Lucas

Le préalable au cours est une bonne connaissance de Git qui couvre la création de branche, le merge (la résolution de conflit), les commandes outils (git log, git status) ainsi que l'interaction avec les dépôts centraux (pull, fetch, push).

Le moyen de vérifier cette bonne compréhension est de comprendre l'ensemble des TP basiques sur GIT : <https://github.com/seblucas/odeva/blob/master/odeva.md#git--basique>.

Je vais embrayer directement les TP avancés

Ressources additionnelles :

- En écrit : les chapitres 1, 2, 3 et 4 du livre Pro Git : <https://git-scm.com/book/en/v2> pour la version anglaise ou <https://git-scm.com/book/fr/v2> pour la version française
- Pour ceux qui partent de plus loin (aucune connaissance Git), il y a une vidéo que j'aime bien : <https://www.youtube.com/watch?v=WBg9mlpzEYU> (anglais).

2 IFI

2.1 Java EE, Julien Wittouck (FI), Guillaume Dufrêne (FA)

2.1.1 Julien Wittouck (FI)

- bases de HTML/CSS
- Java avancé (types génériques, réflectivité)
- Bases de testing (JUnit / Mockito)
- Java EE de base (Servlets, JSP) => cours de CAR.
- Manipulation de Maven (pom.xml, CLI, phases de build)
- Git

Pour un rappel cf. https://juwit.github.io/ifi-2019/cours/01-rappels_de_car/01-rappels.html et <https://juwit.github.io/ifi-2019/cours/01-maven/01-maven.html>

Logiciels :

- maven (déjà installé normalement)
- git (déjà installé)
- java 11 minimum (il y avait un java installé, mais la version 8 il me semble)
- IntelliJ IDEA serait un vrai plus (tous les étudiants l'utilisent depuis 2/3 ans, et ils peuvent disposer d'une licence gratuite)
- docker serait également pratique pour les aspects base de données etc...

2.1.2 Guillaume Dufrêne (FA)

Les basiques ... (parfois oubliés)

Connaissances générales de la POO

Connaissance du langage Java

Savoir compiler (avec javac) et lancer (avec java) un programme simple, savoir ce qu'est un classpath et comment le modifier.

https://gdufrene.github.io/mooc_je_spring/java.html

Les notions abordées en CAR en 1er année

- Relation Client - Serveur

- Architecture 3 tiers / N tiers

- JDBC : https://gdufrene.github.io/mooc_je_spring/jdbc.html

- Servlet / JSP : https://gdufrene.github.io/mooc_je_spring/servlet.html

Logiciels :

Les logiciels peuvent être installés par les étudiants, toutefois, s'il y a déjà une version de java et de tomcat (c'est normalement le cas) évidemment c'est plus simple pour eux.

Concernant l'IDE, ils sont libres de choisir celui qui leur convient ... je préfère qu'ils choisissent celui avec lequel ils sont à l'aise.

S'il y a une version récente de maven installé c'est bien (version 3.6.x) sinon idem ... ils pourront le faire.

Ils ont besoin de git ... c'est déjà installé avec le système.

2.2 .NET, Jounad Tourvieille

(si possible quelques bases de C#)

Logiciels :

visual studio 2017 (déjà installé) ou visual studio 2019. Les deux en version community (<https://visualstudio.microsoft.com/fr/>).

2.3 PHP, Kevin Dunglas / Grégoire Hébert

La connaissance de PHP, des designs pattern et un linux avec PHP 7 à jour sont les seuls prérequis.

3 TAC

3.1 Framework JS, Thomas Verhoken

- Avoir Git sur la machine de développement.
- Avoir les bases sur le langage Javascript (syntaxe, principe de fonctionnement, EcmaScript 6 et +, programmation asynchrone...). Voici un lien qui regroupe

l'essentiel des concepts sur le langage, des bases jusqu'aux notions avancées
: <https://github.com/leonardomso/33-js-concepts>

Logiciels :

GIT

3.2 Web avancé, Xavier Le Pallec

Bases de Javascript

3.3 Android, Maxime Savary-Leblanc

Java

Logiciels :

Dernière version d'Android Studio

3.3.1 SVI, José Rouillard

Logiciels

App Inventor (à installer sur les machines personnelles)