



Mif10 – Projet transversal de master info

2023 – 2024

Sylvain Brandel

<https://forge.univ-lyon1.fr/m1if10/2023>

Encadrants

- Sylvain Brandel sylvain.brandel@univ-lyon1.fr gestion, GL
- Lionel Médini lionel.medini@univ-lyon1.fr Web + J2EE
- Emmanuel Coquery emmanuel.coquery@univ-lyon1.fr BD + VMs

Objectifs

- Pratiquer le développement dans le cadre d'un projet
 - Apprendre à travailler en groupe
 - Planifier et coordonner les efforts
 - Identifier les blocages et gérer les conflits
 - Documenter et communiquer sur son travail
 - S'appuyer sur les compétences de chacun
 - Mettre en pratique les techniques vues en M1 et en licence
 - Programmation avancée
 - Génie logiciel
 - Programmation web
 - Bases de données
 - Avoir un projet complet à mettre dans votre portfolio

Organisation

- 6 étudiants
 - Pas le choix de vos partenaires
- Durée
 - Aujourd'hui → 3 mai 2024
 - ~ 60 heures de travail individuel = ~ 350 heures de travail collectif par groupe
- Travail de groupe
 - Chacun doit pouvoir trouver son rôle et sa place
 - Voir au-delà des compétences techniques
 - Une note de groupe modulée en fonction de la contribution au sein du groupe, jusqu'à +/- 3 points

Sujet du projet

- Création d'une application Web permettant d'évaluer votre **bilan CO2 quotidien** en fonction de vos habitudes :
 - Trajet domicile – travail
 - Télétravail
 - Style d'alimentation
 - Loisirs / vacances ...
- Exemples : bilan carbone par jour et par personne
 - Durant une semaine de ski dans les Alpes au départ de Lyon en voiture à quatre personnes
 - Durant une semaine sur un paquebot de croisière au départ de Venise avec un trajet en avion depuis Lyon
 - Lors du travail avec trajets à vélo ou en voiture
 - Lors du télétravail ...
- Les cas d'usage spécifiques seront à spécifier au sein de votre groupe d'ici la fin de la 1^{ère} semaine. Vous pouvez proposer un sujet qui rentre dans ce cadre :
 - Site grand public géré par des experts
 - Site grand public en crowdsourcing
 - Site à destination de personnes souhaitant faire des simulations à l'échelle d'une région ou d'un pays ...
- Techniquement, votre application devra s'appuyer sur une base de données contenant *a minima* des informations sur les modes de calcul du bilan carbone dans une situation particulière et pouvant être mise à jour par certains utilisateurs (administrateurs), mais également toute autre information nécessaire à la réalisation des cas d'utilisation que vous aurez choisis
- L'interaction avec l'application se fera à travers une interface web

Planning global

[illegible]

- Permanences

- Mardi 9h45 – 11h15 : Emmanuel Coquery et Sylvain Brandel
- Mardi 11h30 – 13h : Lionel Médini et Sylvain Brandel
- Jeudi 14h – 15h30 7/03 et 28/03 : Sylvain Brandel
- Vendredi 14h – 15h30 : Emmanuel Coquery et Sylvain Brandel
- Vendredi 15h45 – 17h15 : Lionel Médini et Sylvain Brandel

Déroulement

- 6 semaines (3 quinzaines) de projet
 - Semaine 1 – 26/02 Mise en route ~ 5 heures de travail / personne
 - Semaine 2 – 04/03 Cadrage ~ 10 heures
 - Semaine 3 – 25/03 Stack et démarrage ~ 10 heures
 - Semaine 4 – 01/04 Proto 1 « MVP » ~ 15 heures
 - Semaine 5 – 22/04 Développement et tests ~ 15 heures
 - Semaine 6 – 29/04 Livraison et présentation ~ 10 heures
- Mêlées
 - Durée : 1 semaine
 - Attention aux enseignements des options

Travail à distance – outils

- On utilise tous les mêmes outils
 - GitLab avant tout pour communiquer, gérer l'information et les tâches
 - Rocket.chat / Mattermost : un pour l'UE, un par groupe
 - Pas de chats via Messenger, groupes WhatsApp, discord, etc.
 - E-mail pour les messages « officiels »
 - Visio si nécessaire, un lien fixe pour toute la durée de l'UE
 - On coupe les distractions :
 - Pas d'autres chats ouverts
 - Pas de notifications externes au projet
 - Téléphone en mode silencieux
 - Navigateur « propre »

Travail à distance - organisations

- On maintient un lien régulier
 - Des stand-ups à intervalle et créneau fixe
 - 3x par semaine :
 - Planning en début de semaine
 - Status update le mercredi
 - Update + rétro en fin de semaine
 - Structure du stand-up : tour de table et màj du board en parallèle
 - J'ai fait XXX
 - Je vais faire YYY
 - ZZZ me bloque ou me ralentit
 - Des pauses café de 15/20 minutes sur un créneau fixe et régulier
 - Parler de tout et de rien
 - Pas obligatoire mais encouragé

Un bilan par semaine sur le wiki

- Les bilans doivent
 - Lister présents, absents excusés et absents non excusés aux stand-ups de la semaine et aux réunions (uniquement pour les membres censés participer)
 - Capture de l'état du board (screenshot en début et fin de mêlée)
- Bilan / rétro à la fin de chaque mêlée
 - Résumé d'avancement
 - Bilan des issues pas finalisées, révision (on garde, on reformule, on jette)
 - Points de blocage technique et/ou humain
 - Etat de la qualité
 - Screenshots SonarQube
- Ces bilans vont sur le wiki

Rôles au sein des groupes

- Définissez des rôles et responsabilités, par exemple :
 - VM et intégration
 - Back-end
 - Front
 - Qualité et tests
 - Produit (UI/UX, cohérence des fonctionnalités)
 - Suivi de la progression (tracker)
 - Documentation / rapport
 - Cohérence de l'équipe
 - Respect de l'organisation
- Les rôles ne sont pas exclusifs, il faut une doublure en cas de blocage, maladie, abandon
- La responsabilité reste partagée, on intègre en continu

Gestion de la qualité

- Identifier les besoins utilisateur
 - Par des entretiens
 - Tester les réalisations en mettant de vraies personnes en situation
- Tests
 - Ecriture des tests (qui teste quoi)
 - Automatisation
 - Documents et rapports pour les tests manuels
- Intégration continue
- Audit de code
 - Outils (SonarQube)
 - Relecture de code croisée

Semaine 1 : Lancement

- Création des groupes
- Mise en place des outils de collaboration
- Choix du sujet
- Jalon 1 :
 - Groupe Gitlab créé
 - Projet Gitlab créé
 - Wiki
 - Page de garde créée qui organisera le contenu au fur et à mesure du projet
 - Page de l'équipe créée avec la liste des membres du projet (Nom, prénom, email, *téléphone*)
+ lien vers le salon privé de l'équipe
 - Enseignants de l'UE rajoutés comme rapporteurs au projet Gitlab
 - Précisez votre adresse Gitlab dans TOMUSS

Semaine 2 : Organisation et conception

- Organisation du travail de groupe
- Phase de conception orientée-usages
- Choix technologiques et mise en place des environnements de développements sur vos machines personnelles
- Jalon 2 :
 - Modalité de travail en groupe (page équipe du wiki mise à jour en fonction)
 - Rôles dans l'équipe,
 - Organisation interne définie
 - Objectifs et cible utilisateur définie (2 ou 3 paragraphes sur la page d'accueil du wiki)
 - Partie UX sur le wiki :
 - Cas d'utilisations principaux de l'application (ensemble de user stories)
 - Maquettes écrans de l'application (restez simple, il faudra les coder plus tard)
 - Des issues décrivant les user stories (si ce n'est pas sur le wiki)

Semaine 3 : Stack et démarrage

- Démarrage technique
- Mise en place de la VM
- Mise en place des scripts de CI/CD
- Mise en place de SonarQube
- Jalon 3 :
 - Test individuel (manuel) de chaque élément de l'architecture
 - Déploiement d'un "Hello world" permettant d'afficher une entrée (simple) de la BD dans l'interface web
 - Maj du wiki avec pointeurs techniques
 - Mémos techniques
 - How-to intégration continue pour tous les membres de l'équipe
 - How-to utilisation de git (branche, merge) pour être au clair dans l'équipe
 - Outils et frameworks/bibliothèques utilisées par le groupe
 - Choix justifié des solutions de persistance des données
 - 1e jet d'architecture (qui sera mis à jour au fur et à mesure)
 - 1 diagramme global + explications
 - Diagrammes UML : description di matériel et du logiciel, vues statiques et dynamiques

Semaine 4 : Prototype MVP

- Une version 0 de l'application est testable sur la VM
- Jalon 4 :
 - Chaîne d'outillage fonctionnelle (Intégration continue : construction, tests, qualité, déploiement)
 - UNE fonctionnalité de base est implémentée et intégrée
 - Exemple : création et affichage d'un événement
 - Màj du wiki

Semaine 5 : Développement et tests

- Développement et tests
- Jalon 5 :
 - Màj du wiki avec écrans des tickets traités et extrait SonarQube

Semaine 6 : Livraison et présentation

- Finalisation (pas d'ajout de fonctionnalités)
 - Refactoring au besoin
 - Correction de tests
 - Documentation
 - Livraison du code
- Démo sur VM
- Soutenances
- Jalon 6 :
 - Guide d'utilisation sur le wiki
 - MàJ de l'architecture
 - Documentation technique

Rendu

- Attention au jour férié ...
- Attendus
 - Rendu de code mardi 30/04 23h59
 - Wiki jeudi 2/05 12h00
 - Évaluation entre pairs vendredi 3/05 23h59
- Projet
 - Mettre les intervenants de l'UE comme reporter de votre projet
 - Créer une branche FINAL correspondant au code de la démo
 - README.md détaillant les dépendances, la procédure de build, et un lien vers une VM de démo
 - Un wiki structuré qui contient les rendus intermédiaires et les notes de réunions
 - Un issue board propre

Soutenances : 2/05 et 3/05 après-midis

- 25 minutes
 - 15 minutes de présentation et démo
 - 10 minutes de questions
 - La durée de présentation est stricte : RÉPÉTEZ
- Jury
 - Jeudi 2/05 : Sylvain Brandel et Emmanuel Coquery
 - Vendredi 3/05 : Sylvain Brandel et Lionel Médini
- Présentation
 - Montrez votre réalisation (sous forme de démo), discutez vos choix techniques et votre organisation de groupe
 - Présentation
 - Mode de fonctionnement collaboratif
 - Choix d'architecture, quelques diagrammes UML, patterns utilisés, etc.
 - Méthodes et outils de suivi et de déploiement du code
 - Bilan de votre rétrospective Agile
 - Quelles leçons tirez-vous de ce projet ?
 - Démonstration
 - Préparez un scénario de démonstration et déroulez-le
 - Mettez en avant les points forts de la réalisation aussi bien techniques qu'en terme d'usages
 - Discutez des compromis que vous avez dû faire, de vos choix finaux en termes de fonctionnalité, d'interface, de sécurité, etc.
 - Questions
 - Préparez-vous

Évaluation

- Le projet sera évalué en trois pans
 - Rendus intermédiaires et rendu final
 - Présentation et démo
 - Évaluation entre pairs
- Critères d'évaluation
 - L'évaluation par les pairs : +1 / 0 / -1 sur la note de groupe
 - Encadrants : +2 / +1 / 0 / -1 / -2 sur la note de groupe
 - Les notes au sein d'un groupe peuvent donc varier de -3 à +3