Guide pédagogique

Module « Programmation distribuée »

Option IL – 9.7 *(4 crédits ECTS)*

Place du module et enjeux

Les techniques de développement d'applications ont évolué pour répondre à des exigences de qualité et de productivité toujours plus importantes. La réutilisation est une solution concrète à cette problématique. Elle peut être mise en œuvre par la programmation distibuée qui permet de développer des applications serveur chargées de fournir des fonctionnalités réutilisées par de multiples applications clientes, éventuellement distantes, au travers d'un réseau informatique.
Ce cours présente un ensemble de techniques permettant de réaliser ces interactions client-serveur entre applications distribuées, depuis les échanges de données (sockets) jusqu'aux composants et services métier (serveurs applicatifs et architectures n-tiers) en passant par les appels distants de fonctionnalités (middlewares). Il propose également des notions de programmation Web avancées.

Teaching guide and syllabus

« Distributed programming » module

SE option – 9.7 *(4 ECTS credits)*

Subject matter importance and associated issues

Software development techniques have evolved to meet ever higher quality and productivity requirements. Software reuse is an actual solution to this issue. Reuse can be achieved thanks to distributed programming. It enables to implement applicative servers that provide functionalities (services) reused by multiple client applications, possibly from a remote location through a network.
This course presents different client-server programming techniques, from data exchange (sockets) to business components and services (application servers and n-tiered architectures) and remote procedure calls (middlewares). Some advanced web programming is also addressed as part of n-tiered architectures.

Responsable : Sylvain Vauttier
Téléphone : 04 34 24 62 85
Courriel : sylvain.vauttier@mines-ales.fr

IMT Mines Alès
École Mines-Télécom

ENSEIGNEMENTS ACADÉMIQUES	**Volume horaire**	Détail des coefficients	Crédits
Programmation distribuée	**50 h**		
○ Client-serveur	20	1	4
○ Architectures n-tiers	30	1	

Matière 1

Titre de la matière : Client-serveur	
Code : 2IA-il 9.7.1	**Titre du module :** «Programmation distribuée »
Semestre : S9	**Cursus de rattachement :** Département 2IA, option IL

Heures présentiel	Heures total	Cours	TD	TP	Projet	Contrôles	Travail personnel	Coef /module	ECTS
20	35	6			14		15	1	1,6

Résumé	Ce cours présente les concepts de la programmation distribuée et les techniques de développement d'architectures client-serveur.

Responsable	Sylvain Vauttier – LGI2P/IMT Mines Alès
Équipe enseignante	François Pfister – ConnectHive Sylvain Vauttier – LGI2P/IMT Mines Alès

Mots-clés	Programmation distribuée, Client-Serveur, Sockets, Middleware
Prérequis	Programmation en langage Java

Contexte et objectif général :

Ce cours expose les concepts de la programmation distribuée et le développement d'architectures client-serveur.

Il présente les différentes techniques de mise en œuvre actuelles, des plus simples, qui servent de couches basses, aux plus élaborées, qui permettent la mise en œuvre d'architectures et de mécanismes de gestion sophistiqués :

- les sockets, permettant les échanges de données,
- les middleware, permettant l'invocation distante de fonctionnalités et les échanges d'objets

Il aborde les variations des architectures client-serveur (centralisées, hiérarchiques, pair-à-pair).

Programme et contenu :

- Principes des architectures C/S
- Communication sous IP (sockets)
- Serveurs multi-tâches
- Remote Procedure Call
- Formes d'architecture C/S
- Réutilisation et composants logiciels
- Bus logiciel et composants distribués
- Mise en place d'une application serveur RMI
- Mise en place d'une application cliente RMI

Méthode et organisation pédagogique :

Les enseignements sont prévus pour 30 élèves. Les TP seront réalisés sur les ordinateurs personnels de ces derniers.

Le découpage est prévu comme suit :

- 6h de cours
- 9h de projet (+ travail personnel)

Acquis d'apprentissage visés :

- compréhension des architectures client-serveur
- développement d'applications distribuées à l'aide de sockets
- développement de composants distribués à l'aide d'un middleware

Évaluation : évaluation sur projet (coef 1)

Retour sur l'évaluation fait à l'élève : retours aux élèves pendant la soutenance.

Support pédagogique et références :

1 Polycopié

Matière 2

Titre de la matière : Architectures n-tiers	
Code : 2IA-il 9.7.2	**Titre du module :** «Programmation distribuée »
Semestre : S9	**Cursus de rattachement :** Département 2IA, option IL

Heures présentiel	Heures total	Cours	TD	TP	Projet	Contrôles	Travail personnel	Coef /module	ECTS
30	55	9			21		25	1	2,4

Résumé	Ce cours présente les concepts des architectures logicielles multi-tiers et le framework industriel de référence JEE (Java Enterprise Edition).

Responsable	Sylvain Vauttier – LGI2P/IMT Mines Alès
Équipe enseignante	François Pfister - ConnectHive Christelle Urtado – LGI2P/IMT Mines Alès Sylvain Vauttier – LGI2P/IMT Mines Alès

Mots-clés	Composants logiciels, Architectures logicielles 3-tiers, JEE
Prérequis	- programmation en langage HTML - programmation en langage JAVA - programmation de Bases de Données

Contexte et objectif général :

Ce cours expose les concepts de la programmation d'architectures logicielles multi-tiers, c'est-à-dire composées d'applications clientes utilisant les services fournis par des serveurs applicatifs interagissant avec des bases de données.

Il présente, au travers de JEE, un exemple de framework facilitant le développement de ce type d'architectures.

Il aborde également le développement de clients lourds (applications distantes) ou légers (pages web dynamiques).

Programme et contenu :

- Infrastructures n-tiers (Java Enterprise Edition)
- Tiers présentation : technologies Java (JSP, Servlet, serveur web Tomcat pour le développement de clients légers et RMI/JNDI pour les clients lourds)
- Tiers métier : composants EJB et serveurs applicatifs Java Enterprise
- Tiers-système d'information : gestion du mapping objet-relationnel (JPA, Hibernate)

Méthode et organisation pédagogique :

Les enseignements sont prévus pour 30 élèves. Les TP seront réalisés sur les ordinateurs personnels de ces derniers.

Le découpage est prévu comme suit :

- 9h de cours
- 21h de projet (+travail personnel)

Acquis d'apprentissage visés :

- développement d'architectures n-tiers avec le framework JEE
- développement de clients légers (web) avec les techonologies Java
- développement de clients lourds (middleware) avec les technologies Java
- développement de mappings objet-relationnel avec JPA et Hibernate

Évaluation : projet (coef. 1)

Retour sur l'évaluation fait à l'élève : retours aux étudiants pendant la soutenance.

Support pédagogique et références :

1 Polycopié

Méthode et organisation pédagogique

Il s'agit d'un enseignement relativement pratique essentiellement basé sur la réalisation de projets.

Modalité d'évaluation

Le niveau d'acquisition des compétences sera évalué selon les exigences suivantes :

N° indicateur	Indicateur
1	Connaitre les savoirs formels et pratiques du socle des fondamentaux
2	Exploiter les savoirs théoriques et pratiques
3	Analyser, interpréter, modéliser, émettre des hypothèses, et résoudre

Répartition

Matière	Contrôle	Coefficients	Type de notation	Indicateurs évalués	Chapitres
Client-serveur	Projet	1	Groupe	2	Tous
Architectures n-tiers	Projet	1	Groupe	2	Tous

Engagement de l'étudiant, éthique et professionnalisme

La démarche éthique est définie dans le règlement intérieur de l'établissement. Chaque étudiant s'engage à en prendre connaissance et à la respecter.

Obligation des cours : Présence obligatoire pour tous à chaque séance

Nombre d'heures estimées de travail personnel *:* pour acquérir les compétences demandées, il est nécessaire que l'étudiant consacre minimum 45 min de travail personnel de compréhension et d'approfondissement par séance de cours.
40h de travail personnel sont estimées, principalement dédiées à la définition et réalisation des projets.

Pénalité pour retard : Tout travail remis en retard sans motif valable peut être pénalisé de 1 point par jour de retard.

Équipe enseignante

Nom	Domaine d'expertise	Courriel/Téléphone
Christelle URTADO	Génie logiciel et programmation orientée-objets	christelle.urtado@mines-ales.fr 04 34 24 62 89
François PFISTER	Génie logiciel et programmation orientée-objets	pfister@connecthive.com
Sylvain VAUTTIER	Génie logiciel et programmation orientée-objets	sylvain.vauttier@mines-ales.fr 04 34 24 62 85

ACADEMIC TEACHING	Teaching hours	Coefficients	Credits
Distributed programming	**50 h**		
○ Client-server	20	1	4
○ Multitier architecture	30	1	

Course 1

Title : Client-server	
Code : 2IA-il 9.7.1	**Title of the module :** « Distributed programming »
Semester : S9	**Associated Cursus :** CSAI Department, SE option

Hours of presence	Total hours	Lectures	Seminar	Labs	Project	Testing	Personnal work	Coef /module	ECTS
20	35	6			14		15	1	1.6

Summary	This course presents distributed programming principles and client-server architectures development techniques.

Head	Sylvain Vauttier – LGI2P/IMT Mines Alès
Teaching Team	François Pfister – ConnectHive Sylvain Vauttier – LGI2P/IMT Mines Alès

Key words	Distributed programming, Client-server, Sockets, Middleware
Prerequisites	Java programming

Context and general objective:

This course presents distributed programming principles and client-server architecture development.

Different programming techniques are detailed, from basic ones (sockets), as lower data exchange layers, to more sophisticated mechanisms (object-oriented middlewares), that enable complex architecture implementation and management.

This course also presents different kinds of client-serveur architectures (centralised, hierarchical, peer to peer, …).

Programme and contents:

- client-server architecture principles
- IP communication sockets
- threaded servers
- remote procedure calls
- client-server architectures
- software components
- object-oriented middlewares
- RMI client-server architecture development

Method and pedagogic organisation:

Activities are organised for a full group of 30 students. Labs are realised on the personal computer of the students (BYOD).

Activities are organized as follows:

- courses (6h)
- project (9h)
- personal work (5h)

Targeted skills or knowledge :

- Being knowledgeable about client-server architecture development
- Socket programming in Java
- Object-oriented middleware programming in Java

Evaluation : *project (coef 1)*

Feedback made to the student : Feedbacks at the end of the project.

Teaching material and references :

Photocopied material – textbook – articles - internet…

Course 2

Title : Multitier architectures									
Code : 2IA-il 9.7.2		**Title of the module :** « Distributed programming »							
Semester : S9		**Associated Cursus :** CSAI Department, SE option							
Hours of presence	Total hours	Lectures	Seminar	Labs	Project	Testing	Personnal work	Coef /module	ECTS
30	55	9			21		25	1	2.4

Summary	This course presents the principles of multitier architectures and the reference industrial framework JEE (Java Enterprise Edition).

Head	Sylvain Vauttier – LGI2P/IMT Mines Alès
Teaching team	François Pfister - ConnectHive Christelle Urtado – LGI2P/IMT Mines Alès Sylvain Vauttier – LGI2P/IMT Mines Alès

Key words	Software components, multitier software architecture, JEE
Prerequisites	Java programming Html programming Database programming

Context and general objective:

This course presents the principles of multitier architectures, i.e. architectures composed of client applications that invoke services provided by applicative servers that in turn use databases.

It details the JEE framework as an example of a framework that support the development of this kind of architectures.

It addresses also the development of thick (remote applications) and thin (dynamic web pages) client applications.

Programme and contents:

- Multitier architectures (Java Enterprise Edition)
- Presentation tier : Java technologies (JSP, Servlet, Tomcat web server for thin clients and RMI, JNDI for thick clients)
- Business tier : Java Enterprise bean components and applicative server
- Information tier : object-relation mapping (JPA, Hibernate)

Method and pedagogic organisation:

Activities are organised for a full group of 30 students. Labs are realised on the personal computer of the students (BYOD).

Activities are organized as follows:

- courses (9h)
- project (21h + personal work)

Targeted skills or knowledge :

- Multitier architecture development using JEE framework
- Thin client development (Java web technologies)
- Thick client development (Java middleware technologies)
- Object-relation mapping development (JPA and Hibernate frameworks)

Evaluation : *project (coef 1)*

Feedback made to the student: feedbacks given for each student at the end of the project.

Teaching material and references :

Photocopied material – textbook – articles - internet…

Method and teaching organisation

This is a classical course containing a theoretical part with standard courses and a practical part through a study case project.

Testing procedures

The student's level of knowledge acquisition will be evaluated according to the following points:

N° Indicator	Indicator
1	To know the formal and practical knowledge constituting the foundation of a given field
2	Exploit theoretical and practical knowledge
3	Analyse, interpret, model, hypothesize and solve problems

Grading scheme

Class	Exam	Coefficients	Administration mode	Evaluated Indicators	Chapters
Client-server	Project evaluation	1	Group	2	all
Multitier architectures	Project evaluation	1	Group	2	all

Student commitments, ethics and professionalism

Expectations concerning ethics are defined in the establishment's code of conduct. Each student is expected to know and respect the code of conduct.

Obligatory presence in classes: Students must attend all courses, seminars and labs.

Estimated hours of personal study: in order to acquire the required learning level, students are expected to spend a minimum of 45min of personal study time per course and project supervised sessions.

Late penalties: Late works are subject to penalties as follows: 1 point per day (ratings between 0 and 20).

Teaching team

Nom	Domaine d'expertise	Courriel/Téléphone
Christelle URTADO	Software Engineering and Object-oriented programming	christelle.urtado@mines-ales.fr 04 34 24 62 89
François PFISTER	Software Engineering and Object-oriented programming	pfister@connecthive.com
Sylvain VAUTTIER	Software Engineering and Object-oriented programming	sylvain.vauttier@mines-ales.fr 04 34 24 62 85

Approbation

Ce guide pédagogique entre en vigueur à compter du 7 janvier 2019
Il est porté à la connaissance des élèves par une publication sur le site de l'école

Rédaction	Vérification	Validation
L'enseignant responsable du module : Sylvain VAUTTIER	Le responsable d'UE / de département : Sylvie RANWEZ	Le directeur de l'école, Pour le directeur et par délégation, Le directeur de la DFA / de la DE : Michel FERLUT