

Interfaces Hommes-Systèmes
Human-Systems Interfaces

Code cours <i>Course code: IHS</i>		Crédits ECTS <i>ECTS Credits: 2,5</i>
Département <i>Department</i>	: IA	Cours <i>Lectures</i> : 12H30
Coordonnateurs <i>Lecturers</i>	: M. RICHARD	T.D. <i>Tutorials</i> : 12H30
Période <i>Year of study</i>	: 3 ^{ième} année <i>3rd year</i>	T.P. <i>Laboratory sessions</i> :
Semestre <i>Semester</i>	: 5 ^{ième} semestre – A <i>5th semester - A</i>	Projet <i>Project</i> :
Evaluation <i>Assessment method(s)</i>	: 1 examen écrit, <i>1 written exam</i>	Non encadré <i>Homework</i> :
Langue d’instruction <i>Language of instruction</i>	: Français <i>French</i>	Horaire global <i>Total</i> : 25H
Type de cours <i>Type of course</i>	: Obligatoire <i>Compulsory</i>	<i>hours</i>
Niveau <i>Level of course</i>	: Avancé <i>graduate</i>	

Compétences attendues :

À l’issue de ce module, l’étudiant maîtrisera :

- La conception et l’implémentation d’un interface graphique de type client lourd en JavaFx
- La conception et l’implémentation d’un interface graphique de type client riche en VueJS

Pré-requis : Module Conception Logiciel

Contenu :

Après avoir présenté les principaux modèles d’architectures d’interfaces (MVC, MVP, MVVM), et les différents types de clients (lourd, léger et riche), ce module s’intéressera à la mise en œuvre de ces concepts dans 2 cadres :

1. Conception et Implémentation d’un client lourd & lien avec l’architecture logicielle sous-jacente : Cette partie détaillera les architectures MVC et MVP tout d’abord en mono-agent puis en multi-agent. Elle présentera également les différents branchements possibles avec l’architecture logicielle du noyau fonctionnel de l’application. La boîte à outils graphique JavaFx sera utilisé ici.
2. Conception et Implémentation d’un client lourd & lien avec l’architecture logicielle sous-jacente : Dans ce chapitre, après présentation des spécificités de la programmation web, nous présenterons les méthodes de conception et d’implémentation de ce type de client en s’appuyant sur le framework VueJS. Enfin les méthodes de liaison avec une architecture micro-services seront étudiées.

Toutes ces notions seront largement mises en œuvre au cours des différentes séances de travaux dirigés.

Bibliographie :

Expected competencies:

At the end of this module, the student will master:

- The design and implementation of a thick client GUI in JavaFx
- The design and implementation of a rich client GUI in VueJS

Prerequisites: Module Conception Logiciel

Content:

After presenting the main interface architecture models (MVC, MVP, MVVM), and the different types of clients (heavy, thin and rich), this module will focus on the implementation of these concepts in 2 frameworks:

1. Design and Implementation of a thick client & link with the underlying software architecture: This part will detail the MVC and MVP architectures first in single-agent and then in multi-agent. It will also present the different possible connections with the software architecture of the application's functional core. The JavaFx graphical toolkit will be used here.
2. Design and implementation of a thick client & link with the underlying software architecture: In this chapter, after presenting the specificities of web programming, we will present the design and implementation methods of this type of client based on the VueJS framework. Finally, we will study the methods of linking with a micro-services architecture.

All these notions will be widely implemented during the different tutorial sessions

Recommended reading: