**Technical manual**

**“Only Flights”**

A picture containing text, clipart

Description automatically generated

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. Configuration of the Admin’s EC2 and the Webserver 2](#_Toc118452107)

[a. Purpose of the using the shell scripts 2](#_Toc118452108)

[b. Sequence of execution 2](#_Toc118452109)

[c. Precise description of the embedded functions inside the codes 2](#_Toc118452110)

[d. Troubles while using the shell scripts ? 3](#_Toc118452111)

[2. Setup database for working 4](#_Toc118452112)

[a. Purpose of the using the shell scripts 4](#_Toc118452113)

[b. Requirement before execution 4](#_Toc118452114)

[c. Precise description of the embedded functions inside the codes 4](#_Toc118452115)

[d. Troubles while using the shell scripts ? 4](#_Toc118452116)

# Configuration of the Admin’s EC2 and the Webserver

## Purpose of the using the shell scripts

To set up these AWS instances easily and simply, two bash scripts are developed and are waiting to be put into action. The only thing that the user must be aware of, is the sequence of their execution.

## Sequence of execution

* Firstly, the user must execute ‘pk\_&\_flask.sh’
* Secondly, the user must execute ‘ansible\_playbook.sh’

## Precise description of the embedded functions inside the codes

‘pk\_&\_flask.sh’

The purpose of the following code is to transfer the private key and the folder, that will contain the flask application, from the user’s local machine directly onto admin’s server. This will be done by the command ‘scp’ embedded into the shell script. The only task, that customer have to do while this process, is to enter the correct full name (including the type of the file – ‘.pem’) of the private key when he or she is asked by the console. Once everything is done successful, the user will be notified by a message. In case of an error (most likely duo to connection time out), another message will appear on the screen that will outline the exact mistake.

**Note:** *The file can be found in the following directory*:

* [Bash Scripts/pk\_&\_flask.sh · main · Andreev,Kaloyan K.K. / Semester 3 - Case Study project · GitLab (fhict.nl)](https://git.fhict.nl/I476236/semester-3-case-study-project/-/blob/main/Bash%20Scripts/pk_&_flask.sh)

‘ansible\_playbook.sh’

By running this shell script on the admin’s instance, the server will execute number of commands that will prepare it for work in a couple of minutes. All commands are divided into several functions for easier configuration and will be performed one after another.

The first four tasks will update all packages so, the machine will be able to install Ansible and to operate properly.

The next two functions are related to the configuration of the forementioned automation tool. The user will be asked to input the IP address of the server where the website will be hosted so, the system will adjust the inventory file of Ansible.

Then, it is time the Playbook to be generated. The main goal of it, is to install and setup apache2 server onto the webserver instance and to make it able to host the pre-made flask application. There will be several directories that have to be created so, everything to work properly.

The last few tasks that the script has to perform are to transfer the flask folder that will be situated in the home directory to specially designed one, that will be made by the Playbook. Afterwards, the user will be asked to input the name of the Private key so, the system to be able to run the Playbook. The final assignment is to activate the Flask website by executing a certain command.

When all functions are done successfully, the user will be notified by a message displayed on the console.

**Note:** *The file can be found in the following directory*:

* [Bash Scripts/ansible\_playbook.sh · main · Andreev,Kaloyan K.K. / Semester 3 - Case Study project · GitLab (fhict.nl)](https://git.fhict.nl/I476236/semester-3-case-study-project/-/blob/main/Bash%20Scripts/ansible_playbook.sh)

## Troubles while using the shell scripts ?

In case of technical failure please report it immediately to one of the following developers so it could be solved as soon as possible

***Contact info:***

* [k.andreev@student.fontys.com](mailto:k.andreev@student.fontys.com)
* [sava.vasilev@student.fontys.com](mailto:sava.vasilev@student.fontys.com)
* [f.marco@student.fontys.nl](mailto:f.marco@student.fontys.nl)

# Setup database for working

## Purpose of the using the shell scripts

To configure the tables inside the database easily and simple, the following shell script and Python code are developed. The user must be aware of the setup of his or hers system, before start executing the scripts.

## Requirement before execution

* The EC2 instance must have connection to the RDS instance
* MySQL must be installed
  + Use the following guide to install and configure MySQL on Amazon Linux: [How to Install MySQL on CentOS 7: A Step-by-Step Guide (hostinger.com)](https://www.hostinger.com/tutorials/how-to-install-mysql-on-centos-7)
* Python version must be at least 3.9
* Pip must be installed
  + Use the following guide to install Pip: [How to Install Pip on CentOS 7 | Linuxize](https://linuxize.com/post/how-to-install-pip-on-centos-7/)
* Python library ‘mysql.connector’ must be installed
  + Use the following guide to install the library: [MySQL :: MySQL Connector/Python Developer Guide :: 4.2 Installing Connector/Python from a Binary Distribution](https://dev.mysql.com/doc/connector-python/en/connector-python-installation-binary.html)

## Precise description of the embedded functions inside the codes

‘database\_tables\_creation.sh’

By executing the following script from the admin’s instance, all database tables, that shown on the ERD diagram, will be created automatically. The shell will start a python code, which aims to establish connection to the RDS instance, situated in the cloud, and by running several queries, to generate the forementioned MySQL tables.

**Note:** *The file can be found in the following directory*:

* [Bash Scripts/database\_tables\_creation.sh · main · Andreev,Kaloyan K.K. / Semester 3 - Case Study project · GitLab (fhict.nl)](https://git.fhict.nl/I476236/semester-3-case-study-project/-/blob/main/Bash%20Scripts/database_tables_creation.sh)
* [Python scrpits for database/OnlyFlights-Database-tables.py · main · Andreev,Kaloyan K.K. / Semester 3 - Case Study project · GitLab (fhict.nl)](https://git.fhict.nl/I476236/semester-3-case-study-project/-/blob/main/Python%20scrpits%20for%20database/OnlyFlights-Database-tables.py)

## Troubles while using the shell scripts ?

In case of technical failure please report it immediately to one of the following developers so it could be solved as soon as possible

***Contact info:***

* [k.andreev@student.fontys.com](mailto:k.andreev@student.fontys.com)
* [sava.vasilev@student.fontys.com](mailto:sava.vasilev@student.fontys.com)
* [f.marco@student.fontys.nl](mailto:f.marco@student.fontys.nl)