**Robotique -> Développement web**

0. expliquer que la robotique n'est pas très loin du dev

1.Se challenger en apprenant/exerçant de nouvelles choses

2. Fuir l'ennui au travail

3.Faire de sa passion son métier

**Principes d'agile:**

Les individus et leurs interactions plus que les processus et les outils

Des logiciels opérationnels plus qu’une documentation exhaustive

La collaboration avec les clients plus que la négociation contractuelle

L’adaptation au changement plus que le suivi d’un plan

**Java :**

**Les bases de java :**

-> types de données, structures conditionnelles, les boucles

**Java oop:**

-> déclarer/ initialiser les objets, Define and use fields and methods, including instance, static and overloaded methods,

Initialize objects and their members using instance and static initialiser statements and constructors, variable scopes, apply encapsulation and make objects immutable

->Create and use subclasses and superclasses, including abstract classes

->Utilize polymorphism and casting to call methods, differentiate object type versus reference type

->Create and use interfaces, identify functional interfaces, and utilize private, static, and default methods

->Create and use enumerations

**Exception Handling:**

-> try/catch/finally, throw, throws, multi catch

->Create and use custom exceptions

**Working with Arrays and Collections:**

-> generics/ wildcards, array, List, Map

->Sort collections and arrays using Comparator and Comparable interfaces

**Working with Streams and Lambda expressions**

-> Implement functional interfaces using lambda expressions, including interfaces from the java.util.function package

-> Use Java Streams to filter, transform and process data

-> Perform decomposition and reduction, including grouping and partitioning on sequential and parallel streams

**Database Applications with JDBC:**

-> Connect to and perform database SQL operations, process query results using JDBC API

**Annotations**

->Create, apply, and process annotations

**Nouveautés du java 8:**

- Lambda

- Streams

- Les interfaces fonctionnelles

- Optionnal

**Angular :**

* Interaction avec le backend-développé en java 8 (Spring boot) – à l’aide des requêtes http et les opérations CRUD en utilisant les services et le dependency injection
* Travail avec components, le databinding, les directives pour la création des pages web dynamiques
* Utilisation du routing pour passer d’une page web à une autre
* Utilisation des formulaires (template-driven & reactive forms) pour assurer l’interaction de l’utilisateur avec l’application
* Utilisation des pipes pour la transformation des outputs
* Utilisation de l’authentification et de la protection des routes pour assurer la sécurité de l’application
* Faire de l’animation des objets html sur des pages web