



## RÉFÉRENCIEL DE COMPÉTENCES POUR DÉVELOPPEURS ET DÉVELOPPEUSES

Le framework pour vous positionner et évoluer dans  
votre carrière.

Sébastien Carceles



# Avant-propos

## Objectif de ce référentiel

Les métiers du développement sont en mutation perpétuelle. Les technologies évoluent, les méthodes de travail changent, les attentes des clients aussi. Il est difficile de s'y retrouver et de savoir où l'on se situe dans ce paysage mouvant.

Les entreprises ont besoin de référentiels pour évaluer les compétences de leurs employés. Les développeur-euses ont besoin de référentiels pour se positionner et évoluer dans leur carrière. Ce référentiel a pour but de répondre à ces deux besoins.

## Positions

Les entreprises nomment les postes de leurs employé-es de différentes manières. Cette nomenclature n'est pas normée : un poste peut être appelé "développeur / développeuse", "ingénieur-e de développement", "consultant / consultante", "expert / experte", "architecte", "lead", "manager", etc. Il peut correspondre à un poste junior, mid-level, senior, lead, manager, etc, et couvrir différents domaines de responsabilités, qui diffèrent d'une entreprise à l'autre.

Ce référentiel propose une nomenclature des postes, qu'on appellera "positions", ainsi qu'une description des compétences attendues pour chaque position.

Dans l'industrie du développement logiciel, les noms de postes ou de positions sont, la plupart du temps, en anglais. C'est pourquoi ce référentiel utilise des noms en anglais.

## Caractéristiques et compétences

À chaque position correspond un ensemble de compétences, organisées par caractéristique principale. Les caractéristiques sont au nombre de 4 : **DEXTERITY - Technical skill**, **STRENGTH - Get stuff done**, **WISDOM - Impact** et **CHARISMA - Communication & leadership**.

### **DEXTERITY - Technical skill**

C'est une mesure de la profondeur de la connaissance d'une technologie.

### **STRENGTH - Get stuff done**

C'est une mesure de la capacité à résoudre des problèmes.

### **WISDOM - Impact**

C'est une mesure de la capacité à prendre des décisions et à avoir de l'impact.

### **CHARISMA - Communication & leadership**

C'est une mesure de la capacité à influencer et inspirer les autres.

## **Positionnement**

Pour être légitime dans une position donnée, il faut en maîtriser toutes les compétences. Il faut également posséder les compétences des positions précédentes dans le chemin parcouru.

## **Évolution de position**

Pour évoluer d'une position à l'autre, il faut naturellement acquérir, consolider et maîtriser toutes les compétences qui la composent.

## **Source**

Ce référentiel est issue du post de blog "Sharing Our Engineering Ladder"<sup>1</sup> de la société Rent the Runway.

Il a ensuite été traduit, retravaillé et adapté pour le marché français.

## **Images**

Photo de couverture par Annie Spratt<sup>2</sup> sur Unsplash.

---

1. <https://dresscode.renttherunway.com/blog/ladder>

2. <https://unsplash.com/fr/@anniespratt>

## À propos du genre

Autant que possible, le présent document utilise des termes non genrés. Dans les cas où c'est nécessaire, le point médian et les pronoms non genrés permettent d'inclure tous les genres.

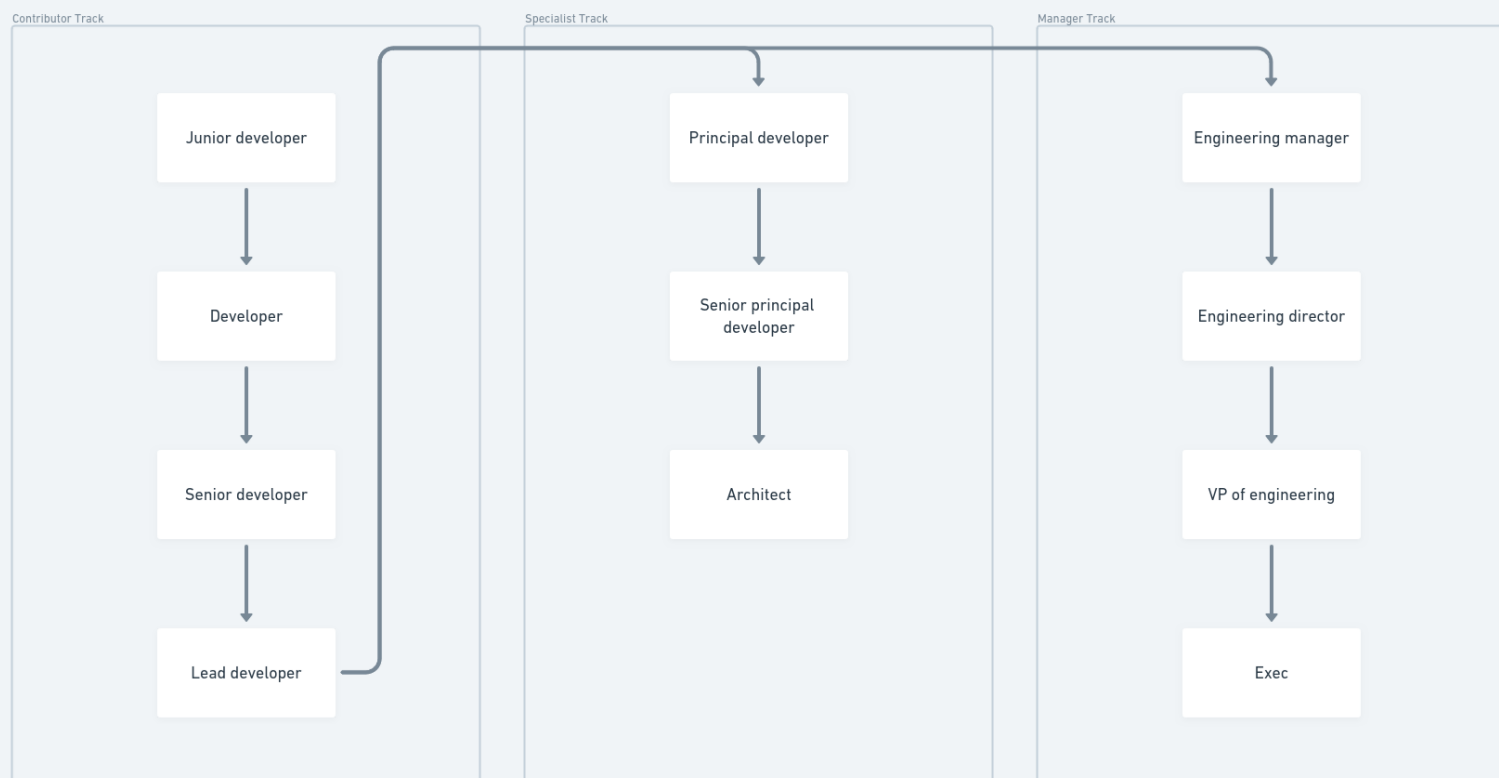
## Propriété intellectuelle

TODO open licence?



# Première partie

## Les différents parcours



Made with Whimsical

La carrière du développeur ou de la développeuse peut emprunter plusieurs chemins. Ils ne sont pas rigoureusement étanches et des passages d'un chemin à l'autre sont possibles.

### Contributor Track

Dans tous les cas, une carrière commence avec la première position du premier parcours, nommé "Contributor Track". Dans ce parcours, au travers des positions qui le composent, la personne acquiert des compétences clés pour être une excellente contributrice au projet.

TODO la suite



Deuxième partie

Contributor Track

# Junior developer

Expérience: 0 à 2 ans

## **DEXTERITY - Technical skill**

Possède une connaissance approfondie des concepts fondamentaux de l'informatique.

Se concentre sur la croissance en tant qu'ingénieur-e, en apprenant les outils, les ressources et les processus existants.

## **STRENGTH - Get stuff done**

Développe ses compétences en productivité en apprenant le contrôle de source, les éditeurs, le système de construction et autres outils, ainsi que les meilleures pratiques de test.

Est capable de prendre des sous-tâches bien définies et de les accomplir.

## **WISDOM - Impact**

Développe ses connaissances sur un seul composant de l'architecture du projet.

## **CHARISMA - Communication & leadership**

Est efficace dans la communication de l'état d'avancement à l'équipe.

Manifeste les valeurs fondamentales de l'entreprise, se concentre sur la compréhension et la mise en pratique de ces valeurs.

Accepte gracieusement les retours d'information et apprend de tout ce qu'il fait.

# Developer

Expérience: 2 à 5 ans

## **DEXTERITY - Technical skill**

Écrit du code correct et propre avec des conseils ; suit systématiquement les meilleures pratiques définies sur le projet.

Participe à la conception technique des fonctionnalités avec des conseils.

## **DEXTERITY - Technical skill**

Fait rarement la même erreur deux fois, commence à se concentrer sur l'acquisition d'une expertise dans un ou plusieurs domaines (par exemple : développement dans un langage donné, meilleures pratiques de performance, utilisation efficace des bases de données, messagerie, etc.)

Apprend rapidement et progresse régulièrement sans avoir besoin de commentaires significatifs constants de la part d'ingénieur·es plus expérimenté·es.

## **STRENGTH - Get stuff done**

Fait des progrès réguliers sur les tâches ; sait quand demander de l'aide pour se débloquer.

## **STRENGTH - Get stuff done**

Est capable de prendre en charge des fonctionnalités de petite à moyenne envergure, de la conception technique à l'achèvement.

Est capable de hiérarchiser les tâches ; évite de s'enliser dans des détails sans importance et des "discussions sans fin".

## **WISDOM - Impact**

Autonome dans au moins un grand domaine du code avec une compréhension globale des autres composants.

## **WISDOM - Impact**

Est capable d'assurer une assistance en cas d'urgence pour leur domaine, y compris les systèmes avec lesquels iel n'est pas familier·ère.

## **CHARISMA - Communication & leadership**

Donne des commentaires opportuns et utiles aux pairs et aux responsables.

Communique les hypothèses et obtient des éclaircissements sur les tâches dès le départ afin de minimiser le besoin de retravailler.

## **CHARISMA - Communication & leadership**

Sollicite les commentaires des autres et est désireux de trouver des moyens de s'améliorer.

Comprend comment son travail s'inscrit dans le projet global et identifie les problèmes liés aux exigences.

# Senior developer

Expérience: 5 à 8 ans

## **DEXTERITY - Technical skill**

Comprend et prend des décisions de conception bien raisonnées et des compromis dans leur domaine.

Est capable de travailler dans d'autres domaines du code avec des conseils.

Ne s'agite pas lors du débogage.

## **DEXTERITY - Technical skill**

Démontre une connaissance des tendances de l'industrie, de l'infrastructure du projet et de son système de construction.

## **STRENGTH - Get stuff done**

Persistant face aux obstacles ; les résout efficacement, en faisant appel à d'autres personnes si nécessaire.

Nécessite un minimum de direction / supervision.

## **STRENGTH - Get stuff done**

Prend l'initiative de résoudre les problèmes avant qu'ils ne lui soient assignés. Recherche des preuves empiriques par le biais de preuves de concept, de tests et de recherches externes.

Livre des produits complexes à l'équipe de QA / Produit, qu'il estime bien préparés et exempts de bugs.

### **WISDOM - Impact**

Responsable de bout en bout sur des projets de complexité croissante ; contribue au code commun.

Examine les cas de test et conseille l'équipe de QA / Produit sur l'impact du code adjacent / régression.

Comprend l'activité commerciale afférente à son domaine d'activité.

### **WISDOM - Impact**

Collabore avec le produit et l'analyse et définit des exigences qui tiennent compte des besoins de toutes les parties.

Possède de l'empathie envers l'utilisateur du logiciel qu'il produit et utilise cette empathie pour orienter la prise de décision.

Identifie les problèmes / risques de son propre travail et de celui des autres.

### **CHARISMA - Communication & leadership**

Communique les décisions techniques par le biais de documents de conception ou de présentations techniques.

Mentore les ingénieur-es juniors par le biais de la collaboration, de l'examen de conception et de l'examen de code. Contribue fréquemment aux réunions informelles et aux démonstrations.

### **CHARISMA - Communication & leadership**

Communique efficacement entre les fonctions. Est capable de bien travailler avec le produit, le design, l'analyse, etc., si nécessaire.

Identifie de manière proactive les problèmes liés aux exigences (manque de clarté, incohérences, limitations techniques) pour son propre travail et le travail adjacent, et communique ces problèmes tôt pour aider à corriger le tir.

# Lead developer

Expérience: 6 à 10 ans

Troisième partie

Specialist Track



# Principal developer

Expérience: 10 à 15 ans

# Senior principal developer

Expérience: 15 à 20 ans

Architect

Quatrième partie

Manager Track

# Engineering manager

Engineering director

VP of engineering

Exec



# Table des matières

Avant-propos	i
I Les différents parcours	1
II Contributor Track	3
Junior developer	4
Developer	5
Senior developer	7
Lead developer	9
III Specialist Track	10
Principal developer	11
Senior principal developer	12
Architect	13
IV Manager Track	14
Engineering manager	15
Engineering director	16
VP of engineering	17
Exec	18