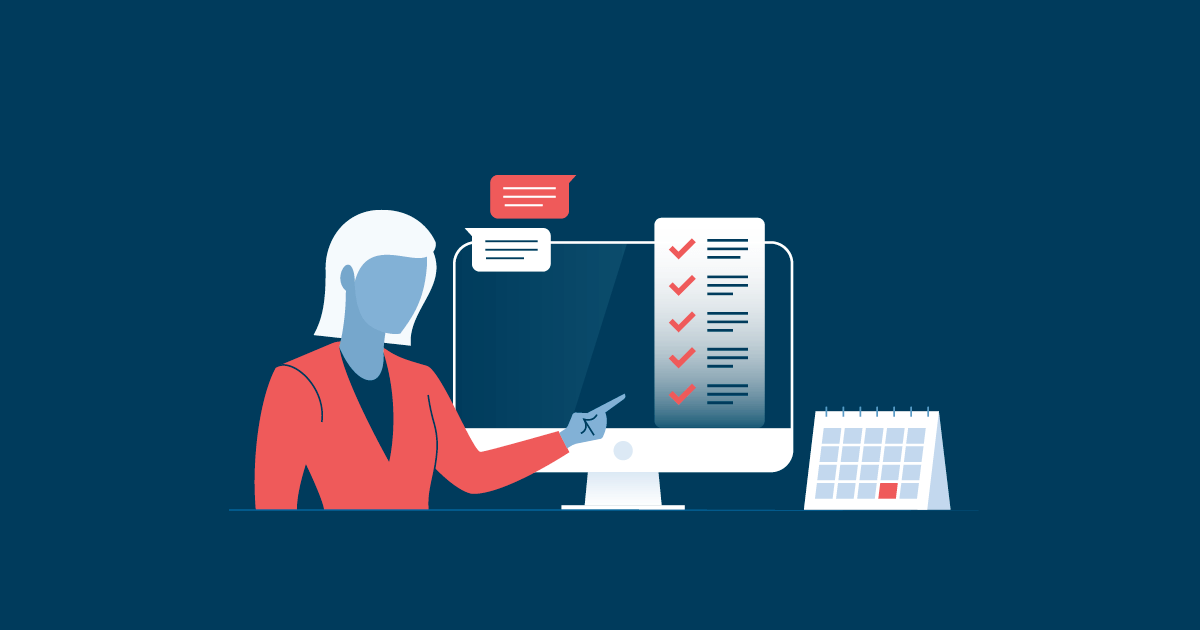
**Case Study 2 – Design Document**

**Media storage service**



Project Members & Student Numbers:

1. Francisco Marcó (4467752)
2. Sava Vasilev (4663438)
3. Kaloyan Andreev (4408020)

Elaboration Date: 20/05/2022

Case Study 2 - Group 10

Table of content

[1. Detailed design 2](#_Toc105927827)

[2. Work estimates 5](#_Toc105927828)

[3. Roll-out plan 6](#_Toc105927829)

[4. Related work 7](#_Toc105927830)

[5. Future work 8](#_Toc105927831)

[6. C4 Model diagram 9](#_Toc105927832)

[6.1. System context 9](#_Toc105927833)

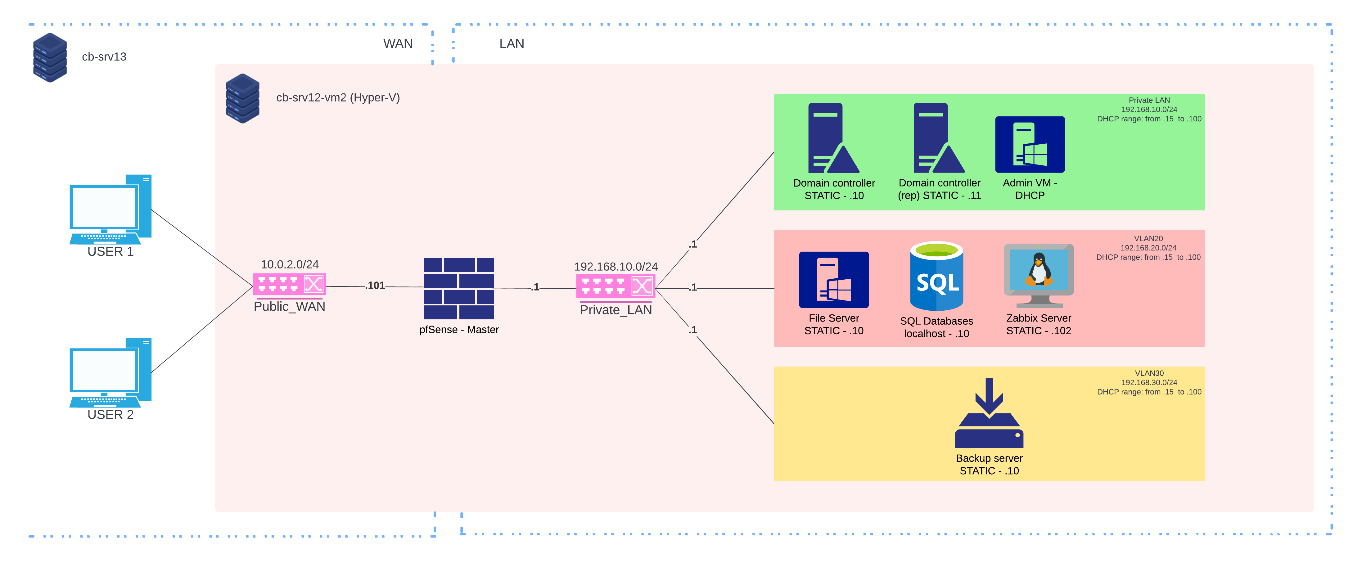
[6.2. Container diagram 9](#_Toc105927834)

[7. UML diagram 10](#_Toc105927835)

[8. ERD diagram 11](#_Toc105927836)

[9. Use Case Diagram 12](#_Toc105927837)

# Detailed design

The project is hosted on a Fontys Server and we are creating our infrastructure on it. It will contain a Firewall, an Active Directory, a Client Server and an admin user machine. For the firewall we chose pfSense software and for the other machines Windows Server 2019. The Client Server is the one hosting the web-based access application for clients and there will also be stored all the client data.

For this project, a database is created. It holds important data both for the clients and for the company’s employees. A table for the company employees contains user login credentials and personal data such as ID number, Name, DOB, Phone number, email, address. Similar information holds the clients’ table. It has client login credentials plus ID, name, DOB, email, phone number and storage directory path.

For this project, a new application is developed specially designed for our needs. It contains a simple login page and a simplified interface of the users directory with all of the files they have access to. It will provide a secure access to the server.

The whole process takes no more than a usual upload and is dependent on the client’s internet speed. Space-wise our product is as efficient as it gets by utilizing dynamic memory and different access levels. We expect the Client Server to be the most accessed part of our infrastructure with the highest load. For that reason, we have made sure the connection will manage to hold the throughput of information going in and out of the server.

During the use of our product errors could happen and we tried to outline most of them. Apart from the more obvious network error messages the first and most basic one specific to us is wrong credentials. Corrupted file is another error state which we hope to avoid. Other error states include:

* **File does not exist**
* **File could not be uploaded**
* **File already exists**
* **Synchronisation error**
* **Server down for maintenance**
* **Unknown error we are working on a fix**

As we aim to keep our servers healthy, we have a sophisticated logging and monitoring policy. With the help of Zabbix all logs, and state of the machines is monitored through the admin user machine. It has full control over the servers and keeps a detailed log on events that are happening on it. By taking these extra measures we make sure to limit the amount of server errors to a minimum and improve the user experience.

For the same reason we also perform regular security checks and have a detailed security analysis on our system. To avoid different threats different measures are being taken. The most basic one is a regular backup which includes a copy off-site thus making sure there is always a state to revert to.

As this is a multi-user platform there are some privacy concerns and we are constantly making sure our privacy policy is up to date. With that we also regularly test our system and make sure each client has the promised privacy. Their data is not only private from outside factors but also from other users and employees. Unless we have proof that the privacy policy has been breached the client remains the only one with access to their files.

As this is a web-based application it needs to be made compatible with each browser. For that reason, we are keeping it simpler in terms of user interface. By doing so we make the platform accessible for more users which might connect using other browsers.

To test our system, we have two user dummies which are used for different types of tests on the system including but not limited to privacy testing, load-limit testing, general user experience testing. For the user experience we utilize the power of crow-sourcing and have random volunteers to test it. For the other two the tests are performed by security experts working for the company.

# Work estimates

The work on the project is divisible in a couple of groups. Documentation – most of which will be finished by the second week but will constantly be kept up to date throughout the project, Web-application – a major part of the project which will start development in the third week (Sprint 2) and finally security analysis – the last major part of the project which will start in week 4, during the development of the web application. By the end of Sprint 3 (week 6) we estimate that the documentation will be ready, the application will be up and running and there will be no major security risks left unhandled.

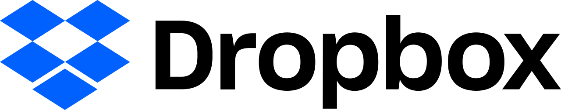
# Roll-out plan

The application will be online by week 4 and will be accessible to the public by week 6. In the meantime, tests need to be performed on the security and the infrastructure product. After week 5 and week 6 the new updated of the application will be uploaded and implemented following more tests to ensure the changes have fixed the issues and to find other potential problems.

# Related work

There is another project online similar to this on which are worth looking into:

* Logo

  Description automatically generated with medium confidenceGoogle Drive
* Dropbox
* MEGA
* Logo, company name

  Description automatically generatedOneDrive
* SpiderOak One

# Future work

Our plans include creating a cross-platform application. Goal is to make this application accessible for multiple devices and to make it possible to upload files from anywhere securely if there is internet. We also plan to have backup server for when the amount of user grows and lastly, we need to implement RAID for our storage system. This will further prevent data loss.

# C4 Model diagram

## System context

Diagrama

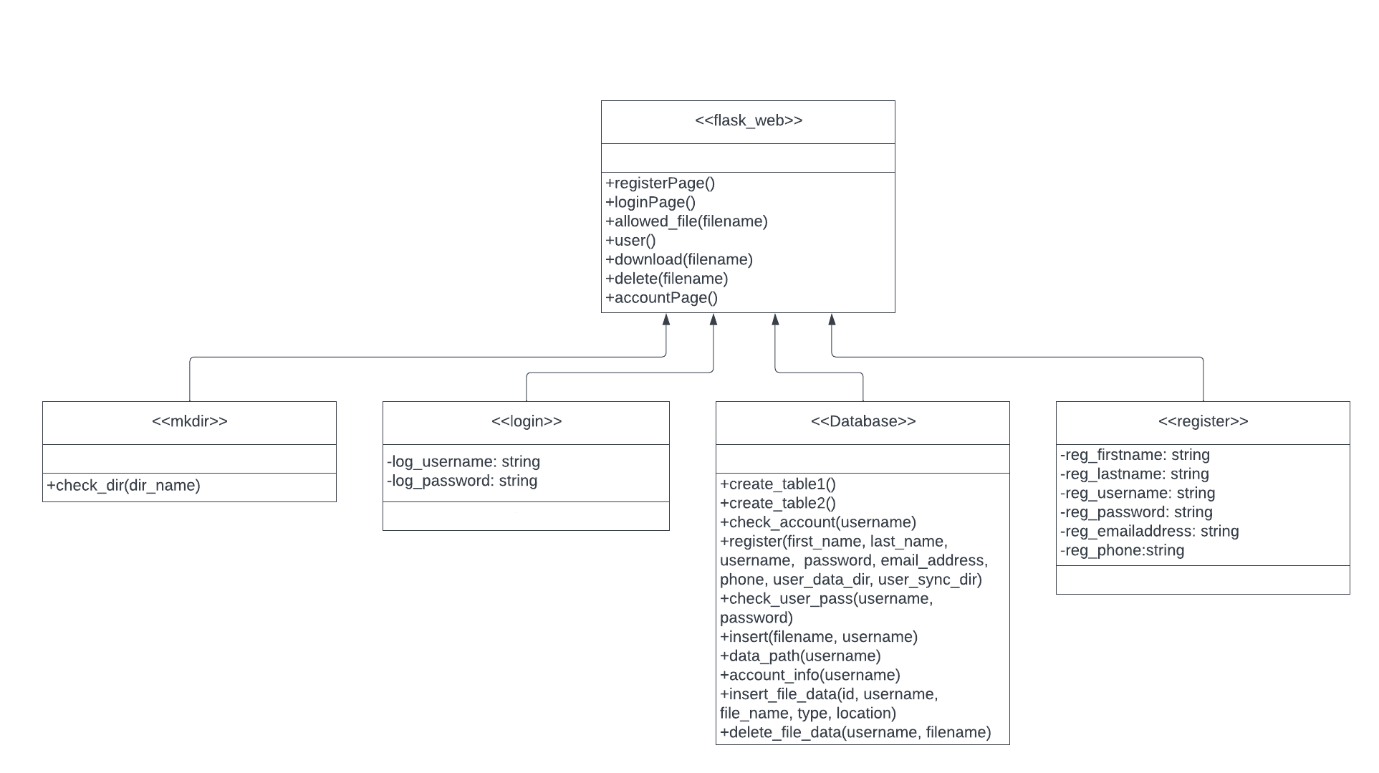
Descripción generada automáticamente

## Container diagram

Diagram

Description automatically generated

# UML diagram



# ERD diagram

A screenshot of a computer

Description automatically generated with low confidence

# Use Case Diagram

Diagram

Description automatically generated