

2021

Solution building bloc



Karl Menino

Version : 1.0

12/10/2021

Table des matières

Table des matières

| | | |
|----|--|---|
| 1. | HISTORIQUE DES REVISIONS..... | 5 |
| 2. | Blocs de construction (SBB) | 5 |
| A. | Build du projet | 5 |
| | Nom du bloc de construction : | 5 |
| | Fonctionnalité fournie : | 5 |
| | Lien vers un exemple d'implémentation ou d'interfaces : | 5 |
| | Travail exceptionnel pour terminer ce bloc de construction : | 5 |
| | Alignement architectural : | 5 |
| B. | Api Rest..... | 6 |
| | Nom du bloc de construction : | 6 |
| | Fonctionnalité fournie : | 6 |
| | Lien vers un exemple d'implémentation ou d'interfaces : | 6 |
| | Travail exceptionnel pour terminer ce bloc de construction : | 6 |
| | Alignement architectural : | 6 |
| C. | MVC..... | 6 |
| | Nom du bloc de construction : | 6 |
| | Fonctionnalité fournie : | 6 |
| | Lien vers un exemple d'implémentation ou d'interfaces : | 6 |
| | Travail exceptionnel pour terminer ce bloc de construction : | 6 |
| | Alignement architectural : | 7 |
| D. | Micro-service | 7 |
| | Nom du bloc de construction : | 7 |
| | Fonctionnalité fournie : | 7 |
| | Lien vers un exemple d'implémentation ou d'interfaces : | 7 |

| | |
|--|----|
| Travail exceptionnel pour terminer ce bloc de construction : | 7 |
| Alignement architectural : | 7 |
| E. Automatisation | 8 |
| Nom du bloc de construction : | 8 |
| Fonctionnalité fournie : | 8 |
| Lien vers un exemple d'implémentation ou d'interfaces : | 8 |
| Travail exceptionnel pour terminer ce bloc de construction : | 8 |
| Alignement architectural : | 8 |
| F. Sécurité | 8 |
| Nom du bloc de construction : | 8 |
| Fonctionnalité fournie : | 9 |
| Lien vers un exemple d'implémentation ou d'interfaces : | 9 |
| Travail exceptionnel pour terminer ce bloc de construction : | 9 |
| Alignement architectural : | 9 |
| G. Base de données | 9 |
| Nom du bloc de construction : | 9 |
| Fonctionnalité fournie : | 9 |
| Lien vers un exemple d'implémentation ou d'interfaces : | 9 |
| Travail exceptionnel pour terminer ce bloc de construction : | 10 |
| Alignement architectural : | 10 |
| H. AWS | 10 |
| Nom du bloc de construction : | 10 |
| Fonctionnalité fournie : | 10 |
| Lien vers un exemple d'implémentation ou d'interfaces : | 10 |
| Travail exceptionnel pour terminer ce bloc de construction : | 10 |
| Alignement architectural : | 11 |
| I. Test | 11 |
| Nom du bloc de construction : | 11 |
| Fonctionnalité fournie : | 11 |



| | |
|--|----|
| Lien vers un exemple d'implémentation ou d'interfaces | 11 |
| Travail exceptionnel pour terminer ce bloc de construction : | 12 |
| Alignement architectural : | 12 |

1. HISTORIQUE DES REVISIONS

| Version | Description | Date | Valideur |
|---------|-------------|------------|-------------|
| 1.0 | | 12/10/2021 | Menino karl |
| | | | |
| | | | |

Ce document décrit les Solutions Building bloc.

Karl menino,

Architecte Logiciel,

MedHead.

2. Blocs de construction (SBB)

A. Build du projet

Nom du bloc de construction :

Création du projet.

Fonctionnalité fournie :

Ce bloc de construction permet de standardiser la création de projet grâce à un outils externe.

Lien vers un exemple d'implémentation ou d'interfaces :

SpringBoot: <https://spring.io/projects/spring-boot>

Maven: <https://maven.apache.org/>

Spring initializr: <https://start.spring.io/>

Travail exceptionnel pour terminer ce bloc de construction :

Serveur tomcat.

Alignement architectural :

Cette SBB valide les principes de l'architecture informatique(B)

B. Api Rest

Nom du bloc de construction :

Api Rest.

Fonctionnalité fournie :

Ce bloc permet de rendre notre application restFull

Lien vers un exemple d'implémentation ou d'interfaces :

Spring boot starter Web:

<https://mvnrepository.com/artifact/org.springframework.boot/spring-boot-starter-web>

Spring rest: <https://spring.io/guides/gs/rest-service/>

Travail exceptionnel pour terminer ce bloc de construction :

Controller

Alignement architectural :

Ce bloc valide l'hypothèse de développement suivant :

- Fournir une API RESTful qui tient les intervenants médicaux informés en temps réel sur : le lieu où se rendre et ce qu'ils doivent faire
- S'assure que la PoC peut être facilement intégrée dans le développement futur

C. MVC

Nom du bloc de construction :

Spring MVC

Fonctionnalité fournie :

Ce bloc permet de standardiser l'architecture de notre application

Lien vers un exemple d'implémentation ou d'interfaces :

OpenClassroom exemple: <https://openclassrooms.com/fr/courses/4504771-simplifiez-le-developpement-dapplications-java-avec-spring>

Travail exceptionnel pour terminer ce bloc de construction :

Model/DAO/Service

Alignement architectural :

Ce bloc valide l'hypothèse de développement suivant :

- Rendre le code facilement maintenable

Ce bloc valide les principes d'architecture suivant :

- Principe B2 : Clarté grâce à une séparation fine des préoccupations
- Principe B6 : Possibilité d'extension grâce à des fonctionnalités déclenchées par les événements

D. Micro-service

Nom du bloc de construction :

Spring Microservice

Fonctionnalité fournie :

Ce bloc permet de standardiser l'architecture de notre application

Lien vers un exemple d'implémentation ou d'interfaces :

Spring microservice: <https://spring.io/microservices>

OpenClassroom: <https://openclassrooms.com/fr/courses/4668056-construisez-des-microservices>

Travail exceptionnel pour terminer ce bloc de construction :

A voir dans l'application final

Alignement architectural :

Ce bloc valide l'hypothèse de développement suivant :

- Rendre le code facilement maintenable

Ce bloc valide les principes d'architecture suivant :

- Principe B2 : Clarté grâce à une séparation fine des préoccupations
- Principe B6 : Possibilité d'extension grâce à des fonctionnalités
Déclenchées par les événements

E. Automatisation

Nom du bloc de construction :

Automatisation des tâches du projet

Fonctionnalité fournie :

Fonctionnalité DevOps

Lien vers un exemple d'implémentation ou d'interfaces :

GitHub: <https://github.com/> &

<https://openclassrooms.com/fr/courses/5641721-utilisez-git-et-github-pour-vos-projets-de-developpement>

GitLab: <https://about.gitlab.com/> &

<https://openclassrooms.com/fr/courses/2035736-mettez-en-place-lintegration-et-la-livraison-continues-avec-la-demarche-devops>

DevOps: <https://openclassrooms.com/fr/courses/6093671-decouvrez-la-methodologie-devops>

Travail exceptionnel pour terminer ce bloc de construction :

Mise en place de la méthodologie DevOps.

Alignement architectural :

Ce bloc valide l'hypothèse de développement suivant :

- Rendre le code facilement partageable,
- Fournir des pipelines d'intégration et de livraison continue (CI/CD)

Ce bloc valide les principes d'architecture suivant :

- Principe B3 : Intégration et livraison continues
- Principe C2 : Référentiel d'architecture centralisé et organisé comme source de référence

F. Sécurité

Nom du bloc de construction :

Sécurisation de l'application

Fonctionnalité fournie :

Ce bloc permet de sécuriser l'application en suivant les recommandations OWASP

Lien vers un exemple d'implémentation ou d'interfaces :

Spring Security: <https://spring.io/projects/spring-security> & <https://openclassrooms.com/fr/courses/7137776-securisez-votre-application-web-avec-spring-security>

Clef Api: <https://stackoverflow.com/questions/48446708/securing-spring-boot-api-with-api-key-and-secret>

Https: <https://www.thomasvitale.com/https-spring-boot-ssl-certificate/>

OWASP: <https://owasp.org/>

Travail exceptionnel pour terminer ce bloc de construction :

Configuration de Spring Security. Mise en place de la clef Api et obligation de passer par le protocole HTTPS.

Alignement architectural :

Ce bloc valide les principes d'architecture suivant :

- Principe B5 : Sécurité de type « shift-left »

Ce bloc valide l'hypothèse de développement suivant :

- S'assurer que toutes les données du patient sont correctement protégées.

G. Base de données

Nom du bloc de construction :

PostgreSQL BDD

Fonctionnalité fournie :

- Persistance des données
- Constitution de la BDD test

Lien vers un exemple d'implémentation ou d'interfaces :

Maven PostgreSQL:

<https://mvnrepository.com/artifact/org.postgresql/postgresql>

Spring Data: <https://spring.io/projects/spring-data>

ETL: <https://www.talend.com/fr/resources/guide-etl/>

FINESS: <http://finess.sante.gouv.fr/fininter/jsp/index.jsp>

Travail exceptionnel pour terminer ce bloc de construction :

- Création et déploiement de la BDD
- Importation des données de FINESS
- Peuplement de la BDD

Alignement architectural :

Ce bloc valide les principes d'architecture suivant :

- Principe C4 : Favoriser une culture de “learning” avec des preuves de concept, des prototypes et des Spike

H. AWS

Nom du bloc de construction :

Déploiement de L'application final sur AWS

Fonctionnalité fournie :

Ce bloc permet le déploiement de notre application final.

Lien vers un exemple d'implémentation ou d'interfaces :

AWS: <https://aws.amazon.com/fr/> &

<https://openclassrooms.com/fr/courses/4810836-decouvrez-le-cloud-avec-amazon-web-services>

Spring Cloud Gateway : <https://spring.io/projects/spring-cloud-gateway>

Spring Cloud Config : <https://spring.io/projects/spring-cloud-config>

Spring Cloud Netflix : <https://spring.io/projects/spring-cloud-netflix>

OpenClassroom: <https://openclassrooms.com/fr/courses/4668216-optimisez-votre-architecture-microservices>

Travail exceptionnel pour terminer ce bloc de construction :

A voir lors de l'application final.

Alignement architectural :

Ce bloc valide les principes d'architecture suivant :

- Principe A2 : Maximiser les avantages pour l'entreprise.
- Principe B1 : Continuité des activités des systèmes critiques pour les patients
- Principe C3 : Normes ouvertes pour garantir des normes élevées

Ce bloc valide l'hypothèse de développement suivant :

- Plus de 90 % des cas d'urgence sont acheminés vers l'hôpital compétent le plus proche du réseau
- Nous obtenons un temps de réponse de moins de 200 millisecondes avec une charge de travail allant jusqu'à 800 requêtes par seconde, par instance de service ;
- S'assurer que les équipes de développement chargées de cette PoC sont en mesure de l'utiliser comme un jeu de modules de construction pour d'autres modules.

I. Test

Nom du bloc de construction :

Test de l'application

Fonctionnalité fournie :

Ce bloc s'assure de la qualité de notre application

Lien vers un exemple d'implémentation ou d'interfaces :

- OpenClassroom: <https://openclassrooms.com/fr/courses/6100311-testez-votre-code-java-pour-realiser-des-applications-de-qualite>
- Junit: <https://junit.org/junit5/>
- Mockito: <https://site.mockito.org/>
- Surefire: <https://maven.apache.org/surefire/maven-surefire-plugin/>
- JACOCO : <https://www.jacoco.org/>
- Cucumber: <https://cucumber.io/>
- SONARCLOUD: <https://sonarcloud.io/>
- Selenium: <https://www.selenium.dev/>
- GATLING: <https://www.baeldung.com/introduction-to-gatling>

Travail exceptionnel pour terminer ce bloc de construction :

- Mise en place des tests unitaire, d'intégration et d'acceptation.
- Vérifier qualité du code avec SONARCLOUD.

Alignement architectural :

Ce bloc valide les principes d'architecture suivant :

- Principe B4 : Tests automatisés précoces, complets et appropriés
- Principe C4 : Favoriser une culture de "learning" avec des preuves de concept, des prototypes et des Spike

Ce bloc valide l'hypothèse de développement suivant :

- S'assurer que votre PoC est entièrement validée avec des tests d'automatisation reflétant la pyramide de test (tests unitaires, d'intégration, d'acceptation et E2E) et avec des tests de stress pour garantir la continuité de l'activité en cas de pic d'utilisation