

# Table des matières

Introduction Générale . . . . .	1
<b>1 Nom Chapitre</b>	<b>2</b>
1.1 Titre . . . . .	2
1.1.1 Sous-Titre . . . . .	2
1.1.1.1 Sous-Sous-Titre . . . . .	2
Conclusion . . . . .	4
<b>Bibliographie</b>	<b>a</b>

# Table des figures

1.1	Logo Docker. . . . .	2
-----	----------------------	---

# Liste des tableaux

1.1	Description textuelle du diagramme de cas d'utilisation s'authentifier . . .	3
-----	--	---

# Manifests

1.1	Deployment MySQL . . . . .	4
-----	----------------------------	---

# Introduction Générale

Les objectifs d'une entreprise de services sont d'une part la satisfaction de ses clients à travers des services répondant à leurs attentes et réalisés dans les budgets et les délais impartis, et d'autre part l'optimisation des ressources afin que l'entreprise puisse réaliser ses travaux de manière rentable. Dans une banque ayant un réseau étendu, et de par le fait que les exigences des clients sont persistantes et importantes, les simples agences dispersées partout ne peuvent plus satisfaire les clients surtout dans une ère numérique dominée par les innovations technologiques et cherchant à optimiser les services et les maximiser tout en assurant le confort des clients et leurs satisfactions. Amen Banque a souhaité s'engager dans le développement de l'informatique à forte valeur ajoutée, d'où le développement d'un progiciel d'un e-Bank dans le but est de faciliter les transactions des clients tout en les garantissant un taux maximal d'efficacité, flexibilité et confort et un taux minimal de perte de temps. Pour mener ce projet à terme, il sera entamé selon la méthode agile Scrum et les artefacts dégagés de l'application seront dispersés sur un ensemble de trois chapitres structurés comme suit :

Le premier chapitre, intitulé "Analyse et spécification des besoins", se concentrera d'abord sur l'examen des solutions existantes et l'identification des problèmes. Ensuite, il proposera une solution.

Le deuxième chapitre, "Conception", expliquera la manière dont la solution a été conçue et utilisera des diagrammes de cas d'utilisation pour clarifier les besoins.

Le troisième chapitre, "Réalisation", fournira des informations sur les outils utilisés dans le projet et présentera quelques scénarios d'exécution.

Le rapport comprendra également un quatrième chapitre intitulé "Intégration et Déploiement". Ce chapitre se concentrera sur les aspects liés à l'intégration des différentes composantes de l'application et sur les étapes de déploiement de l'application dans un environnement opérationnel. Il examinera les processus et les outils utilisés pour garantir une intégration sans heurts des différentes parties de l'application et pour assurer son bon fonctionnement lors de son déploiement. Cette section sera essentielle pour s'assurer que l'application est prête à être utilisée par les utilisateurs finaux après la phase de développement.

Enfin, le rapport se conclura par une synthèse générale, mettant en avant les principales réalisations du travail et exposant les perspectives futures pour le développement de l'application.

# Chapitre 1

## Nom Chapitre

### 1.1 Titre

#### 1.1.1 Sous-Titre

##### 1.1.1.1 Sous-Sous-Titre

Une Figure



FIGURE 1.1 – Logo Docker.

### Isolation et Portabilité avec Docker

Docker est un outil de virtualisation légère basée sur des conteneurs, ce qui signifie que nous pouvons isoler efficacement nos applications et leurs dépendances. Cette isolation garantit que chaque composant de notre application fonctionne de manière prévisible, sans interférence avec les autres. De plus, les conteneurs Docker sont hautement portables, ce qui signifie que nous pouvons les exécuter de manière cohérente sur différents environnements, que ce soit sur nos machines de développement, nos serveurs de test ou nos serveurs de production.

Un tableau

<b>Titre</b>	S'authentifier
<b>Acteur</b>	Client
<b>Description d'enchaînement</b>	
<b>Pré-condition</b>	Client inscrit
<b>Scénario nominal</b>	
<ul style="list-style-type: none"> <li>— L'utilisateur saisit son login et password</li> <li>— Soumettre la requête d'authentification</li> <li>— Le système vérifie le compte</li> <li>— Si les informations sont correctes, on affiche la page de OTP</li> <li>— Le client saisit le code</li> <li>— Si le code est correct, le client a accès</li> </ul>	
<b>Erreur possible</b>	
Compte n'existe pas, password incorrecte , code OTP incorrect	
<b>Post-condition</b>	Client authentifié

TABLE 1.1 – Description textuelle du diagramme de cas d'utilisation s'authentifier

un bout de code

#### Manifests 1.1 – Deployment MySQL

```
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: mysql-amen-deployment
5  spec:
6    replicas: 1
7    selector:
8      matchLabels:
9        app: mysql-amen
10   template:
11     metadata:
12       labels:
13         app: mysql-amen
14     spec:
15       containers:
16         - name: mysql-amen-container
17           image: mysql:latest
18           env:
19             - name: MYSQL_ROOT_PASSWORD
20               value: root
21             - name: MYSQL_USER
22               value: amen
23             - name: MYSQL_PASSWORD
24               value: amen
25             - name: MYSQL_DATABASE
26               value: amen
```

## Conclusion

texte de conclusion



# Bibliographie

- [1] *DevOps pour les nuls*. URL : [https://www.asprom.com/application/IBM\\_2.pdf](https://www.asprom.com/application/IBM_2.pdf).