

Republic of Tunisia
Ministry of Higher Education and Scientific
Research

University of Sfax
Faculty of Sciences of Sfax



Computer Science and Communication
Department

Master's Thesis
Research Master in Computer Science

MASTER'S THESIS

Submitted to

Faculty of Sciences of Sfax
Computer Science and Communication Department

In partial fulfillment of the requirements for the degree of

RESEARCH MASTER IN COMPUTER SCIENCE

By

Author Name

THESIS TITLE

Mr. President's Name

Ms. Reviewer's Name

Mr. Supervisor's Name

President

Reviewer

Supervisor

July 2020

Author's Declaration of Originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Author Name

01-07-2025

Dedication

[YOUR TEXT GOES HERE]

Acknowledgments

[YOUR TEXT GOES HERE]

Abstract

[YOUR TEXT GOES HERE]

The thesis is written in [language] and is [number of pages in main document] pages long, including [number] chapters, [number] figures and [number] tables.

List of Abbreviations and Terms

FSS	Faculty of Sciences of Sfax
-----	-----------------------------

Table of Contents

List of Figures	9
List of Tables	10
introduction generale	1
0.1 Introduction	1
0.2 Présentation de l'organisme d'accueil	1
0.2.1 Domaine d'activité	1
0.2.2 Organisation interne	1
0.2.3 Produits et services de l'entreprise	1
0.2.4 Services informatiques et outils internes	1
1 Objectifs du mémoire et besoins de l'entreprise	2
1.1 Introduction	2
1.2 Cas et exemples	2
1.3 Les besoins fonctionnels	2
1.4 Les besoins non fonctionnels	2
2 Étude théorique et analyse bibliographique	3
2.1 Concepts généraux	4
2.1.1 Gestion des secrets	4
2.1.2 Gestion de configuration	4
2.1.3 Infrastructure as Code (IaC)	4
2.1.4 Continuous Integration et Continuous Deployment (CI/CD)	4
2.1.5 DevOps	4
2.1.6 Conteneurisation	4
2.1.7 Orchestration	4
2.1.8 Monitoring et observabilité	4
2.1.9 Stockage distribué	4
2.1.10 Gestion des logs	4
2.1.11 Reverse proxy	4
2.1.12 Gestion de pare-feu	4
2.1.13 Convention de commits	4
2.2 Les outils et les solutions	4
2.2.1 Proxmox	4
2.2.2 Terraform	4
2.2.3 Cloud-init	4

2.2.4	Vault	4
2.2.5	Consul	4
2.2.6	Ansible	4
2.2.7	Kubernetes	4
2.2.8	Kustomize	4
2.2.9	Helm	4
2.2.10	Longhorn	4
2.2.11	Grafana	4
2.2.12	Prometheus	4
2.2.13	Loki	4
2.2.14	Tempo	4
2.2.15	Argo CD	4
2.2.16	MetalLB	4
2.2.17	GitLab CI	4
2.2.18	NGINX	4
2.2.19	Commitlint	4
2.2.20	Husky	4
2.2.21	Semantic Release	4
2.3	Les projets informatiques de la société	4
2.3.1	Introduction	4
2.3.2	Oneex Front	4
2.3.3	Oneex Back	4
2.3.4	Oneex Scanner	4
2.3.5	Oneex ScanApp	4
2.3.6	Oneex CSharp	4
3	Réalisation du projet	5
3.1	Introduction	5
3.2	Objectifs du projet	5
3.3	Architecture du projet	5
3.3.1	Infrastructure as Code avec Terraform	5
3.3.2	Configuration automatique avec Ansible	5
3.3.3	GitOps avec Argo CD	5
3.3.4	Monitoring et observabilité avec Grafana, Prometheus, Loki et Tempo	5
3.3.5	Stockage distribué avec Longhorn	5
3.3.6	Gestion des secrets avec Vault	5
3.3.7	Mise en place de services internes	5
3.3.8	Sécurité et gestion de pare-feu avec pfSense	5
3.3.9	Mise en place des services de test et de staging	5
3.3.10	Mise en place de la CI/CD avec GitLab CI	5
3.3.11	Mise en place des services de production pour les clients	5

General Conclusion	6
3.4 Conclusion générale	6
3.5 Les limites du projet	6
3.6 Les améliorations possibles	6
3.7 Les enseignements personnels	6

List of Figures

List of Tables

introduction generale

0.1 Introduction

0.2 Présentation de l'organisme d'accueil

0.2.1 Domaine d'activité

0.2.2 Organisation interne

0.2.3 Produits et services de l'entreprise

0.2.4 Services informatiques et outils internes

Les services informatiques

Les outils internes

1. Objectifs du mémoire et besoins de l'entreprise

1.1 Introduction

1.2 Cas et exemples

1.3 Les besoins fonctionnels

- Automatisation des processus de provisioning.
- Gestion centralisée des configurations.
- Mise en place d'un monitoring complet.
- Sécurisation des accès aux services.

1.4 Les besoins non fonctionnels

- Haute disponibilité des services critiques.
- Sécurisation des données sensibles.
- Scalabilité pour accueillir de nouveaux projets.
- Réduction du temps moyen de déploiement.

2. Étude théorique et analyse bibliographique

2.1 Concepts généraux

2.1.1 Gestion des secrets

2.1.2 Gestion de configuration

2.1.3 Infrastructure as Code (IaC)

2.1.4 Continuous Integration et Continuous Deployment (CI/CD)

2.1.5 DevOps

2.1.6 Conteneurisation

2.1.7 Orchestration

2.1.8 Monitoring et observabilité

2.1.9 Stockage distribué

2.1.10 Gestion des logs

2.1.11 Reverse proxy

2.1.12 Gestion de pare-feu

2.1.13 Convention de commits

2.2 Les outils et les solutions

2.2.1 Proxmox

2.2.2 Terraform

2.2.3 Cloud-init

3. Réalisation du projet

3.1 Introduction

3.2 Objectifs du projet

3.3 Architecture du projet

3.3.1 Infrastructure as Code avec Terraform

3.3.2 Configuration automatique avec Ansible

3.3.3 GitOps avec Argo CD

3.3.4 Monitoring et observabilité avec Grafana, Prometheus, Loki et Tempo

3.3.5 Stockage distribué avec Longhorn

3.3.6 Gestion des secrets avec Vault

3.3.7 Mise en place de services internes

3.3.8 Sécurité et gestion de pare-feu avec pfSense

3.3.9 Mise en place des services de test et de staging

3.3.10 Mise en place de la CI/CD avec GitLab CI

3.3.11 Mise en place des services de production pour les clients

Conclusion

3.4 Conclusion générale

3.5 Les limites du projet

3.6 Les améliorations possibles

3.7 Les enseignements personnels