

# Le(s) cahier(s) des charges pour le projet tuteuré.

## Recommandations générales

Tout d'abord des recommandations pour tout document technique que vous devez préparer. Pour simplifier le partage il est bon de pouvoir le distribuer au format pdf, quitte à fournir en plus les sources (au format markdown, latex, ods, docx, ...).

Vous pouvez en général le préparer avec le logiciel de votre choix (je conseille par exemple overleaf pour éditer à plusieurs du LaTeX en ligne ou l'éditeur intégré de gogs, ou encore docs). Il convient de ne pas perdre trop de temps sur le style en cherchant des fontes compliquées ou un design très léché (ce n'est pas un poster). Ce qui compte avant tout, c'est la lisibilité. Je recommande une taille de fonte de 12 pt. Le texte est justifié à gauche et à droite. Les figures doivent être lisibles et donc d'une taille adaptée par rapport aux détails présentés. En particulier, il faut une bonne résolution des images. Les figures ont une légende.

Concernant le contenu, il faut prendre le soin de vous relire et, quitte à simplifier votre style pour ne pas quitter votre zone de confort, privilégier avant tout la clarté de votre propos. Il faut utiliser le moins de jargon possible. Parfois on ne peut pas faire autrement que d'introduire un mot technique. Dans ce cas, il faut bien expliquer de quoi il s'agit. Il faut éviter les abréviations, surtout si le document n'est pas très long. Si vous voulez absolument utiliser une abréviation, il faut l'expliquer la première fois qu'elle est utilisée en note de bas de page ou entre parenthèses. Si le document fait plus que quelques pages il faut regrouper ces abréviations et leurs définitions à la fin du document.

De manière générale il faut mettre en oeuvre ce que vous avez vu dans vos cours d'expression et communication.

## Contrainte spécifique au PT

Pour faciliter le travail du correcteur, il faudra que tous vos documents (cahiers des charges, rapports etc) **soient disponibles sur le dépôt git de votre projet en markdown en plus du pdf.**

En plus des deux cahiers des charges que nous allons décrire plus loin, les autres documents à fournir (**en markdown et en pdf**) pour le PT sont :

- un **readme** expliquant comment installer le logiciel
- une **documentation pour l'utilisateur** (ce n'est pas la même chose que la documentation technique pour les développeurs).
- le **rapport final** (voir recommandations du responsable des projets).

## Recommandations spécifiques au cahier des charges

On va faire la distinction entre deux cahiers des charges.

Le premier est le **cahier des charges fonctionnelles**. Il doit être compréhensible pour un non informaticien.

Le second est le **cahier des charges techniques et méthodologiques**. Il doit être compris par les membres de l'équipe et par votre tuteur.

La première page d'un cahier des charges comporte le nom du projet, la date de création du cahier des charges, les responsables référents du projet et les auteurs ayant contribué à sa rédaction et l'adresse du dépôt git.

### Cahier des charges fonctionnelles

Je suggère le plan suivant qu'il faudra adapter selon votre projet pour le cahier des charges fonctionnelles.

Présentation du contexte.

Indiquer qui est le client. Esquisser les besoins, les contraintes, l'existant.

Étude détaillé des objectifs (analyse des besoins).

On explique de manière assez précise les fonctionnalités attendues, les utilisateurs etc. Typiquement on peut illustrer par un diagramme de cas d'usage (DCU, voir cours de conception). On peut aussi fournir des synopsis, c'est-à-dire des petits scénarios concrets (*user stories* en anglais) illustrant certains cas d'usage (voir cours de Conception). Si le projet s'appuie sur une interface (web ou autre) qui est assez claire dans l'esprit du client, on peut aussi faire un wireflow (voir cours IHM).

Calendrier et priorisation des objectifs.

En accord avec le client, on fixe des jalons (dates et rdv fixés en avance) avec des objectifs assez précis du livrable pour chaque jalon.

Par exemple pour le projet tuteuré annuel, sur un projet assez classique avec une interface graphique, on pourrait s'engager comme ci-dessous :

- jalon 0 : cahier des charges fin septembre (à faire signer par le client, avec parfois des corrections)
- jalon 1 : maquette d'une interface graphique fin octobre.
- jalon 2 : prototype fonctionnel avec seulement une partie des objectifs fin décembre.
- jalon 3 : version de test fin janvier.
- jalon 4 : version finale fin février.

Il faut fixer des priorités sur les objectifs avec le client. On peut pour se faire utiliser la notation [MoSCoW](#) : M pour must (vital), S pour should (essentiel), C pour could (confort), W pour won't (probablement jamais livré mais envisageable).

On détaille d'autant plus les objectifs en sous-objectifs ayant eux-même une priorité que ceux ci sont vitaux ou essentiels. Les objectifs moins prioritaire pourront être revus en détails ultérieurement avec le client.

Cahier des charges techniques et méthodologiques.

Celui-ci peut-être plus court que le cahier des charges fonctionnelles et pourra être complété en plusieurs fois.

On détaille les technologies envisagées en justifiant les choix.

Il devra contenir une bibliographie (articles, livres, sites webs, videos, ...).

On esquisse au moins en partie une proposition de design.

On peut par exemple réaliser un diagramme de classe au moins pour le modèle ou pour une base de donnée, quitte à le compléter dans une version ultérieure.

Concernant la méthodologie, on détaille les méthodes de travail envisagées (agile? pair programming? test driven?) et outils qu'on souhaite mettre en place (framaboard/trello, convention de codage, organisation du repo, outil ou script de test, outil ou script de déploiement, etc).

Pour le PT, il faudra **obligatoirement utiliser un repo git** (à renseigner) et mettre en oeuvre une politique de test ambitieuse (tests unitaires reproductibles a minima) et documenter convenablement le code (javadoc, doxygen). Il faudra bien réfléchir au déploiement du logiciel (make) et à la documentation utilisateur (readme, manuel utilisateur, aide intégrée).

Il faudra aussi expliquer comment l'équipe fonctionnera et quel rôle privilégié on envisage pour chaque membre de l'équipe, sachant qu'une spécialisation trop poussée et dangereuse car elle peut rendre des membres de l'équipe nécessaire.

Ce cahier des charges contient aussi un diagramme de Gantt prévisionnel qui reprend les jalons du cahier des charges, et décrit au moins en partie le calendrier prévisionnel pour les tâches correspondant aux objectifs fixés par le client.

Ce diagramme de Gantt sera mis à jour pour tenir compte du travail réel.

En particulier le rapport final devra contenir au moins deux diagrammes. Le diagramme prévisionnel et le réel.

## Remarques

En pratique, *il n'y a qu'un seul cahier des charges sur les projets dans l'industrie.* Nous dévions un peu de ces pratiques pour des raisons pédagogiques.

Ce document décrit ce que veut le client, et précise comment va s'organiser la relation entre le client et le fournisseur (éléments de planning, comment le fournisseur sera évalué, etc.). Pour les projets de développement, le cahier des charges référence souvent au moins les spécifications fonctionnelles, qui détaillent précisément ce que veut le client (diagrammes de use case, priorisation, etc.). Souvent les entreprises ont un référentiel d'exigences techniques qu'elles recensent dans un autre document, les spécifications techniques qui décrivent comment on veut que ce soit fait, pour que le produit s'intègre facilement dans le SI. Ensuite, soit on est allé encore plus loin, et on fournit des spécifications techniques détaillées (diagramme de classe etc.) qui détaillent comment on veut que ce soit fait, et le fournisseur n'a plus qu'à coder, soit on laisse le fournisseur décider de l'implémentation (il n'a que les spécifications techniques générales à respecter), et il fournit alors en général le code et les spécifications techniques détaillées.

En conclusion, soit le cahier des charges référence les spécifications qui sont un document à part, soit il les intègre (mais du coup c'est assez long) mais dans des paragraphes bien distincts.