

Guide de rédaction du rapport de PFE

Introduction générale

Chapitre 1 : Présentation du cadre de projet

1. Introduction
2. Présentation de la [société d'accueil]
3. Etude de l'existant
 - 3.1. Description de l'existant
 - 3.2. Critique de l'existant
 - 3.3. Solution proposée
4. Choix de modèle de développement
5. Planning prévisionnel
6. Conclusion

Chapitre 2 : Spécification des besoins

1. Introduction
2. Spécification des besoins fonctionnels
3. Spécification des besoins non fonctionnels
4. Présentation des cas d'utilisation
 - 4.1. Présentation des acteurs
 - 4.2. Diagramme des cas d'utilisation global
 - 4.3. Description des cas d'utilisation

Chapitre 3 : Conception du système

1. Introduction
2. Modélisation dynamique
 - 2.1. Diagrammes de séquences
 - 2.2. Diagrammes de collaboration
 - 2.3. Diagrammes d'états
 - 2.4. Diagrammes d'état-transition
 - 2.5. Diagrammes d'activité
3. Modélisation statique
 - 3.1. Diagramme de classes
 - 3.2. Modèle relationnel
 - 3.3. Dictionnaire de données
4. Architecture logicielle de l'application
5. Conclusion

Chapitre 4 : Réalisation du système

6. Introduction
7. Environnement de développement
 - 7.1. Environnement matériel
 - 7.2. Environnement logiciel
8. Principales interfaces graphiques
9. Conclusion

Conclusion générale

Bibliographie et Webographie

Guide de rédaction du rapport de PFE

Introduction générale

[Tous le long du rapport/présentation, l'étudiant doit utiliser le pronom « nous » à la place de « je »]

L'introduction générale comporte, généralement, deux parties.

1. Dans la première partie, l'étudiant présentera son sujet à travers des informations précises et posera par la suite la problématique à résoudre avec clarté et sans évocation de résultats. [Il ne faut pas parachuter des introductions « passe partout »]
2. Dans la deuxième partie, l'étudiant présentera le plan de son rapport en évoquant, brièvement, le contenu de chaque chapitre.

Chapitre 1 : Présentation du cadre de projet Objectifs du chapitre

Ce chapitre comprend, généralement, trois parties ; la présentation de la société où s'est déroulé le stage, une étude de l'existant sur les modalités de travail actuelles, la critique de l'existant et les solutions envisagées par l'étudiant.

1. Présentation de la [société d'accueil]

Cette partie comprend une brève description de la société d'accueil : son secteur d'activité, un bref historique, les différents départements ou services et enfin son organigramme. Il faudrait, surtout, insister sur le service ou département auquel l'étudiant a passé son stage et s'accentuer en outre sur les outils informatiques utilisés.

Attention : La présentation de la société n'est pas une publicité pour celle-ci ; il ne s'agit pas de vanter ses mérites ou les services qu'elle offre.

2. Etude de l'existant

Cette partie comprend, généralement, trois parties.

2.1. Description de l'existant

Il est question d'expliquer comment le travail s'effectue, actuellement, au sein de la société (en rapport avec l'application qui va être développée par l'étudiant).

2.2. Critique de l'existant

Cette partie permet d'évoquer les insuffisances de la solution actuelle. En effet, les points faibles de la solution actuelle doivent figurer dans la section suivante et aussi dans l'application qui sera développée par l'étudiant.

2.3. Solution proposée

Deux cas se présentent : soit il y a une application existante qui présente certaines lacunes et, donc, la proposition consiste à apporter des améliorations, soit tout est géré manuellement et il faudrait, donc, informatiser le processus de travail actuel. Dans les deux situations, il faudrait en quelques lignes présenter la ou les propositions possibles (en indiquant ses avantages et ses inconvénients) et justifier le choix de la solution adoptée.

[L'étudiant insistera ici sur l'originalité de son sujet par rapport aux travaux existants]

3. Choix de modèle de développement

Ici, l'étudiant doit donner la démarche qu'il va adopter pour développer son application. En outre, le choix de modèle de cycle de vie de développement d'un logiciel doit être justifié.

4. Planning prévisionnel

Ici l'étudiant doit mettre le plan de son travail pendant la période de son stage. Planning prévisionnel devra être représenté comme suit :

Tableau 1 : Planning prévisionnel

Etape	Novembre				Décembre				Janvier		
	1	2	3	4	1	2	3	4	1	2	3
Etude préalable											
Conception											
Réalisation											
Test et Validation											

Chapitre 2 : Spécification des besoins Objectifs du chapitre

Ce chapitre comprend, généralement, deux parties : les besoins fonctionnels et les besoins non fonctionnels.

1. Introduction

Lors de l'analyse des besoins, l'étudiant doit faire la distinction entre besoins fonctionnels et besoins non-fonctionnels. Les besoins fonctionnels répondent aux points précis du cahier des charges, et sont donc requis par la société. Ils ne sont pas négociables en général, c'est le "besoin primaire" de la société.

Les besoins non-fonctionnels sont des besoins/contraintes liés à l'implémentation et à l'interopérabilité générale. Ils peuvent être fixés par la société, ou par l'étudiant.

2. Spécification des besoins fonctionnels

Ce sont les besoins indispensables auxquels doit répondre l'application.

Pour plus de clarification, il est recommandé de présenter les besoins en indiquant les besoins globaux puis les détailler.

Pour cela, il est possible d'utiliser les puces ou les numérotations comme suit :

2.1. Besoin fonctionnel [global 1]

Exemple d'un besoin global : « Gestion des étudiants »

2.1.1. Sous-besoin 1

Exemple d'un sous-besoin relatif à la « Gestion des étudiants » on peut mettre : « Ajouter un étudiant »

2.1.2. Sous-besoin 2

3. Spécification des besoins non fonctionnels

Ce sont les besoins qui permettraient d'améliorer la qualité des services de l'application comme la convivialité et l'ergonomie des interfaces, l'amélioration du temps de réponse, ...

Il est, également, possible de les présenter sous forme de puces.

Remarque importante : Les étudiants doivent adopter le langage de modélisation UML, le chapitre de spécification des besoins devrait comprendre les diagrammes de cas d'utilisation dont nous détaillons dans la suite la structuration.

4. Présentation des cas d'utilisation

4.1.1. Présentation des acteurs

Dans ce paragraphe, l'étudiant présentera en bref les différents acteurs de l'application.

4.1.2. Diagramme des cas d'utilisation global

A ce niveau, l'étudiant devra mettre le diagramme de cas d'utilisation global en le dessinant avec le logiciel approprié pour la modélisation conceptuelle tel que : Visual Paradigm, PowerAMC, StarUML, etc.

4.1.3. Description des cas d'utilisation

A ce niveau, les cas d'utilisation peuvent être présentés par deux façons que ce soit par acteur, soit par fonctionnalité. Généralement, si les fonctions des acteurs sont complètement indépendantes, c'est la première solution qui est adoptée. Si en revanche, une fonctionnalité du système fait intervenir plusieurs acteurs, c'est la deuxième possibilité qui est adoptée.

Les cas d'utilisation présentant certaines ambiguïtés doivent être complétés par une description textuelle en présentant les points suivants :

Tableau 2 : Description du cas d'utilisation « xxx » pour l'acteur x

Cas n°	Numéro du cas d'utilisation
Acteur(s) :	Nom(s) de(s) acteur(s)
Objectif :	C'est le but du cas d'utilisation.
Pré-condition(s) :	Condition(s) devant être remplie(s) pour exécuter le cas d'utilisation.
Post-condition(s) :	Condition(s) nécessaire(s) pour que le cas d'utilisation soit considéré comme achevé.
Scénario nominal :	C'est le scénario indiquant les étapes pour réaliser le cas d'utilisation (il ne comprend pas d'alternatives) : il peut être, également, remplacé par un diagramme de séquence.
Scénario alternatif :	C'est le scénario indiquant les éventuelles étapes différentes liées aux choix de l'utilisateur, par exemple.

Chapitre 3 : Conception du système Objectifs du chapitre

Ce chapitre a pour objectif de présenter la solution conceptuelle proposée par l'étudiant. En d'autres termes, ce chapitre devrait répondre à la question COMMENT FAIRE.

1. Modélisation dynamique

1.1. Diagrammes de séquences

Un diagramme de séquence est un diagramme d'interaction dont le but est de décrire comment les objets collaborent au cours du temps et quelles responsabilités ils assument. Il décrit un scénario d'un cas d'utilisation

Exemple d'un diagramme de séquences d'authentification :

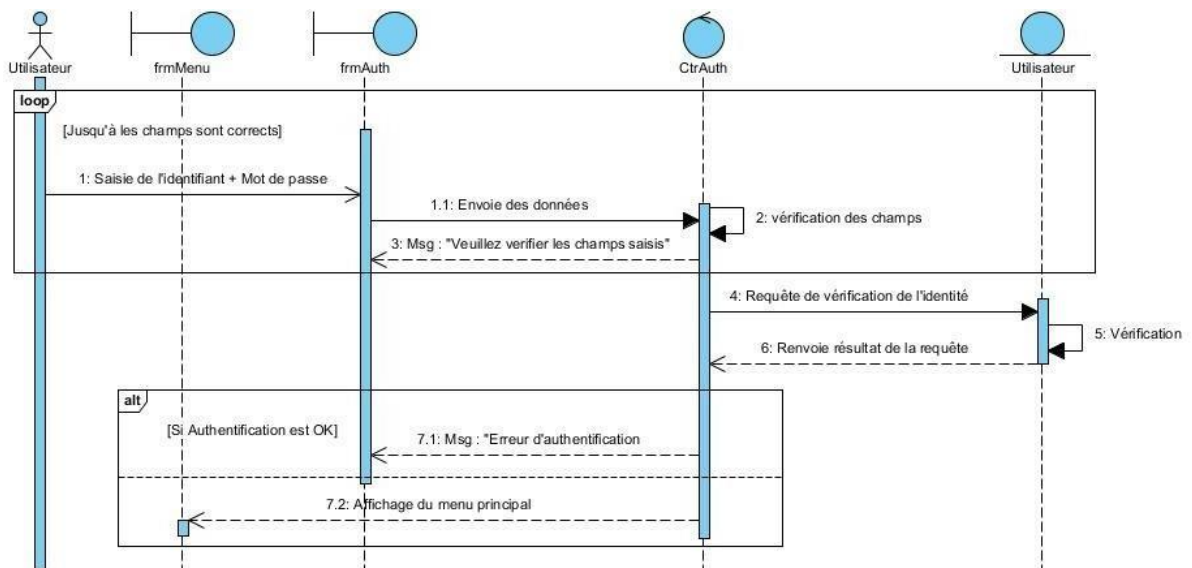


Figure n : Diagramme de séquences "Authentification"

Attention : l'étudiant doit mettre un seul diagramme de séquences parmi ceux qui se ressemblent d'un point de vu scénario.

Remarque : L'étudiant mettre au maximum 5 diagrammes de séquences.

2. Modélisation statique

2.1. Diagramme de classes

Ici sera dressé le diagramme de classes de la future base de données.

Attention : les classes doivent comporter les attributs (nom + type de données) et les méthodes.

2.2. Modèle relationnel

Ici sera dressé le modèle relationnel de la base de données en expliquant préalablement les règles de passage.

2.3. Dictionnaire de données

Les colonnes récupérées depuis le model relationnel doivent être représentées comme suit :

Tableau 3 : Dictionnaire de données

Nom de la colonne	Type de données	Taille	Obligatoire (Oui/Non)	Valeur par défaut	Clé		Nom de la Table
					Primaire	Etrangère	

Architecture logicielle de l'application

Ici sera dressé le diagramme de composant qui donnera un aperçu sur l'architecture logiciel de l'application qui a été développée.

Chapitre 4 : Réalisation du système

Objectifs du chapitre

Ce chapitre a pour objectif de présenter la solution logicielle et l'environnement de développement qui sont utilisés afin d'aboutir à développer l'application.

1. Introduction

Dans ce chapitre, l'étudiant présentera l'environnement matériel et logiciel adopté pour avoir un produit fini et présentera aussi les principales interfaces graphiques. En effet, ce chapitre sera composé par deux parties. La première partie détaillera l'environnement de développement et la seconde partie concernera la mise en œuvre de la solution proposée.

2. Environnement de développement

2.1. Environnement matériel

C'est l'environnement sous lequel l'étudiant a développé son application : les caractéristiques de l'ordinateur telles que la fréquence du processeur, la taille de la mémoire centrale ou s'il s'agit d'une application réseau, les routeurs ou hubs, serveurs, ...

2.2. Environnement logiciel

Ce sont les outils logiciels utilisés pour le développement de l'application ou de la base de données, la modélisation des différents diagrammes de conception, ...

2.3. Principales interfaces graphiques

Au niveau de cette rubrique, il faut placer les principales interfaces graphiques développées qui devraient être toutes commentées par un paragraphe de 2 à 3 lignes expliquant son contenu.

A noter qu'il ne faut pas placer toutes les interfaces de l'application, mais uniquement les plus importantes et celles qui seraient différentes. Les autres interfaces sont placées en annexes.

Conclusion générale

La conclusion du rapport doit comprendre, impérativement, un rappel de l'objectif du stage de perfectionnement et une récapitulation du travail fait en présentant les résultats (en d'autres termes, les réponses aux problèmes posés au début).

Il est, également, recommandé de porter un œil critique sur le travail fait en soulevant certaines insuffisances ou améliorations possibles.