**Case Study – Design Document**

**“Only Flights”**



Project Members & Student Numbers:

* Kaloyan Andreev (4408020)
* Francisco Marcó (4467752)
* Sava Vasilev (4663438)

Elaboration Date: 10/10/2022

Case Study - Group 10

Table of content

[1. Design details 2](#_Toc123657777)

[2. Technical specifications 4](#_Toc123657778)

[2.1. EC2 instances 4](#_Toc123657779)

[2.2. RDS instance 4](#_Toc123657780)

[2.3. ECS Task definition 5](#_Toc123657781)

[2.4. Application Load Balancer 6](#_Toc123657782)

[3. Resource providers 7](#_Toc123657783)

[4. Future work 8](#_Toc123657784)

[5. C4 Model diagram 9](#_Toc123657785)

[6.1. System context 9](#_Toc123657786)

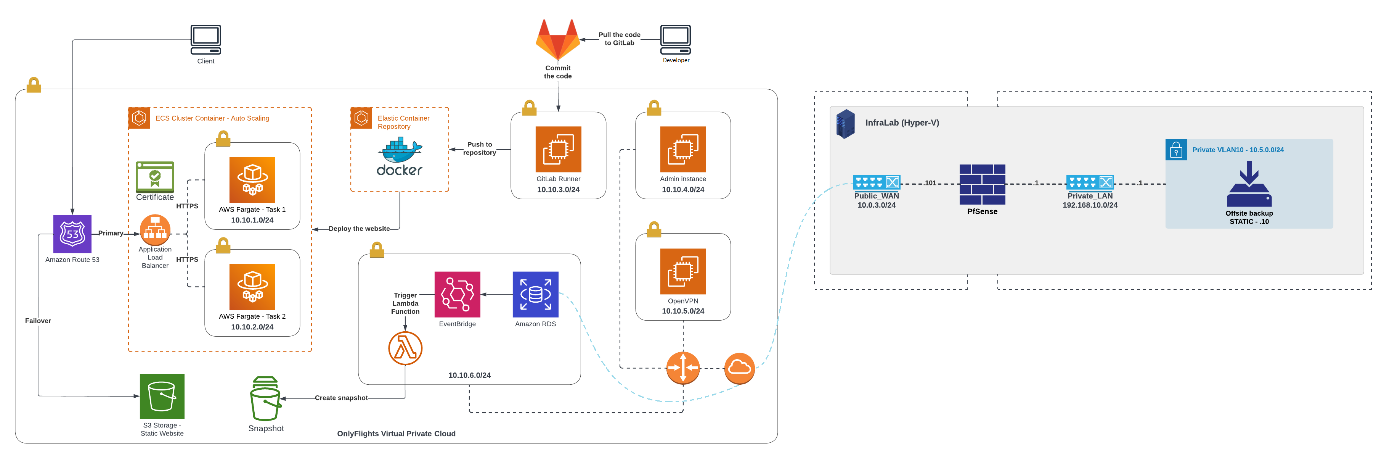
[6.2. Container diagram 9](#_Toc123657787)

[6. UML Diagram 10](#_Toc123657788)

[7. ERD Diagram 11](#_Toc123657789)

# Design details

The project named “Only Flights” will be situated in two environments – the major part will be in Amazon Web Services where most of the stuff will take place, while the off-site backup will be hosted on a separate server provided by Fontys Hogeschool.

The cloud environment will contain one Virtual Private Cloud, where all services will be situated. There will be separate subnet for each server (ec2) and RDS instance. To deploy the “Only Flights website” there will be an Elastic Container Service – ECS, where a one service with two Fargate tasks will be run. Again, each Fargate task will be in its own subnet. To make the website easily accessible for the end-users, a new domain name will be registered as well as a new certificate for authorization of it. To balance the traffic to the website, an Application Load Balancer will be setup in the ECS cluster. In that way, the traffic could be also redirected from port 80 to port 443 which will enhance the security. In case of a failover of the website, there will be a record in the Route53 hosted zone which will redirect all users to a static webpage hosted in S3 bucket. For easier deployment of new web features, there will be GitLab CI/CD pipelines implemented. There will be few tests that will be performed before the new changes are push to Elastic Container Repository – ECR and then to the GitLab Runner and ECS cluster.

Several lambda functions will be also created that will automate the processes of starting and stopping instances in the cloud. In addition, there will be another lambda that will take snapshots of the RDS every day at certain time so, the data is protected.

For monitoring, it is plan two tools to be used. In AWS, there will be configured Prometheus on the three EC2 machines – Admin (the main one), OpenVPN and GitLab Runner. A few Grafana tables will be created that will illustrate the most important processes. In our on-premises server, there will be Zabbix configured that will take care of the off-site backup instance. CloudWatch dashboards will be made to monitor several metrics of some of the most active servers in AWS as well.

On-premises server will contain a PfSense Firewall which will be attached to two virtual switches – one to access the public environment (WAN) and another to access the private environment (LAN). In the LAN side there will be a private virtual LAN, where the off-site backup will be placed. There will be also a Zabbix configured to monitor all instances inside the server.

Both infrastructures will be connected to each other by using OpenVPN connection. In that way, the connection can be made to turn on only when it is time for backup of the RDS instance. In this case the costs will be reduced significantly, and the security will be brought up on next level.

As it was mentioned earlier in the document, there will be seven subnets. The first one of them will contains the RDS SQL Server Database, where all information about the company’s customers such as personal and account details will be stored. The database will be configured in way that it will be available in multiple availability zones and the data will be backed up once per week and daily snapshots will be created. The backups and snapshots will be saved for at least seven days before they will be removed. A read-replica of the database will be created as well.

The other subnets will contain the OpenVPN server, which will be turn on/off only when it is time for backup of the RDS, the GitLab Runner – needed for deployment of the CI/CD pipelines and the Admin instance for monitoring all machines and accessing the database data.

# Technical specifications

## EC2 instances

|  |  |  |  |
| --- | --- | --- | --- |
|  | **GitLab Runner** | **Admin** | **OpenVPN** |
| Amazon machine image (ami) | ami-08f4df18aa6f249eb | ami-08f4df18aa6f249eb | ami-08f4df18aa6f249eb |
| Instance type | t2.micro | t2.micro | t2.micro |
| VPC | 10.11.0.0/16 | 10.11.0.0/16 | 10.11.0.0/16 |
| Subnet | 10.11.3.0/24 | 10.11.4.0/24 | 10.11.5.0/24 |
| IP address | 10.11.3.10 | 10.11.4.10 | 10.11.5.10 |
| Public SSH key | OnlyFlights\_GitRunner\_key\_pair | OnlyFlights\_Admin\_key\_pair | OnlyFlights\_OpenVPN\_key\_pair |
| Security group | gitlab-security-group | admin-security-group | openvpn-security-group |
| Tags | Name = "GitLab Runner"  Importance = "Production" | Name = "Admin"  Importance = "Monitoring" | Name = "OpenVPN"  Importance = "Backup" |

## RDS instance

|  |  |
| --- | --- |
|  | **RDS** |
| Identifier | onlyflights-database |
| Allocated storage | 50 GB |
| Engine | mysql |
| Engine version | 8.0 |
| Instance class | db.t3.medium |
| Storage encrypted | True |
| Security group | rds-security-group |
| Subnet group | rds\_subnet\_group |
| Backup retention period | 1 |
| Backup window | "04:00-06:00" |
| Maintenance window | SUN:08:00-SUN:10:00 |
| Multi-availability zone | True |
| Publicly accessible | True |
| Tags | Importance = "Database |

## ECS Task definition

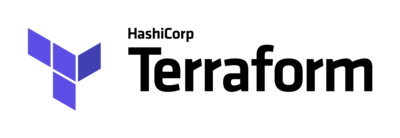
|  |  |
| --- | --- |
|  | **Task definition** |
| Name | flask\_webapp |
| Image | 153658864904.dkr.ecr.eu-central-1.amazonaws.com/onlyflgihts-ecr:latest |
| CPU | 1024 |
| Memory | 3072 |
| Size in gib | 21 |
| CPU architecture | X86\_64 |
| Operating system family | Linux |
| Task role ARN | arn:aws:iam::153658864904:role/ecsTaskExecutionRole |
| Execution role ARN | arn:aws:iam::153658864904:role/ecsTaskExecutionRole |
| Requires compatibilities | Fargate |
| Network mode | awsvpc |
| Tags | Key = "ecs:taskDefinition:createdFrom"  Value = "ecs-console-v2" |

## Application Load Balancer

|  |  |  |
| --- | --- | --- |
|  | **onlyflights-port80-listeners** | **onlyflights-port443-listeners** |
| Name of ALB | onlyflights-alb | onlyflights-alb |
| SSL Policy |  | ELBSecurityPolicy-2016-08 |
| Port | 80 | 443 |
| Protocol | HTTP | HTTPS |
| Type   * Port * Protocol * Status code | Redirect   * 443 * HTTPS * HTTP\_301 | Forward   * onlyflights-ip-target-group |

# Resource providers

During the development of the current project, resources from the following providers were used.

* ***Amazon Web Services***
* ***Terraform***
* ***Ansible***
* ***Flask***
* ***Zabbix***

******

* ***Microsoft SQL Server***
* ***Docker***

# Future work

There is a plan in the future to distribute the whole system – cloud infrastructure, website and automation procedures, to as many as possible aircraft companies so, they could save time and financial resources and to have one secure and optimized platform that will be updated and maintained regularly.

# C4 Model diagram

## System context

Diagrama

Descripción generada automáticamente

## Diagrama Descripción generada automáticamenteContainer diagram

# UML Diagram

# Diagram Description automatically generatedERD Diagram