

GitJournal

Table des matières

1	Analyse préliminaire	4
1.1	Introduction	4
1.2	Objectifs.....	4
1.3	Gestion de projet	5
1.4	Planification initiale	5
2	Analyse / Conception.....	5
2.1	Contexte produit	5
2.2	Contextes techniques	6
2.2.1	Opérationnel	6
2.2.2	Validation	6
2.2.3	Développement.....	6
2.3	Concept	6
2.4	Analyse fonctionnelle.....	7
2.5	Stratégie de test.....	7
2.6	Risques techniques	7
2.7	Planification	7
3	Réalisation.....	8
3.1	Points de design spécifiques	8
3.1.1	8
3.1.2	8
3.1.3	9
3.2	Déroulement	9
3.2.1	Sprints	9
3.2.2	Stories	9
3.3	Description des tests effectués	9
3.4	Bilan.....	9
3.4.1	Erreurs restantes	9
3.4.2	Stories	9
3.4.3	Dette technique.....	9
3.5	Recours à l'intelligence artificielle	9
3.6	Liste des documents fournis	10
4	Conclusions	10
5	Annexes.....	11
5.1	Résumé du rapport du TPI / version succincte de la documentation	11
5.2	Sources – Bibliographie.....	11
5.3	Journal de travail	11
5.4	Manuel d'Installation	11
5.5	Manuel d'Utilisation.....	11
5.6	Archives du projet	11

NOTE L'INTENTION DES UTILISATEURS DE CE CANEVAS :

Toutes les parties en italiques sont là pour aider à comprendre ce qu'il faut mettre dans cette partie du document. Elles n'ont donc aucune raison d'être dans le document final.

De plus, en fonction du type de projet, il est tout à fait possible que certains chapitres ou paragraphes n'aient aucun sens. Dans ce cas il est recommandé de les retirer du document pour éviter de l'alourdir inutilement.

1 Analyse préliminaire

1.1 Introduction

Ce chapitre décrit brièvement le projet, le cadre dans lequel il est réalisé, les raisons de ce choix et ce qu'il peut apporter à l'élève ou à l'école. Il n'est pas nécessaire de rentrer dans les détails (ceux-ci seront abordés plus loin) mais cela doit être aussi clair et complet que possible (idées de solutions). Ce chapitre contient également l'inventaire et la description des travaux qui auraient déjà été effectués pour ce projet.

Ces éléments peuvent être repris des spécifications de départ.

1.2 Objectifs

Ce chapitre énumère les objectifs du projet. L'atteinte ou non de ceux-ci devra pouvoir être contrôlée à la fin du projet. Les objectifs pourront éventuellement être revus après l'analyse.

Ces éléments peuvent être repris des spécifications de départ.

Séparer les objectifs en deux catégories : les objectifs de formation et les objectifs du produit.

1.3 Gestion de projet

Décrivez comment vous allez gérer votre projet. Quels sont les outils, les pratiques, les personnes impliquées, les rôles,... ?

1.4 Planification initiale

No	Objectifs	Durée	Dates	Date Sprint Review	
1	Donner son Token à l'app	~2H	~46H30	05.05 16.05	vendredi 18.05.2025 9H00
	Changer de Repository	~10H30			
	Afficher mes commits sous forme de JDT	~12H			
	Reprendre mon JDT sur un autre poste	~10H			
	Documentation	~10H			
2	Gérer les entrées	~10H50	~36H50	05.05 16.05	vendredi 23.05.2025 9H00
	Exporter en PDF	~18H			
	Documentation	~8H			

	28.04 - 02.05 Après Matin Midi	05.05 - 09.05 Après Matin Midi	12.05 - 16.05 Après Matin Midi	19.05 - 23.05 Après Matin Midi	26.05 - 30.05 Après Matin Midi	02.06 - 06.06 Après Matin Midi
Lundi	-	05:35	05:35	05:35	03:10 Examens	03:10 -
Mardi	-	-	-	-	-	-
Mercredi	-	07:15	07:15	07:15	07:15	-
Jeudi	-	- 03:10	- 03:10	- 03:10	Férié	-
Vendredi	07:15	07:15	07:15	07:15	Férié	-

Le total d'heures dans ma planification n'atteint pas les 90H prévue pour ce projet. Dans ces heures, je n'ai pas pris en compte le vendredi 2 mai 2025. C'est le jour où j'ai reçu mon CDC et j'ai pris la décision de ne pas l'inclure dans mon premier sprint. Ce vendredi a duré 7H15.

2 Analyse / Conception

2.1 Contexte produit

Placer le produit dans son contexte d'utilisation. Par exemple :

- **Site Web Internet**
- **Application, Web, intranet**
- **Application mobile**
- **Infrastructure dupliquer des sites multiples**

Présenter les utilisateurs du système ainsi que les rôles qu'ils peuvent avoir

Présenter les données/information que le système prend en charge

Au minimum :

- *Un ou plusieurs schémas de contexte montrant le produit dans son environnement d'utilisation, ainsi que ses utilisateurs. Ce type schéma doit être accompagné d'explications textuelles*

2.2 Contextes techniques

2.2.1 Opérationnel

Placer le produit dans son contexte d'opérations.

Quelles sont les machines impliquées ?

Combien sont-elles ?

Comment communiquent-elles ?

Quels sont les services et/ou applications tierces utilisées (p.ex : base de données) ?

Où se situe votre travail ?

Justifier les choix

Ce chapitre doit impérativement contenir au moins une représentation graphique

Au minimum :

- *Un ou plusieurs schémas de contexte montrant le produit dans son environnement technique. Ce type schéma doit être accompagné d'explications textuelles*

2.2.2 Validation

Faire la même description que ci-dessus, mais pour l'environnement dans lequel votre produit doit être validé avant d'être déclaré publiable.

Expliquer comment le produit sera déployé dans le contexte de production une fois validé.

Justifier les choix

2.2.3 Développement

Décrire l'environnement dans lequel le projet va être réalisé.

Ce chapitre doit permettre à tout nouvel arrivant dans l'équipe de projet d'installer son environnement de travail

Expliquer comment le produit sera déployé dans le contexte de validation une fois que vous l'estimez prêt.

Justifier les choix

2.3 Concept

Le concept complet avec toutes ses annexes :

Un ou plusieurs schéma conceptuel, accompagné d'explications textuelles, de type :

- *Modèle conceptuel des données. Expliquer chaque entité/attribut possédant une particularité.*
- *Schémas d'architecture montrant la structure interne du produit*
- *Diagramme de flux*
- *Diagramme d'état.*

Générez les images à partir des applications utilisées (draw.io, visio, workbench, figma, ...). N'ayez recours à la capture d'écran que s'il n'y a pas d'autre moyen de faire.

Par exemple :

- Multimédia : carte de site, ~~maquettes papier~~, story board préliminaire, ...
- Bases de données : interfaces graphiques, modèle conceptuel.
- ~~Programmation : interfaces graphiques, maquettes, analyse fonctionnelle...~~ **Ceci est dans l'analyse fonctionnelle ci-dessous**
- ...

2.4 Analyse fonctionnelle

Décrivez de manière précise en vous appuyant sur des éléments graphiques (maquettes, schémas) la manière dont vous envisagez que votre utilisateur va travailler avec votre produit.

Possibilité (si IceScrum est utilisé) : Reprendre le contenu des User Stories d'IceScrum : Story + tests d'acceptance (avec IceTools) + maquettes

2.5 Stratégie de test

En faisant référence au contexte de validation précédemment décrit,

Décrire la stratégie globale de test :

- types de des tests et ordre dans lequel ils seront effectués.
- ~~les moyens à mettre en œuvre.~~
- couverture des tests (tests exhaustifs ou non, si non, pourquoi ?).
- données de test à prévoir (données réelles ?) **et comment elles seront mises en place.**
- les testeurs extérieurs éventuels.

2.6 Risques techniques

- risques techniques (complexité, matériel critique, disponibilité de systèmes externes, manque de compétences...).

Décrire aussi quelles solutions ont été appliquées pour réduire les risques (priorités, formation, actions, ...).

Attention : le manque de temps n'est pas un risque technique

2.7 Planification

Révision de la planification initiale du projet :

- ~~planning indiquant les dates de début et de fin du projet ainsi que le découpage connu des diverses phases.~~
- ~~partage des tâches en cas de travail à plusieurs.~~

~~Il s'agit en principe de la planification **définitive du projet**. Elle peut être ensuite affinée (découpage des tâches). Si les délais doivent être ensuite modifiés, le responsable de projet doit être avisé, et les raisons doivent être expliquées dans l'historique.~~

Cette section n'est présente que si la planification initiale a dû être revue suite à l'analyse

3 Réalisation

3.1 Points de design spécifiques

Ce chapitre est constitué de plusieurs sous-chapitre.

Chaque sous-chapitre explique un point de design technique particulier, quelque chose que vous avez dû inventer pour répondre au besoin et qui ne peut pas s'expliquer par de simples commentaires dans le code.

Il s'agit d'explications techniques sur le fonctionnement du système. Les explications sont appuyées par des diagrammes, ou de très brefs éléments de code.

NE PAS mettre ici des pratiques usuelles que tout professionnel de la branche connaît déjà. Par exemple, n'EXPLIQUEZ PAS ICI CE QU'EST LE PATTERN MVC.

Exemple (simplifié à l'extrême) : Protection contre des formulaires mal intentionnés ou modifiés

- **Au moment de générer le formulaire, le script php :**
 - **Concatène les noms de tous les champs contenus dans le formulaire**
 - **Calcule un hash SHA256 de la chaîne obtenue**
 - **Ajoute un input nommé « CSRF » de type hidden dans le form**
- **A la réception du POST du formulaire**
 - **Concatène les noms des indices de \$_POST**
 - **Calcule un hash SHA256 de la chaîne obtenue**
 - **Vérifie que la valeur du champ CSRF correspond**

3.1.1 ...

3.1.2 ...

3.1.3 ...

3.2 Déroulement

3.2.1 Sprints

Pour chaque sprint :

- 1. Résumer le déroulement du sprint, citer le résultat (objectif) de sa revue**
- 2. Donner la synthèse de la Sprint retrospective**
- 3. Donner le planning du sprint suivant**

3.2.2 Stories

Résumer comment s'est passé la réalisation de chaque story, ses difficultés, les alternatives envisagées mais rejetées, ses surprises, ...

3.3 Description des tests effectués

~~Pour chaque partie testée de votre projet, il faut décrire:~~

- ~~• les conditions exactes de chaque test~~
- ~~• les preuves de test (papier ou fichier)~~
- ~~• tests sans preuve: fournir au moins une description~~

Faire la synthèse de tous les tests d'acceptance effectués tout au long du projet

3.4 Bilan

3.4.1 Erreurs restantes

S'il reste encore des erreurs :

- Description détaillée
- Conséquences sur l'utilisation du produit
- Actions envisagées ou possibles

3.4.2 Stories

Ce qu'on pensait faire vs ce qu'on a fait

3.4.3 Dette technique

Reporter la dette technique connue. S'appuyer sur la pratique des // TODO

3.5 Recours à l'intelligence artificielle

**Comment avez-vous utilisé l'IA dans votre projet.
Si vous ne l'avez pas utilisée, pourquoi ?**

Ce chapitre doit contenir au minimum 200 mots

3.6 Liste des documents fournis

Lister les documents fournis au client avec votre produit, en indiquant les numéros de versions

- *le rapport de projet*
- *le manuel d'Installation (en annexe)*
- *le manuel d'Utilisation avec des exemples graphiques (en annexe)*
- *autres...*

4 Conclusions

Développez en tous cas les points suivants :

- *Objectifs atteints / non-atteints*
- *Bilan personnel : points positifs / négatifs*
- *Difficultés particulières*
- *Suites possibles pour le projet (évolutions & améliorations)*

5 Annexes

5.1 Résumé du rapport du TPI / version succincte de la documentation

5.2 Sources – Bibliographie

Liste des livres utilisés (Titre, auteur, date), des sites Internet (URL) consultés, des articles (Revue, date, titre, auteur) ... Et de toutes les aides externes (noms)

5.3 Journal de travail

Date	Durée	Activité	Remarques

Référence à votre journal de travail (en PDF)

5.4 Manuel d'Installation

5.5 Manuel d'Utilisation

5.6 Archives du projet

Media, ... dans une fourre en plastique