



**Crois en toi**  
Tu peux tout déchirer !

# RNCP - Méthodologie de la rédaction du projet

## Framework de la rédaction du projet RNCP

### ▼ 1. Choisir son outil de rédaction

#### ▼ Notion

Fonctionne en blocs

Nombre de blocs illimités pour les particuliers.

Lors de l'inscription choisir "pour les études", lors de la première connexion en bas à droite "bénéficier du forfait education +"

Possibilité de créer différents types de blocs

Notamment des titres dépliant qui permettent de naviguer facilement vers la partie dont on a besoin.

On peut insérer des images, créer des tableaux, des listes, faire une To Do list de ce qu'il vous reste à faire, diviser votre doc en colonnes...

Dès qu'une idée vous vient, vous pouvez ouvrir l'app Notion et la noter dans la partie correspondante.

#### ▼ Word

Puis passer sur Word pour la mise en page finale avec pagination

### ▼ 2. Le plan

Déjà donné dans le référentiel

Possibilité de le rediviser en sous-parties

### ▼ Compétences du référentiel

- Liste des compétences du référentiel qui sont couvertes par le projet
- Faire des références aux compétences dans tout le projet (cf compétence...)
- cf : Référentiel activités & compétences  
<https://www.banque.di.afpa.fr/EspaceEmployeursCandidatsActeurs/recherche-detaillee>

### ▼ Cahier des charges

- Cahier des charges ou expression des besoins du projet
- Cahier des charges :
  - Décris le contexte du projet, les besoins fonctionnels, les objectifs, les paramètres, la sécurité, les normes à respecter...
  - Général, ne rentre pas dans les détails.

### ▼ Gestion de projet

- Gestion de projet (planning et suivi, environnement humain et technique, objectifs de qualité)

### ▼ Spécifications fonctionnelles

- Spécifications fonctionnelles du projet
- "Ensemble des fonctions d'un logiciel ou d'une application afin de déterminer le périmètre fonctionnel du projet".
- Elles se concentrent sur le fonctionnement de l'interface utilisateur.
- Étape en amont des phases de maquettage & de développement.
- Décrire avec précision les interactions avec l'utilisateur pour chaque écran qu'il rencontre.
- Aspect fonctionnel du projet, cas d'utilisation des fonctionnalités, des éléments interactifs, chaque étape du parcours utilisateur.
- Chronophage car doit être fait pour chaque écran.
- Ne pas oublier la mise en place de scénarios de tests.
- Spécifications fonctionnelles générales : specs fonctionnelles, besoin métier
- Spécifications fonctionnelles détaillées : description des fonctionnalités & des sous fonctions.
- Différentes des spécifications techniques (architecture, technologies & matériel)

- Besoin des spécifications fonctionnelles pour réaliser le cahier des charges

## ▼ Spécifications techniques

- Spécifications techniques du projet, élaborées par le candidat, y compris pour la sécurité
- Architecture, environnement, technologie, matériel, domaine & hébergement, accessibilité, sécurité, maintenance...
- Traduction des specs fonctionnelles en moyens & en solutions.
- Exemple de différences avec les spécificités fonctionnelles :
  - Le besoin fonctionnel : un temps de chargement rapide ;
  - La Spécification technique du besoin : la page doit utiliser les standards HTML et contenir des images d'un poids inférieur à 200 Ko.
- Exemple sécurité :
  - Injection de code (injections SQL, JavaScript)
  - Authentification et gestion des sessions
  - Validation des entrées (risque d'attaques XSS qui permet de rediriger vers un autre site ou voler les cookies de connexion quand un utilisateur rentre ses infos, générant un code HTML ou JS ou CSRF)
  - Exposition de données sensibles (mot de passe en texte brut, données personnelles), fuite de données
  - Configuration incorrecte (paramètres par défaut non sécurisés, informations sensibles exposées)
  - Gestion des erreurs (informations sensibles dans les messages d'erreur permet de comprendre la structure de l'application et les points d'entrée potentiels).
  - Dépendance non sécurisée (bibliothèques non sécurisées ou obsolètes)
  - Sécurité des fichiers (accès aux fichiers)
  - Manque de protection contre les attaques de déni de service (trop de demandes malveillantes qui empêchent les utilisateurs réels de se connecter).
    - Buffer overflow : un assaillant envoie une quantité excessive de données à une application, ce qui peut causer un débordement de mémoire et permettre à l'attaquant d'exécuter du code malveillant.
  - Sécurité des API
  - Absence de chiffrement (pour les données en transit ou en repos)

- Déploiement de logiciels non testés (peut entraîner des conflits de logiciels ou des vulnérabilités de sécurité)
- Broken Access Control : avoir accès à une page administrateur grâce à un URL.
- Désérialisation non sécurisé (token qui passe de la partie front à la partie back qui peut-être déchiffré puis rechiffré et donc son rôle peut être modifié et on peut avoir les accès à des ressources admin).
- Sandbox inadéquat
- Falsification des requêtes côté serveur

### ▼ Extraits de code

- Réalisations du candidat comportant les extraits de code les plus significatifs et en les argumentant, y compris pour la sécurité.

### ▼ Jeu d'essai

- Présentation du jeu d'essai élaboré par le candidat de la fonctionnalité la plus représentative (données en entrée, données attendues, données obtenues).
- Ensemble des données de test
- Évalue la performance, la fonctionnalité, la sécurité, la compatibilité & la fiabilité.
- On peut faire des simulations de scénarios réels.
- On vérifie la rédaction des risques.
- Peut-être fait manuellement ou automatiquement via des outils et des scripts de test
- Doit être représentatif des conditions réelles d'utilisation & couvrir un large éventail de scénarios.
- Démonstration de remplissage de la base de données avec des données, l'explication du résultat attendu, l'effectuation d'une ou plusieurs actions par l'étudiant (en tant qu'utilisateur).
- Les jeux d'essai peuvent contenir des événements de types Appeler, Retour, Exception, Requête, Réponse, Trace à granularité fine et Emuler. Chaque événement ajouté devient une étape dans le jeu d'essai. Un jeu d'essai de base inclut systématiquement tous les événements Appeler, Retour et Exception

### ▼ Veille

- Description de la veille, effectuée par le candidat durant le projet, sur les vulnérabilités de sécurité.
- À adapter en fonction de votre projet

- Faire une veille consiste à vérifier périodiquement que ces points ne sont pas en danger. Bien préciser les failles trouvées et comment elles ont été corrigées. Parler de votre démarche de recherche.
- Garder une trace de cette veille

### ▼ Situation de travail

- Description d'une situation de travail ayant nécessité une recherche et effectuée par le candidat durant le projet.
- Exemple d'une partie du plan qu'il faudra rediviser en plusieurs sous parties

## ▼ 3. Brainstorming

Mettre toutes nos idées en vrac dans chacune partie du plan

## ▼ 4. Le plan détaillé

Rassembler nos idées en sous parties

On commence à construire petit à petit

Organiser en paragraphes ayant des liens entre eux

## ▼ 5. L'introduction

Évocation du contexte du projet, pourquoi avoir choisi de travailler là-dessus, quelle était la problématique, story telling,

Combien de personnes ont travaillé sur ce projet, qui a fait quoi, qui est l'équipe, se présenter.

## ▼ 6. La conclusion

Ce que vous avez appris du projet, ce qu'il apporte en général et ce qu'il vous a apporté d'un point de vue professionnel & personnel.

Ouverture sur les prochaines features du projet ou les prochaines étapes, avec les deadlines que vous vous fixez.

Possibles remerciements

## ▼ 7. La rédaction

Faire des phrases avec toutes nos idées, les liées les unes aux autres

Avoir un bon équilibre entre la partie rédactionnelle et les schémas (40/60 ou 50/50)

## ▼ 8. La mise en page

Mise en page finale sur Word (sur Google Drive si pas de Licence)

Vous pouvez copier-coller, ça générera automatiquement les titres du sommaire

- Nombre de pages : 40 à 60 pages hors annexes, environ 75 000 caractères

### ▼ La police

- Time New Roman, taille 12
- Arial, taille 11
- Calibri, taille 12
- Garder la même police partout
- Interligne 1,5
- Texte justifié

### ▼ Les chapitres

Faire un saut de page à chaque nouveau chapitre

### ▼ Les marges

2,54cm à gauche, droite, haut, bas

### ▼ Page de garde

Page de garde avec votre nom, nom de votre projet, date, nom du titre (Concepteur Développeur d'Application), lieu des études.

### ▼ Sommaire

Page sommaire : plan avec numéro de page (peut se faire automatiquement sur word)

### ▼ Entête

Entête : nom du projet, votre nom

### ▼ Pied de page

Pied de page : numérotation des pages, date, logo de l'école

# Holberton

## actual digital school

### ▼ 9. Faire relire par une IA

Paragraphe par paragraphe, bien faire attention à ce que l'IA suggère.

Permet d'éliminer des fautes d'orthographe, de reformuler des tournures de phrases qui ne serait pas adaptées

Demander des conseils à l'IA, à voir si on les suit

## ▼ 10. Relecture des proches

Permet de vérifier que le contenu est compréhensible

Ne pas hésiter à faire relire par plusieurs personnes

## ▼ 11. Relecture par les pairs

Permet d'avoir un retour technique.

Prenez le temps de vous relire entre vous