

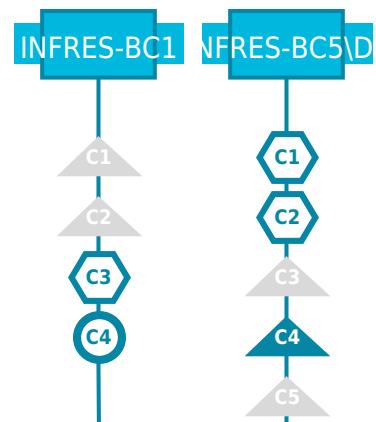
Pourquoi cette UE ?

Le développement logiciel (ou développement informatique) regroupe essentiellement la composante programmation du génie logiciel. Il est lié à un langage de programmation (Java, C, Android...) ou à un contexte de programmation (ordinateur, système embarqué, parallélisme...). Le développement logiciel est le cœur de tous projets informatiques. Ce module présente le développement logiciel dans le cadre de l'environnement de système embarqué, comme l'IOT ou bien android.

Eléments constitutifs de l'UE

	coefficients	
INFRES_8_1a_DL-1 Systèmes embarqués et I.O.T	1	
INFRES_8_1a_DL-2 Informatique mobile: Androïd	1	
Volume d'heures d'enseignement encadré	Volume d'heures de travail personnel	Nombre d'ECTS
65	0	3

Alignement curriculaire

Parmi les compétences visées par la formation, lesquelles sont développées dans cette UE ?

- BC1** L'UE ne contribue pas à ce bloc de compétences
- BC1** L'UE contribue à ce bloc de compétences
- C1** Compétence non adressée dans cette UE
- C1** Compétence mise en œuvre dans cette UE
- C1** Compétence enseignée dans cette UE
- C1** Compétence évaluée dans cette UE
- C1** Compétence enseignée et évaluée dans cette UE

Contexte et enjeux de l'enseignement

Le cours présente les contraintes de conceptions et de développement d'un système embarqué aussi bien suivant une approche bare-metal que via un OS temps réel. Les notions de temps réels sont abordées à travers diverses définitions et cas concrets. Les problématiques liées à la nécessité d'une conception par interruptions, par approches parallèles et concurrentes sont abordées. Les mécanismes d'ordonnancement et algorithmes de tâches sont présentés. La mise en œuvre est réalisée à travers des TP sur microcontrôleur (entrée sortie, interruption, ordonneur, ...). Une réflexion sur l'optimisation de code, la performance et les impacts sur la mobilité (autonomie, coût, ...) sont systématiquement réalisés.

Prise en compte des dimensions socio-environnementales

Prérequis

Ce cours nécessite la connaissance de la programmation C/C++, la connaissance de la structure d'ordinateur, microprocesseur et microcontrôleur ainsi que d'avoir suivi le cours de conception des systèmes.

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	
Cours intégré (cours + TD)	10
TD	
TP	18
Projets	
Travail en autonomie encadré	
Contrôles et soutenances	1
Travail personnel	

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

Connaissance des enjeux et les contraintes d'une application temps réel sur un système embarqué. Connaissances des offres algorithmiques d'ordonnancement de tâches pour une application nécessitant parallélisme avec contrainte de temps. Connaissance des moyens d'implémentation sur systèmes embarqués. Connaissance des moyens d'interfaçage avec un processus industriel et d'en mesurer les performances. Expérience concrète sur projet

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

- Introduction mise en contexte sous forme magistrale de 2 heures.
- Pour la suite, alternance entre cours magistraux, démonstrations et TP de validation.
- Exploitation de 2 supports de cours (Introduction sur les systèmes embarqués et temps réels (30 pages)
- Exploitation du temps réels sur microcontrôleurs (90 pages))

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

Projet visant à concevoir et développer une solution embarquée pilotant un processus, collectant des données suivant des contraintes de temps fortes et échangées avec un poste de collecte et de supervision (PC ou autre).

Plan de cours

- Approche conceptuelle à scrutation/Interruption / OS temps réel
- Contraintes « temps réel »
- Appel Système, accès concurrentiel, atomicité, ordonnancement, notion de tâches temps réel
- Notion de priorité, file d'attente
- Calcul et validation d'ordonnancabilité,
- Algorithme d'ordonnanceur (RM, DM, ED, ...)
- Résolution des problèmes d'interblocage, invention de priorité,..
- Algorithme d'allocation de ressources
- Mise en œuvre sur micro-controleur implémentation sur système nu et système RTOS
- Intégration des contraintes de temps sur Entrée/sortie, CNA, PWM, serial, ... analyse de code et mesure de performances

Méthode et organisation pédagogique :

- Introduction mise en contexte sous forme magistrale de 2 heures.
- Pour la suite, alternance entre cours magistraux, démonstrations et TP de validation.
- Exploitation de 2 supports de cours (Introduction sur les systèmes embarqués et temps réels (30 pages)
- Exploitation du temps réels sur microcontrôleurs (90 pages))

Ressources et références

Les supports pédagogiques sont disponibles en ligne sous Campus

Contexte et enjeux de l'enseignement

Androïd est un système d'exploitation basé sur Linux, et destiné à l'origine au téléphone mobile (smartphone) et tablette. Il est le système d'exploitation le plus répandu sur les équipements mobiles, et à ce jour intègre de nouveau support comme les téléviseurs ou les montres connectées. Il est donc un système incontournable dans le domaine de l'informatique. Le développement d'application Androïd est basé sur le langage de programmation Java, avec des spécificités propres au développement en environnement contraint (peu d'espace de stockage, puissance de calcul limitée...), et une IHM spécifique. A l'issu de ce cours, les élèves seront capables de développer une application Androïd.

Prise en compte des dimensions socio-environnementales

Prérequis

Ce cours nécessite que les apprentis aient des connaissances concernant : la modélisation objet, la programmation Java, les notions de base du web

Modalités d'enseignement et d'évaluation

	Nb d'heures
Cours	
Cours intégré (cours + TD)	35
TD	
TP	
Projets	
Travail en autonomie encadré	
Contrôles et soutenances	1
Travail personnel	

Objectifs pédagogiques

(à la fin de cet enseignement, l'étudiant sera capable de ...)

- Appréhender la structure d'une application mobile Androïd
- savoir développer de bout en bout une application mobile sur Androïd
- connaître les bonnes pratiques

Activités

(CM, TD, TP, projet, sortie terrain, etc.)

- cours magistraux pour présenter les bases, le contexte
- séances mixtes, cours avec mise en pratique

Évaluations et retours faits aux élèves

(évaluations qui comptent pour la note ou qui permettent à l'étudiant de se situer, corrigés, feedback personnalisé...)

TP
QCM

Plan de cours

- Introduction à Android
- Configuration d'environnement : - Installation - Émulateur et débuggeur
- Structure d'un projet
- Ressources et Internationalisation
- Activités
- Tâches asynchrones
- Liste de Vues
- Persistance des informations
- Vues personnalisées
- Détection des gestes
- Capteurs
- Notifications
- Services en arrière-plan

Ressources et références

Les supports pédagogiques sont disponibles en ligne sous Campus.