

Projet M2 DFS&DE : document de cadrage

Mail : stephane.gareddu@ac-corse.fr

*Mots-clés : Programmation, JavaScript, JSON, IHM, PHP, site web, bases de données, etc.
(Document partiellement basé sur le travail de l'équipe pédagogique de l'année précédente)*

Introduction

Le présent document est votre document de référence. N'hésitez pas à vous y référer si vous avez un doute ! Vous pouvez bien sûr me contacter si vous avez des questions.

Vous êtes non-alternants en M2, vous devez donc réaliser un projet, en tenant compte de tous les aspects : conception, implémentation, cahier des charges, gestion de projet, documentation technique, interface(s), stockage et gestion des données, aspects juridiques (vie privée & RGPD, autres...etc), dimensions éthiques, hébergement, fonctionnalités.

Deux cas se présentent cette année : de nouveaux projets sont créés, ou des projets de M1 sont poursuivis. Pour ce dernier cas, l'idée est bien évidemment de ne pas évaluer ce qui a déjà été fait, mais plutôt la quantité et la qualité de ce qui va l'être.

Après nos 12 premières heures de cours, nous avons déjà des équipes constituées et des projets définis. Tout le monde (vous, l'équipe enseignante, moi) est d'accord sur ces derniers : par conséquent, désormais, plus aucun changement ne devra être effectué. Il faut avancer !

Nom	Prénom	Filière	Projet
ABBOURI	HAFSA	DE	résuméRéunion
BESSAM	MOUNIA	DE	résuméRéunion
CATTAROSSI DARTIGUELONGUE	THOMAS	DFS	HealthyCore
EZ-ZAHER	MANAL	DFS	résuméRéunion
GHOULAM	MEHDI	DFS	Mindlet
MAHAM BOWBA	YOUSSOUF	DE	résuméRéunion
MARCHISELLI	ANTHONY	DFS	HealthyCore
MENGHI	ANTHONY	DFS	Mindlet
ROBLES	NICOLAS	DFS	HealthyCore
SECK	ABDOU SAMATH	DE	HealthyCore

Dans la suite de ce document, et sauf mention contraire, ce qui est écrit s'applique à tous les groupes sans exception.

Vous devrez effectuer une soutenance orale du projet et fournir un rapport, qui compteront pour la note finale. Mais bien d'autres points seront évalués, ils sont détaillés ici.

Contraintes

Tout ce qui suit est obligatoire. Vérifiez bien tout au long du projet que vous validez tous les points suivants !

1. Le projet possède :
 - a. un front-end (ergonomique, responsive design) et un back-end (POO, idéalement combinant une API et des micro-services), tous deux attribués aux DFS
 - b. une partie "data", qui sera attribuée aux DE
 - c. des communications entre le front et le back via un réseau
2. L'application sépare le stockage des données, leur présentation, leur traitement. Pour cela vous vous appuierez sur une architecture *n-tiers* (par exemple MVC), et expliquerez (le cas échéant) comment elle s'intègre dans une architecture hybride
3. Des sous-équipes sont constituées, et leurs rôles définis :
 - a. Equipe base de données/data
 - b. Equipe front end
 - c. Equipe back end
 - d. Equipe DevOps

Attention : cela n'interdit pas une certaine souplesse, et les rôles peuvent ponctuellement être interchangeables.
4. Mettre en place des outils de gestion de projet :
 - a. Dépôt de code & documentation en ligne
 - b. Outils de planification, répartition et de suivi des tâches
 - c. Outils de communication inter-équipe et avec l'équipe d'encadrement
5. Tenir compte des aspects de sécurité (stockage et chiffrement des données, interactions avec l'utilisateur via requêtes HTTP...), de protection des données (RGPD), et d'éthique (les mentionner dans le rapport !). Un petit threat model est apprécié !
6. Les données doivent être anonymisées quand cela est possible, et sinon pseudonymisées (les scripts le prouvant doivent être fournis)
7. Une orientation SEO pour les DFS est appréciée (si site web public)
8. Faire une évaluation des performances actuelles et des projections réalistes en cas de montée en charge (c'est-à-dire de succès de votre projet). Cela veut dire prévoir des solutions (même partielles) pour que l'appli s'adapte
9. Les fonctionnalités métier critiques doivent faire l'objet de tests unitaires qui seront détaillés dans le rapport (et bien entendu mis en place effectivement !). Vous pourrez faire un lien avec des cas d'utilisation (plus précisément, les *scenarii*...)
10. L'application doit contenir suffisamment de données, ayez recours au *dummy data* lorsque cela est nécessaire
11. Vous êtes libres de choisir :
 - a. les technologies : on préfèrera toutefois, quand cela est possible, des technologies

- robustes, sûres, et open-source
 - b. la solution d'hébergement
12. Le rapport contiendra un état de l'art: la techno que vous avez choisi d'utiliser, ses principales technologies concurrentes.
13. Le rapport contiendra un récapitulatif de tous les éléments précédents. Il contiendra obligatoirement en annexe :
- a. un cahier des charges (voir ci-après)
 - b. une documentation technique (ni sur-détaillée, ni sous-détaillée). Le code est commenté intelligemment
14. Il contiendra aussi les difficultés et bugs rencontrés, et mettra en valeur l'importance de la phase de conception (y compris les réflexions autour des tests unitaires) pour leur résolution, et vous y apporterez un regard critique
15. Les DE devront apporter leur expertise et leur contribution (et pas seulement en "appelant" une IA...) : il leur faudra collecter, stocker, gérer pertinemment des données, par exemple en mettant en place des algorithmes d'intelligence artificielle, des statistiques avancées, ou toute autre tâche que ferait difficilement un DFS ...
16. Vous mentionnerez explicitement toutes vos sources (liens, ouvrages), et indiquerez clairement les points pour lesquels vous avez sollicité une IA générative (réécriture de texte ou de code, idées...)

Timeline générale

À compter de maintenant, des points de situation seront désormais réalisés oralement avec tous les membres des groupes ($\approx 10\text{min}/\text{groupe}$) en début de chaque bloc de séances. Toujours en début de séance, je vous demande aussi, pour chaque groupe, un fichier contenant un mini journal de bord (5 lignes maximum par personne) résumant le travail accompli.

Vous devrez également me transmettre différents documents à différentes dates (voir ci-dessous). Par conséquent, l'évaluation sera principalement continue.

Les jalons ci-après sont obligatoires, mais vous devez aussi avancer sur d'autres points. Pour cela, je communiquerai individuellement à chaque groupe d'autres tâches spécifiques à accomplir, soit durant la séance, soit pour la prochaine échéance, soit pour une date donnée.

1. Pour le 03/10/2025 :
 - a. un résumé de (1 feuille A4 recto-verso maximum) expliquant d'où vous partez, où vous voulez arriver, quelles tâches sont à accomplir (détaillez-les). Cela va vous paraître fastidieux, mais vous qui êtes à fond dans votre projet, poser vos idées par écrit vous aidera à réfléchir autrement sur ce dernier. Cela clarifiera votre pensée et votre compréhension (et la mienne !). Dites qui accomplira ces tâches basiques (vous pouvez les regrouper en *sprints*) et selon quelle timeline (vous pouvez utiliser un Gantt). Cela sera utile pour le cahier des charges et pour l'évaluation de votre progression. Ne vous jetez pas sur le code, je veux voir de la conception !

- b. un début de documentation technique sous forme de diagramme des cas d'utilisation (pas de diagramme fantaisiste : utilisez Modelio qui est gratuit et se prête bien à cela). Ce diagramme s'intègrera parfaitement au cahier des charges
 - c. un *mockup* de votre IHM (s'il n'a pas déjà été fait) : présentez les différentes vues ou du moins celles qui sont le plus significatives selon vous
 - d. Si l'appli est déjà (même partiellement) développée, fournissez des screenshots
2. Pour le 10/10/2025 :
- a. un MCD (s'il n'a pas déjà été présenté en M1) qui illustre comment vous stockez vos données. Rappel : on peut tout à faire un MCD même si on travaille en *noSQL* ou avec une BDD vectorielle (moyennant quelques ajustements lors du passage au MPD)
 - b. son implémentation en MPD (fournissez des fichiers, et des extraits de code)
 - c. une présentation de l'architecture globale existante ou envisagée (vous pouvez vous appuyer sur des diagrammes UML de composants, et/ou de déploiement) ainsi qu'une première justification de vos choix techniques. Un début d'implémentation est à ce stade tout à fait apprécié : dans ce cas fournissez-moi un diagramme de classes, même partiel !
3. Pour le 07/11 : un prototype partiel qui contient obligatoirement des fonctionnalités métier
4. D'autres échéances sont à venir...
5. 12/2025-01/2026 (à définir) : rendu des rapports, soutenance

Cahier des charges

Vous allez devoir rédiger un cahier des charges selon le plan suivant, et l'ajouter à votre rapport final. Comme vous pourrez le remarquer, je vous demande déjà un bon nombre d'éléments qui serviront par la suite à ce cahier des charges.

Plan du cahier des charges fonctionnel :

1. Présenter le contexte du projet, préciser s'il existe des projets déjà réalisés se rapprochant de vos visées, évaluation de leur concurrence éventuelle
2. Définir le besoin avec précision (faites comme si vous aviez une demande d'un client) en ayant recours, à minima, aux diagrammes des cas d'utilisation ou assimilés
3. Identifier les résultats attendus en leur associant une planification (Par exemple version minimale v0 semaine 2, version acceptable v1 semaine X, version optimale v2, version rêvée v3 semaine T), et des membres de votre équipe
4. Cadrer les contraintes du projet (en citant par exemple celles qu'on vous a données)
5. Lister les ressources nécessaires (humaines/matérielles/infrastructure/logiciel/data)

Au niveau des contraintes techniques, votre projet doit donner lieu au développement d'un prototype fonctionnel avec :

- Une infrastructure (de préférence à base de conteneurs)
- Un backend en micro-services ou sous forme d'API

- Une ou plusieurs bases de données
- Un frontend, ergonomique et qui doit s'adapter au device de l'utilisateur ou une application native

Il est possible de proposer un (micro) service qui se base sur des méthodes d'IA.

Modalités d'évaluation et précisions

L'idée est d'évaluer non pas le seul produit fini mais bien une progression régulière (implication, planification, réalisation, constance du travail, collaboration au sein de l'équipe).

Bien sûr, parfois on avance plus ou moins vite, et on planche sur des bugs ou des comportements inattendus de l'applications. Mais un travail quasi nul durant les séances, suivi d'un projet merveilleux présenté à la fin, n'obtiendra pas une bonne note. Je veux pouvoir assister à l'évolution de votre projet, afin que je m'assure que ce dernier est le fruit d'un travail de groupe.

Cela ne m'intéresse pas de ne connaître la répartition des tâches et leur durée uniquement pour la soutenance et/ou le rapport : au contraire, là encore, je veux pouvoir mesurer l'évolution du projet. Il est clair qu'entre les durées prévues et les durées effectives il peut y avoir de gros écarts, et cela est normal. C'est d'autant plus intéressant. Mais c'est pour cela que le plus tôt possible les rôles et tâches de chacune et chacun doivent être définis, et que je dois y avoir accès. Cela ne se fait pas à postériori !

Le travail sera personnel et équitablement réparti entre les membres d'un même projet, en fonction de leur filière et de leurs compétences.

Concernant la documentation technique, il nous faut parler un langage commun. Je ne vais pas lire des centaines/milliers de lignes de code réparties dans des dizaines de fichiers. Je ne connais pas toutes les technos, par contre vous devrez être capables de m'expliquer ce que vous faites. J'ai donc besoin d'éléments synthétiques. Pour cela, il existe différents langages de modélisation (UML, mais pas que), utilisez-les ! Ne faites pas de schémas fantaisistes que seuls vous pouvez comprendre. À vous de vous poser la question de ce que le schéma doit montrer, et de choisir la manière la plus appropriée pour le représenter.

Durant nos séances, je mettrai deux casquettes :

- 1) Le client agaçant qui ne comprend rien mais qui veut tout et très vite
- 2) Le directeur technique/concepteur qui parfois réfléchira avec vous, parfois vous imposera des contraintes

(Le mot "casquette" est à prendre ici au figuré).

L'intelligence artificielle (LLM...) peut être utilisée, elle est une source au même titre que d'autres sources plus traditionnelles. Mais elle doit l'être en toute transparence, et en tout état de cause avec parcimonie. Ainsi, dans votre rapport, vous ferez apparaître clairement (couleur différente, pictogramme, encadré...) les éléments pour lesquels vous avez utilisé une IA. L'usage de l'IA doit être ensuite détaillé dans les annexes (quelle IA ? Quand ? Comment ? Pourquoi ?) et justifié.

Vous fournirez les logs (prompts envoyés, réponses des IA, et "discussions" éventuelles avec ces dernières).

Par exemple :

Où (fichier/section) :

Quel outil (nom/version/URL) :

But : assistance rédaction/code :

Prompts utilisés & réponses sauvegardées : (fichier)

Ce qui a été modifié : ligne(s) précise(s) / bloc

Justification : pourquoi acceptable/pertinent

Risques évalués : biais, droits, confidentialité

Il en va de même pour toutes les autres sources : sites ou ouvrages consultés, provenance du code, documentations consultées...

Les liens avec vos enseignements de Master et plus généralement votre cursus seront appréciés, c'est-à-dire les liens avec les concepts vus en cours. Implémentation d'algorithmes, utilisation des techniques de conception, méthodologies diverses...etc.

Je veux pouvoir dans l'idéal tester votre application depuis chez moi, quand je le souhaite, et ainsi regarder de temps en temps où vous en êtes. Vous devez me fournir une petite documentation concernant la reproductibilité de votre environnement (comment je peux l'installer).

À titre indicatif, voici une grille de notation dont la plupart des critères peuvent être quantifiés :

	Très bien	bien	Moyen	Peu	Pas du tout
Avant chaque séance					
A effectué les tâches données					
A respecté les échéances données (documents envoyés en temps et en heure)					
A respecté la répartition des tâches et des rôles					
Qualité des documents et information fournis (vidéos, code...)					
Durant chaque séance					
A présenté ses avancées par rapport à la fois précédente					
Chacun peut expliquer son code, ses documents, ses environnements de travail					
Tient compte des remarques du client (moi)					
Tient compte des remarques du concepteur (moi)					
Qualité de la progression (production de code, implémentations...)					
Qualité des interactions (pertinence dans les entretiens)					
Le rapport contient :					
Un cahier des charges					
Des annexes avec l'état de l'art et toutes les sources (IA comprise)					
Une justification des choix techniques					
Une présentation des éléments les plus significatifs de l'application					
Une séparation claire entre client, serveur, données (n-tiers)					
Des éléments concernant la sécurité, l'éthique, la loi					
Une documentation technique précise et pertinente, des diagrammes standardisés					
Des éléments de gestion de projet (GANTT)					
Une mise en valeur du travail des DFS et des DE					
Une phase d'analyse et conception détaillée					
Application					
Respect du cahier des charges					
Ergonomie					
Fonctionnalités présentes vs. Fonctionnalités prévues					
Bugs éventuels					
Authentification sécurisée					
Qualité du code (clarté, commentaires, tests unitaires...)					
Interface d'administration					