

*My Booking Services*

-

# Méthodologie de Projet

-

*SPE-CLO5*

## I. Gestion du temps, cycles de travail (Sprint)

### A. Sprints d'une semaine et réunions hebdomadaires

Dès le début du projet, nous avons suivi un rythme de travail constitué de cycles d'une semaine (Sprint). Le début de tout Sprint est marqué d'une réunion entre tous les membres. Ces réunions sont définies par un Ordre Du Jour précis, et se voient attribuer une limite de temps.

Leur structure est toujours la même, à savoir :

- Présentation par chacun des actions entreprises lors du cycle de travail précédent
- Remontée au Groupe des problèmes rencontrés
- Recherche de résolution des problèmes éventuels, si le temps imparti le permet
- Mise en perspective de l'avancement du travail vis-à-vis du CDC (Sujet ETNA)
- Identification des priorités et des sujets à traiter lors du prochain Sprint
- Répartition des tâches entre les membres du Groupe
- Mise à jour du Board Kanban.

### B. Quels outils ?

- Hangout pour les réunions
- Google Docs pour les Compte-Rendus de Réunion si applicable
- Dashboard Kanban GitLab pour le suivi des tâches et leur répartition

## II. Gestion des Ressources et méthodologie DevOps

### A. Mobilisation des Ressources

Si la pondération des effectifs sur chaque tâche est dictée par des priorités, la répartition des Ressources sur les différentes tâches est faite en fonction des compétences, mais surtout des appétences.

Par exemple, si certains membres souhaitent monter en compétence sur le volet Ansible du projet, nous mettons en place un roulement sur la réalisation des tâches, ou une division de celles-ci (un Rôle Ansible par membre intéressé...).

Pour toute tâche liée à l'infrastructure Système, un tandem Développeur/Ops est créé.

La résolution des problèmes majeurs mobilise l'ensemble du Groupe, qu'ils soient liés au Développement ou à la partie Système/Opérations.

## B. Quels outils ?

- Dashboard Kanban GitLab pour le suivi des tâches et leur répartition.

# III. Méthodes de Développement

## A. Méthode, documentation et templating

Nous avons mis en place du Développement en TDD (Test-Driven Development), où les spécifications des Services sont écrites avant l'implémentation.

Grâce à la mise en place rapide de documentations Swagger autogénérées, nous avons pu facilement normaliser les interfaces des Services et uniformiser leur développement.

Un template de projet NestJs nous a par ailleurs permis de faciliter l'initialisation des Services, et de les uniformiser.

## B. Quels outils ?

- Swagger pour la documentation
- Repository GitLab de template

# IV. Intégration / Déploiement continu

## A. Travail en CI / CD


Nous avons mis en place des pipelines d'intégration continue des sources du projet, ce afin de garantir la qualité de la base de code.

Le déploiement continu de la solution applicative et des services associés (reverse proxy...) est également intégré à nos pipelines.

## B. Quels outils ?

- Pipelines GitLab CI/CD, Runner + Docker executor.

## V. Annexe

 **Swagger**  
Powered by SMARTBEAR

# Property Microservice 1.0 OAS3

A REST API for property management.

### Hotels

POST	/services/users/api/v1/hotels
GET	/services/users/api/v1/hotels
PUT	/services/users/api/v1/hotels
GET	/services/users/api/v1/hotels/{id}
DELETE	/services/users/api/v1/hotels/{id}

### Rooms

POST	/services/users/api/v1/rooms
GET	/services/users/api/v1/rooms
PUT	/services/users/api/v1/rooms
GET	/services/users/api/v1/rooms/{id}
DELETE	/services/users/api/v1/rooms/{id}

Steeve &gt; My Booking Services &gt; Milestones

Open 5 Closed 0 All 5

Filter by milestone name

**1. Define Project Architecture**

Mar 16, 2020–Apr 24, 2020

Steeve / My Booking Services

4 Issues · 0 Merge Requests

0

**2. Build Environment**

Mar 16, 2020–Apr 24, 2020

Steeve / My Booking Services

10 Issues · 0 Merge Requests

0

**5. Achieve Continuous Deployment**

Apr 24, 2020–May 14, 2020

Upcoming Steve / My Booking Services

1 Issue · 0 Merge Requests

0

**4. Achieve Continuous Integration**

Apr 24, 2020–May 14, 2020

Upcoming Steve / My Booking Services

4 Issues · 0 Merge Requests

0

**3. Build Application & Tests**

Mar 16, 2020–May 14, 2020

Steeve / My Booking Services

12 Issues · 0 Merge Requests

0

In Progress13

Build User Service  
Services  
#4

Build Booking Service  
Services  
#7

Build User Service Tests  
Services Testing  
#9

Build Booking Service Tests  
Services Testing  
#11

Build Mailing Service Tests  
Services Testing  
#12

Secure all Services (JWT + Bcrypt + Roles)  
Security Services  
#5

Create Architecture Diagram  
Architecture Documentation Project Management  
#13

Done8

Design Database  
Architecture Services  
#18

Define Micro-Service bounds  
Architecture Services  
#27

Build Property Service  
Services  
#6

Build Property Service Tests  
Services Testing  
#10

Build Mailing Service  
Services  
#8

Orchestrate Service Image Building  
Docker Services  
#15

Setup Swagger on all Services  
Documentation Services  
#29

Done & Deployed9

[ALL] Add ssh keys to Nodes  
Ansible Environment  
#21

[MANAGER] Install & configure Docker  
Ansible Docker Environment  
#1

[MANAGER] Create Swarm  
Ansible Docker Environment  
#22

[WORKERS] Make Workers join Swarm  
Ansible Docker Environment  
#23

[MANAGER] Create and run Registry  
Ansible Docker Environment  
#2

[MANAGER] Install & configure Gitlab  
Ansible Environment  
#3

[MANAGER] Create Gitlab users  
Ansible Environment  
#24



Search or jump to...

Steeve &gt; My Booking Services &gt; Pipelines

All 251 Pending 0 Running 0 Finished 251 Branches Tags

Run Pipeline

Clear

Status	Pipeline	Triggerer	Commit	Stages	
<span>passed</span>	#251 latest		P development -> fcf5c84 testing cache	<span>✓</span> <span>✓</span> <span>✓</span>	⌚ 00:13:13 📅 1 hour ago
<span>passed</span>	#250		P development -> a96c7e36 testing cache	<span>✓</span> <span>✓</span>	⌚ 00:05:40 📅 1 hour ago
<span>failed</span>	#249		P development -> 0528ae20 testing cache	<span>✓</span> <span>✗</span>	⌚ 00:05:03 📅 1 hour ago
<span>failed</span>	#248		P development -> 1a030902 testing cache	<span>✗</span> <span>⏸</span>	⌚ 00:04:05 📅 1 hour ago
<span>canceled</span>	#247		P development -> 35027edc testing cache	<span>⏸</span> <span>⏸</span>	⌚ 00:00:02 📅 1 hour ago
<span>failed</span>	#246		P development -> 306438f7 Update .gitlab-ci.yml	<span>✓</span> <span>✗</span> <span>⏸</span>	⌚ 00:11:21 📅 19 hours ago
<span>canceled</span>	#245		P development -> 4b91f306 Update .gitlab-ci.yml	<span>✓</span> <span>⏸</span> <span>⏸</span>	⌚ 00:03:23 📅 19 hours ago
<span>failed</span>	#244		P development -> c4b89291 Update .gitlab-ci.yml	<span>✓</span> <span>✗</span> <span>⏸</span>	⌚ 00:04:20 📅 19 hours ago
<span>passed</span>	#243		P development -> 5801f021 Update .gitlab-ci.yml	<span>✓</span> <span>✓</span> <span>✓</span>	⌚ 00:05:40 📅 19 hours ago
<span>failed</span>	#242		P development -> 0db4ba8c Update .gitlab-ci.yml	<span>✓</span> <span>✗</span> <span>⏸</span>	⌚ 00:02:05 📅 19 hours ago
<span>failed</span>	#241		P development -> 7ec4ffb1 Update .gitlab-ci.yml	<span>✓</span> <span>✗</span> <span>⏸</span>	⌚ 00:02:07 📅 19 hours ago