



Elaborato [en] – State Exam 2020/2021

Design of the network infrastructure, database and WEB pages for the
creation and use of a cloud storage

May 31st, 2021

Index

Analysis.....	4
1. Network infrastructure.....	4
1.a Data Center	4
Security and QoS.....	4
LAN design and addressing.....	5
Network operation	6
VPN addressing.....	6
Specification's equipment and connectors.....	6
1.b Offices.....	7
Security and continuity of service	7
LAN, WLAN, and addressing design	7
Equipment and connectors specifications.....	8
1.c Network scheme.....	9
Detail of the Amsterdam structure.....	10
2. Informative System.....	10
2.a Database	10
Conceptual Diagram.....	11
Reading rules.....	11
E-R scheme	12
Logical Scheme.....	12
Users.....	12
Credentials	12
Folders	13
File	13
Access Folders.....	13
Share File	13
Folders options	13

File options	14
Entity description and relationships	14
Database creation and management	14
Creating tables.....	14
Inserting and editing data	17
Data visualization.....	19
2.b Website.....	20
Site map	20
WEB pages.....	21
Home.....	21
Home cloud.....	21
Shared with me.....	22
Preferred.....	23
Bin.....	23
Registration	24
Login	24
Edit Profile.....	25
PHP code	25
Registration	26
Login	28
Edit Profile.....	29
Upload file	31
3. Project management.....	32
3.a Work Breakdown Structure	33
3.b Gantt chart	34
4. Conclusions.....	35

Analysis

The MicroDrive company, already structured in North America, needs to expand its data center network also in Europe to begin providing its services as required by EU regulations. The company needs to create a network and an information system for the management and provision of various services including cloud storage.

The company requires the initial creation of three structures for the placement of servers in the cities of Milan, Frankfurt and Amsterdam: these three structures must be expandable and interoperable with the future structures that will be built. The main office will be in Frankfurt, while the legal and administrative office, including the personnel offices, will be in Amsterdam. The three offices will have to manage South, Central and Northern Europe respectively, directing data traffic as needed. The three offices must be interconnected, the databases, even if located in different structures, must be able to see and communicate with each other.

The company also requires various features for the web application, including the ability for customers to upload files and folders and consequently be able to share them with other users of the company.

As for employees, they will have access to a local system that cannot be reached from outside the company except through the use of VPN. The local server will allow the management of the various administrator functions by interacting with the different servers in the different offices.

1. Network infrastructure

1.a Data Center

To proceed with the construction of the required structures, it is necessary to create a network infrastructure capable of supporting the huge amount of data that must continuously be provided to customers.

The network present in the data centers is scaled according to the size of the same: the Frankfurt facility will need a more performing infrastructure than the other two.

Within the German structure it is estimated the presence of about 15,000 servers, in the Dutch and Italian ones about 10,000 between service servers and data servers.

Considering the difference in capacity of the structures, fiber optic (FO) bindings to Internet Service Providers (ISPs) will have to guarantee different speeds: at least five connections plus five backup connections are required in Frankfurt, at least three connections and three of backup. The connections are intended with a symmetrical 25Gbps bitrate; backup connections must be provided by different operators and have a separate packet path.

Security and QoS

Security in data centers is ensured by separating servers accessible from the Internet (WEB Server) from those containing data (FTP Server) thanks to the use of a DMZ. Then there are Firewall, IPS and IDS with the function of blocking unauthorized traffic and prevention and detection of network intrusions.

To ensure optimal quality of service, the structures must be connected to each other via VPN using site-to-site connections with the BG/MPLS protocol which allows, thanks to agreements with ISPs, the secure

and priority transmission of traffic through the nodes of the network, this in order to exchange data between the different structures.

Furthermore, to ensure continuity of service and minimize risks, the structure must be equipped with anti-intrusion and fire-fighting systems: access must be limited to authorized persons only to avoid data being stolen or tampered with and equipment being damaged. The fire extinguishing system must be strictly CO2 to avoid short circuits in the equipment.

In addition, all FTP servers must have a RAID 6 configuration to avoid data loss in the event of a failure in one or more disks per server.

The backups are carried out continuously, by cloning the data of the structure on the servers of another, doing so even in the event of total destruction of the structure and total loss of data on site it will be possible to continue providing the service through the servers placed in the other two structures.

To guarantee service even in the event of local blackouts, the data centers need UPS and emergency generators capable of keeping the entire structure active for long periods.

Finally, an essential component of the data center is the ventilation and air conditioning system, which must be able to keep the rooms at a temperature between 20 and 22 ° C.

LAN design and addressing

The rack configuration provides for 40 servers and 2 switches for each, these are then connected in groups of 10 to higher level switches and subsequently the level 2 switches merge into two pairs of central star-point switches. The redundancy of the devices is aimed at the continuity of the service even in case of malfunction of the single device that can be hot swapped.

The local networks to be created for each facility are two different: in the first, accessible from the outside and therefore located in a demilitarized zone (DMZ), the WEB servers will be located for accessing the available services; in the second, the data servers will be located, so that they are not reachable from the outside.

The creation of the two networks will take place in a logical way, not by laying independent devices but by using VLANs capable of isolating the traffic of the two networks. Given the centrality of the structures in question, it will be necessary to prepare ready-made equipment for expanding the network, minimizing inefficiencies.

The LANs will have private class A addresses (10.0.0.0 / 8) divided into compatible subnets to allow communication between the three structures.

Facility	Function	Address	Submask	Gateway
Frankfurt	Internal network	10.0.0.0	255.224.0.0	10.0.0.1
Frankfurt	DMZ network	10.32.0.0	255.224.0.0	10.32.0.1
Amsterdam	Internal network	10.64.0.0	255.224.0.0	10.64.0.1
Amsterdam	DMZ network	10.96.0.0	255.224.0.0	10.96.0.1
Milan	Internal network	10.128.0.0	255.224.0.0	10.128.0.1
Milan	DMZ network	10.160.0.0	255.224.0.0	10.160.0.1

The use of class A addresses allows the management of huge quantities of devices as well as considerable flexibility in case of future expansions. Given the complexity of the network it is not possible to use the DHCP protocol, the addresses must be static.

As regards access to WEB servers from the Internet, the NAT protocol is used in static mode for all devices located in the DMZ. It is therefore necessary to purchase a class B address class. These addresses must be manually assigned and cannot vary.

Network operation

The network routes traffic arriving into the DMZ except for traffic from VPNs that goes to the Intranet. In the DMZ the servers are able to provide all the required services, starting from the WEB server configured on port 443 to provide HTTPS service with TLS/SSL encryption. When the WEB server is requested to access the data, it will connect with the Intranet servers on port 20 via FTP protocol for file transfer.

Users outside the network cannot access the Intranet under any circumstances, the only ones authorized to do so are employees based in Amsterdam. These have access to the Intranets of all the facilities also through the use of VPNs, which allow site-to-site connection, making the Frankfurt and Milan Intranets appear as if directly connected on site.

VPN addressing

Facility	Address	Submask	Gateway A	Gateway B
Amsterdam - Frankfurt	172.16.0.0	255,255. 192.0	172.16.0.1	172.16.0.2
Frankfurt - Milan	172.16.64.0	255.255. 192.0	172.16.64.1	172.16.64.2
Milan - Amsterdam	172.16.128.0	255.255. 192.0	172.16.128.1	172.16.128.2

Specification's equipment and connectors

To build data centers, equipment in the order of thousands of servers, hundreds of rack switches, dozens of second-level switches and some third-level switches are required. You also need routers for connections to the internet and VPNs.

Servers must be equipped with dual Intel Xeon processors and 512GB of RAM to provide the necessary computing power; the data servers must also host 10 SSDs of 5TB each for a total of 50TB per server. Two 10Gbps rj-45 ports are also required.

The switches in the racks must have 48 10Gbps rj-45 ports for connection with servers and 4 25Gbps SFP28 uplink ports for fiber optic connection with higher-level switches.

Tier two switches must have 48 25Gbps SFP28 ports and 8 100Gbps SFP28 uplink ports. The centralizing switches, with the star center function, must instead have 32 100Gbps SFP28 ports and 4 400Gbps uplinks.

Routers are then required which must have 20 25Gbps SFP28 ports for connection with external network devices and switches.

The cables necessary for connecting the servers with the switches present in the racks must be Cat7 FTP, in order to be able to have connections at 10Gbps and avoid interference and noise given by the multitude of cables present.

For fiber optic connections, multimode cables are preferred, in particular LC UPC cables must be used between the switches in the racks and those of the second level, between the second level switches and those centralizing MTP-12 cables of 12 fibers each. OM4-4xLC UPC cables are required between the centralizing switches and routers, capable of centralizing the signal from 4 25Gbps SFP28 ports into a 100Gbps SFP28.

The network also requires Firewall, IPS and IDS placed between the centralizing router and switch.

1.b Offices

The company's offices are also located in the registered and administrative headquarters of MicroDrive in Amsterdam, these are spread over 3 floors and include about 120 workstations.

There is also a mail server to locally manage e-mail correspondence with configuration of the SMTP protocols on port 25 and the POP3 service on port 110.

Since the offices and the data center share the same structure, the equipment is also partially shared. The connection to the Internet takes place via a dedicated symmetrical 10Gbps connection, while access to the data center network takes place through the main switches of the data center itself.

Security and continuity of service

Access to the company's internal network and devices must only take place through the use of credentials specifically created by the network administrator. These rely on an AAA server for authentication and verification of permissions, as well as for saving activity logs.

The AAA server is also used for access via WLAN, security is guaranteed by the corporate WPA2 protocol using the AES algorithm.

Given the MicroDrive company policy, which allows agile work for its employees, it is necessary to create an infrastructure to which it is possible to connect remotely through the use of a VPN in Remote Access mode. For the realization of the same, the use of the SSL/TLS protocol suite is preferred, which allows you to guarantee security in the connection as well as authentication. To make these connections you need a Network Access Server (NAS) which, collaborating with the AAA server, allows access to the network.

In the offices, since continuity of service is essential, UPSs are positioned to supply essential workstations and devices with electricity in the event of a blackout. For longer periods of power failure, the data center's emergency generators are activated, supplying electricity to the offices as well.

Backups are performed automatically every hour and the data is crammed simultaneously in both Milan and Frankfurt. The backup is incremental and includes all wired devices in the offices.

Also, in the offices there are CO₂ fire and intrusion systems.

LAN, WLAN, and addressing design

The network sees the star center of the offices placed together with the servers, from there the central switches are connected by fiber optic cables to the floor switches. On the first and second floor there are 50 fixed workstations each, on the third floor there are 20 offices with relative workstations.

Between devices such as VoIP phones, printers, access points and more, there are around 300 wired devices in total.

Structured cabling must take place following ISO 11801, placing the wall sockets for wired devices at a maximum of 90 meters from the floor switches and the end devices at less than 5 meters from the wall socket. As far as the optical fibers are concerned, there are no problems in the structure given the maximum possible distance of about 2000m.

Five access points per floor are used for configuring the WLAN, partially overlapping and with the same SSID. Particular attention is paid to the choice of radio channels to be used, avoiding that these interfere with each other.

In order to manage all the devices, it is necessary to use VLANs, configuring different addresses between the wired stations and the WLANs, also separating the floors. Private class C addresses are used and the PAT protocol is used to access the Internet, capable of navigating more than 65,000 devices with a single public address using the socket.

Floor	Function	Address	Submask	Gateway
first	LAN	192.168.0.0	255.255.255.0	192.168.0.1
first	WLAN	192.168.10.0	255.255.255.0	192.168.10.1
second	LAN	192.168.1.0	255.255.255.0	192.168.1.1
second	WLAN	192.168.11.0	255.255.255.0	192.168.11.1
third	LAN	192.168.2.0	255.255.255.0	192.168.2.1
third	WLAN	192.168.12.0	255.255.255.0	192.168.12.1

Each floor has two separate networks, but these are able to interact with each other. This, in addition to facilitating management, also allows the connection of a large number of devices while maintaining class C addresses. Given the versatility and dynamism of the network, most of the addresses are assigned using the DHCP protocol. The devices turned on 24 hours a day, such as printers and NAS servers, have a static configuration.

Equipment and connectors specifications

In the offices, adequate equipment is required to provide internet to fixed and mobile workstations. Two switches per floor are required to ensure correct configuration, capable of providing internet to wired devices and access points. To do this you need 2 switches per floor with 48 1Gbps rj-45 ports and 4 10Gbps SFP + uplinks, these will be coupled through two 10Gbps Twinax cables: with this configuration these switches are able to work as a single one.

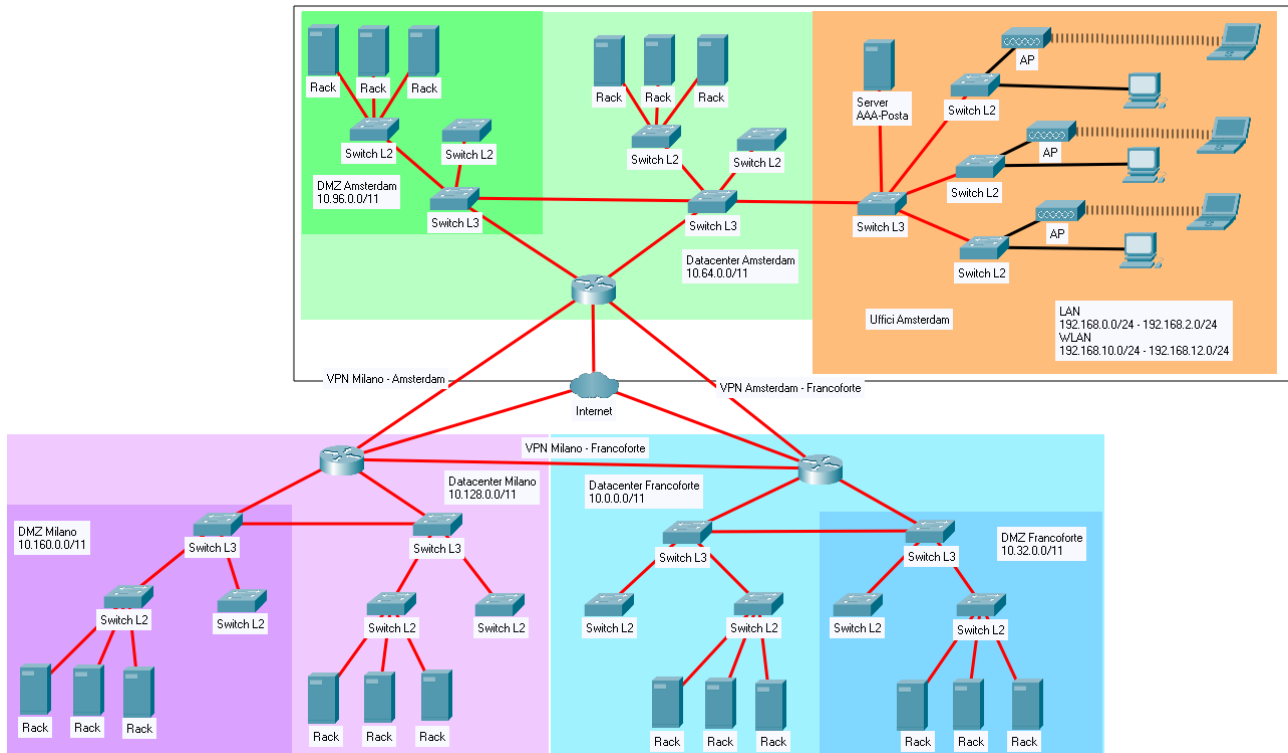
The wall sockets must be placed within 90m from the switch, the distance must be the same also for the Access Points. For this reason, the devices will be placed centrally in the structure.

The APs must be connected with double Cat7 UTP cable capable of providing connections at 10Gbps and must be placed partially overlapping to ensure signal continuity. The Access Points must support the AX protocol and therefore the 2.4 and 5 GHz frequencies.

The centralizing switch, which acts as the star center, must have 20 SFP + ports and 4 25Gbps SFP28 uplinks. It is connected to the data center switches and to the access switches positioned on the floors by means of multimode fiber optic cables with LC UPC connectors.

A server connected directly to the centralizing switch via fiber optic cable is used to manage the Mail and DHCP services.

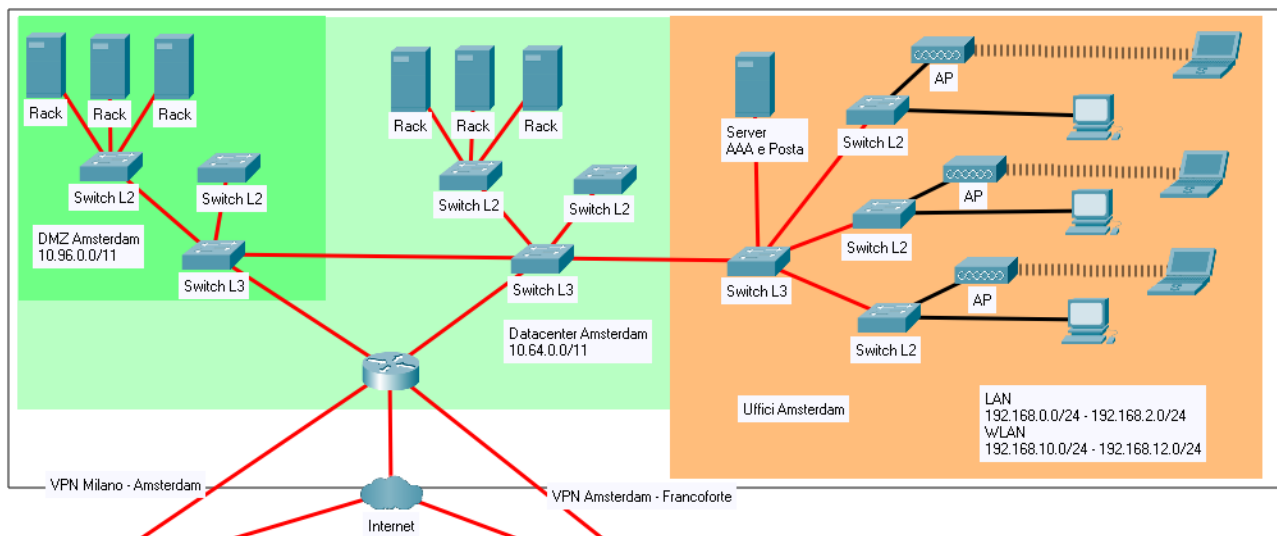
1.c Network scheme



The networks present in the structures of Amsterdam (data centers in green and offices in orange), Milan (purple) and Frankfurt (blue) are identified in the diagram. The diagram includes the centralizing and second level switches (for access to the offices), the servers and the relative switches are grouped together under the representation of the racks.

There are also direct links between the structures, representing site-to-site VPNs, in addition to connections with the Internet.

Detail of the Amsterdam structure



2. Informative System

The information system requires users to register via a dedicated page on the MicroDrive website: the information requested is an E-mail, Name, Surname, Date of Birth, Gender, Country of residence and a username to be used as a MicroDrive. You will then need to use a password that meets specific security criteria (at least 8 characters, one lowercase, one uppercase, one number and one special character). Instead, it is optional to enter a phone number for account recovery.

Once registration is complete, it will be necessary to confirm it by using the link sent to the recovery email box provided by the user. After verifying the account, it will be possible to access the MicroDrive services.

If you have already registered, you will need to log in.

Once logged in to the cloud service, you can upload files and folders using the drag and drop function or using the appropriate button, which opens a window for browsing the local directories. Uploaded files and folders must have an adequate display by means of icons that represent the different types of files, based on the extensions.

When a folder is uploaded, it is recorded in the database, and consequently, also every file and folder it contains.

Files and folders can be shared with other users, marked as "favorites" or deleted, or moved to the trash until they are permanently deleted.

The maximum file size that can be uploaded is 10GB, there are some limitations on extensions for security reasons.

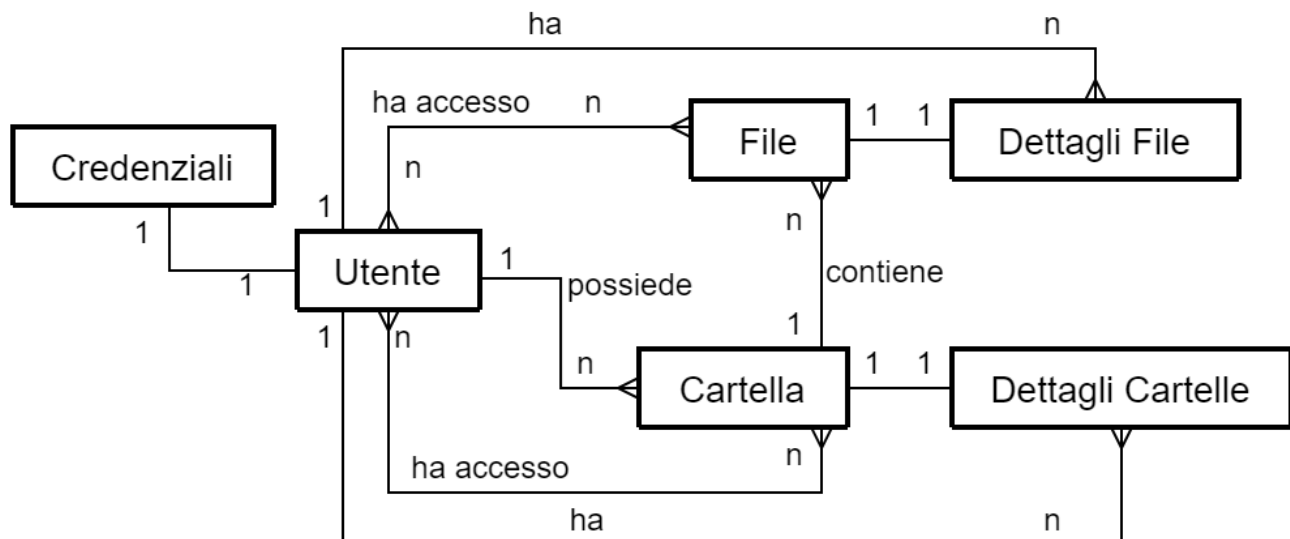
2.a Database

The database, necessary for the operation of the application, contains the credentials entity, which is used for access to the cloud storage service. The entities for the registration of the folders and files contained in them are also essential.

To allow certain features, such as sharing, entities are required for user access to folders and for sharing individual files, in addition to those containing information in the case of folders or files marked as favorites or trashed.

The database will be managed using the Linux operating system, with MySQL Relational Database Management System (RDBMS), the database will be coded in SQL language.

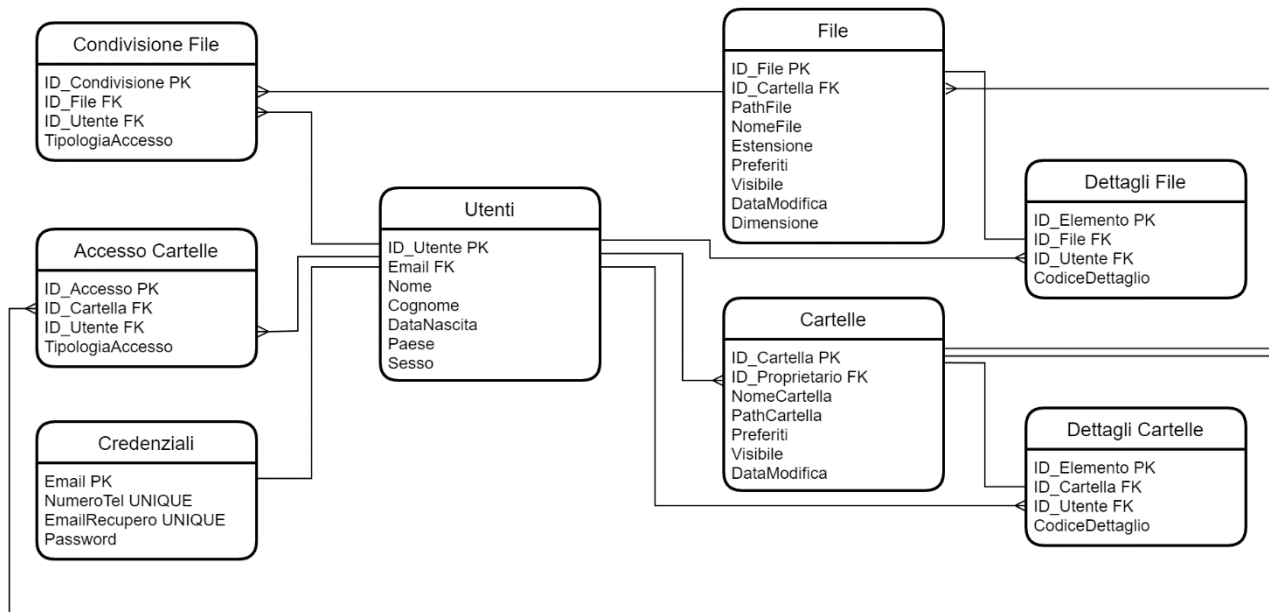
Conceptual Diagram



Reading rules

- A user has credentials, the credentials belong to only one user.
- A user owns multiple folders, a folder is owned by only one user.
- A folder contains multiple files, a file is contained by only one folder.
- The more users have access to more files, the more files are accessible to more users.
- The more users have access to more folders, the more folders are accessible to more users.
- A file has details, details belong to only one file.
- A folder has details, details belong to only one folder.
- One user has more details about files, the details of a file are for one user only.
- One user has more folder details, one folder details are for one user only.

E-R scheme



To solve the problem of entities with many-to-many relationships, you must also include the CondivisioneFile and AccessoCartelle entities.

The new reading rules include:

- a user has shared multiple files; a file can be shared with multiple users;
- a user can have access to multiple folders, a folder can be shared with multiple users.

Logical Scheme

Users

Field name	Key	Typology	Length	Notes
ID_Utente	PK	INT	10	Autoincrementante
Email	FK	CHAR	255	
Nome		CHAR	50	Not null
Cognome		CHAR	50	Not null
DataNascita		DATE	-	Not null
Paese		CHAR	50	Not null
Sesso		CHAR	1	

Credentials

Field name	Key	Typology	Length	Notes
Email	PK	CHAR	255	
NumeroTel		CHAR	14	UNIQUE
EmailRecupero		CHAR	255	UNIQUE, Not null
Password		CHAR	50	Not null

Folders

Field name	Key	Typology	Length	Notes
ID_Cartella	PK	INT	15	Autoincrementante
ID_Proprietario	FK	INT	10	
NomeCartella		CHAR	255	Not null
PathCartella		CHAR	255	UNIQUE
Preferiti		CHAR	1	Not null
Visibile		CHAR	1	Not null
DataModifica		DATE	-	Not null

File

Field name	Key	Typology	Length	Notes
ID_File	PK	INT	20	Autoincrementante
ID_Cartella	FK	INT	15	Not null
NomeFile		CHAR	255	UNIQUE
PathFile		CHAR	255	Not null
Estensione		CHAR	5	Not null
Preferiti		CHAR	1	Not null
Visibile		CHAR	1	Not null
DataModifica		DATE	-	Not null
Dimensione		INT	15	Not null

Access Folders

Field name	Key	Typology	Length	Notes
ID_Accesso	PK	INT	20	Autoincrementante
ID_Cartella	FK	INT	15	
ID_Utente	FK	INT	10	
TipologiaAccesso		CHAR	3	Not null

Share File

Field name	Key	Typology	Length	Notes
ID_Condivisione	PK	INT	25	Autoincrementante
ID_File	FK	INT	20	
ID_Utente	FK	INT	10	
TipologiaAccesso		CHAR	3	Not null

Folders options

Field name	Key	Typology	Length	Notes
ID_Accesso	PK	INT	15	Autoincrementante
ID_Cartella	FK	INT	15	
ID_Utente	FK	INT	10	
TipologiaAccesso		CHAR	4	Not null

File options

Field name	Key	Typology	Length	Notes
ID_Condivisione	PK	INT	20	Autoincrementante
ID_File	FK	INT	20	
ID_Utente	FK	INT	10	
TipologiaAccesso		CHAR	4	Not null

Entity description and relationships

There is a total of eight entities: the *utenti* table contains the entire registry of registered users and has a 1 to 1 relationship with *credenziali*. This entity contains sensitive information: assigned MicroDrive email, phone number and recovery email, password.

Each user has multiple *cartelle* which can be marked as favorites or be trashed, in which case they will no longer be visible in the directories but only in the trash section of the site. The date of the last modification is also recorded. In the event that the folder is marked as a favorite or is trashed, its reference is added to the *opzioni cartelle* table within which it is established, by means of a code, whether the linked folder is among the favorites or deleted.

A folder contains several *file*, of which the name and extension are saved in addition to the size and characteristics as for folders.

The folders also have a reference to the path, with which it is possible to go back to the parent folder and consequently also to those children. Instead, files have a path that indicates the actual location of the file on the servers.

Sharing can be done either by whole folder or by single file, respectively through the *accesso cartelle* and *condivisione file* entities.

Database creation and management

To create and manage the database using MySQL, you must proceed with the execution of SQL Query. Subsequently, all the operations that require the use of the database must be carried out by executing Queries that query the database and provide the correct results.

Creating tables

To create the tables, you must use Query of the DDL (Data Definition Language) type. In particular, the Queries necessary for the creation of the entire DB follow.

Users Entity

```
CREATE TABLE 'users' (
CREATE TABLE `utenti` (
  `ID_Utente` INT(10) NOT NULL AUTO_INCREMENT ,
  `Email` CHAR(255) NOT NULL ,
  `Nome` CHAR(50) NOT NULL ,
  `Cognome` CHAR(50) NOT NULL ,
  `DataNascita` DATE NOT NULL ,
  `Paese` CHAR(50) NOT NULL ,
  `Sesso` CHAR(1) ,
```

```
PRIMARY KEY (`ID_Utente`) ,  
FOREIGN KEY (`Email`) REFERENCES `credenziali`(`Mail`)  
);
```

Credential Entity

```
CREATE TABLE `credenziali` (  
  `Mail` CHAR(255) NOT NULL AUTO_INCREMENT ,  
  `NumeroTel` CHAR(14) ,  
  `EmailRecupero` CHAR(255) NOT NULL ,  
  `Password` CHAR(50) NOT NULL ,  
  PRIMARY KEY (`ID_Utente`)  
  UNIQUE (`NumeroTel`)  
  UNIQUE (`EmailRecupero`)  
);
```

Folder Entity

```
CREATE TABLE `cartelle` (  
  `ID_Cartella` INT(15) NOT NULL AUTO_INCREMENT ,  
  `ID_Proprietario` INT(10) NOT NULL ,  
  `PathCartella` CHAR(255) NOT NULL ,  
  `NomeCartella` CHAR(255) NOT NULL ,  
  `Preferiti` CHAR(1) NOT NULL ,  
  `Visibile` CHAR(1) NOT NULL ,  
  `DataModifica` DATE NOT NULL ,  
  PRIMARY KEY (`ID_Cartella`) ,  
  UNIQUE `NomeCartella` (`NomeCartella`, `PathCartella`) ,  
  FOREIGN KEY (`ID_Proprietario`) REFERENCES `utenti`(`ID_Utente`)  
);
```

File Entity

```
CREATE TABLE `file` (  
  `ID_File` INT(20) NOT NULL AUTO_INCREMENT ,  
  `ID_Cartella` INT(15) NOT NULL ,  
  `PathFile` CHAR(255) NOT NULL ,  
  `NomeFile` CHAR(255) NOT NULL ,  
  `Estensione` CHAR(5) NOT NULL ,  
  `Preferiti` CHAR(1) ,  
  `Visibile` CHAR(1) ,
```

```
`DataModifica` DATE CURRENTDATE ,  
`Dimensione` INT(15) ,  
FOREIGN KEY (`ID_Cartella`) REFERENCES `cartelle`(`ID_Cartella`) ,  
UNIQUE `NomeFile` (`NomeFile`, `ID_Cartella`) ,  
PRIMARY KEY (`ID_File`)  
);
```

Access EntitiesFolders

```
CREATE TABLE `accessocartelle` (  
`ID_Accesso` INT(20) NOT NULL AUTO_INCREMENT ,  
`ID_Cartella` INT(15) NOT NULL ,  
`ID_Utente` INT NOT NULL ,  
`TipologiaAccesso` CHAR(3) NOT NULL ,  
PRIMARY KEY (`ID_Accesso`) ,  
FOREIGN KEY (`ID_Cartella`) REFERENCES `cartelle`(`ID_Cartella`) ,  
FOREIGN KEY (`ID_Utente`) REFERENCES `utenti`(`ID_Utente`)  
);
```

Folder OptionsEntity

```
CREATE TABLE `opzionicartelle` (  
`ID_Elemento` INT(15) NOT NULL AUTO_INCREMENT ,  
`ID_Cartella` INT(15) NOT NULL ,  
`ID_Utente` INT NOT NULL ,  
`CodiceDettaglio` CHAR(4) NOT NULL ,  
PRIMARY KEY (`ID_Elemento`) ,  
FOREIGN KEY (`ID_Cartella`) REFERENCES `cartelle`(`ID_Cartella`) ,  
FOREIGN KEY (`ID_Utente`) REFERENCES `utenti`(`ID_Utente`)  
);
```

File Share Entity

```
CREATE TABLE `condivisionefile` (  
`ID_Condivisione` INT(25) NOT NULL AUTO_INCREMENT ,  
`ID_File` INT(20) NOT NULL ,  
`ID_Utente` INT(10) NOT NULL ,  
`TipologiaAccesso` CHAR(3) NOT NULL ,  
PRIMARY KEY (`ID_Condivisione`) ,  
FOREIGN KEY (`ID_File`) REFERENCES `file`(`ID_File`) ,  
FOREIGN KEY (`ID_Utente`) REFERENCES `utenti`(`ID_Utente`)  
);
```

File OptionsEntity

```
CREATE TABLE `opzionifile` (  
  `ID_Elemento` INT(20) NOT NULL AUTO_INCREMENT ,  
  `ID_File` INT(20) NOT NULL ,  
  `ID_Utente` INT(10) NOT NULL ,  
  `CodiceDettaglio` CHAR(4) NOT NULL ,  
  PRIMARY KEY (`ID_Elemento`) ,  
  FOREIGN KEY (`ID_File`) REFERENCES `file`(`ID_File`) ,  
  FOREIGN KEY (`ID_Utente`) REFERENCES `utenti`(`ID_Utente`)  
);
```

Inserting and editing data

To enter and modify data in the tables, Query of the DML (Data Manipulation Language) type must be used.

In particular, the Queries needed to manage some system functions follow.

Entering Credentials

```
INSERT INTO `credenziali` (  
  `Email`,  
  `NumeroTel`,  
  `EmailRecupero`,  
  `Password`  
)  
VALUES (  
  'utentel@microdrive.com',  
  '+393336669999',  
  'recupero1@mail.com',  
  'admin'  
);
```

Registering a User

```
INSERT INTO `utenti` (  
  `ID_Utente`,  
  `Email`,  
  `Nome`,  
  `Cognome`,  
  `DataNascita`,  
  `Paese`,
```

```
`Sesso`  
)  
VALUES (  
NULL,  
'utente1@microdrive.com',  
'Nome1',  
'Cognome1',  
'2000/01/01',  
'Italia',  
'M'  
);
```

Insert a Folder

```
INSERT INTO `cartelle` (  
`ID_Cartella`,  
`ID_Proprietario`,  
`PathCartella`,  
`NomeCartella`  
)  
VALUES (  
NULL,  
'1',  
'Home/',  
'Home'  
);
```

Insert a File

```
INSERT INTO `file` (  
`ID_File`,  
`ID_Cartella`,  
`PathFile`,  
`NomeFile`,  
`Estensione`,  
)  
VALUES (  
NULL,  
'1',  
'user1/Home/Elaborato.dox',  
'Elaborato',
```

```
'docx',  
);
```

Rename a File

```
UPDATE `file`  
SET `PathFile` = 'C:/user1/Home/Elaborato1.docx',  
    `NomeFile` = 'Elaborato1'  
WHERE `file`.`ID_File` = 1;
```

Rename a Folder

```
UPDATE `cartelle`  
SET `PathCartella` = 'Home/Folder1',  
    `NomeCartella` = 'Folder1'  
WHERE `cartelle`.`ID_Cartella` = 2;
```

Data visualization

To view the data in the tables you must use Query of the type QL (Query Language - Data Query Language DQL). In particular, the Queries needed to manage some system functions follow.

Selecting All User Files 1

```
SELECT * FROM file, cartelle  
WHERE (  
    (file.ID_Cartella=cartelle.ID_Cartella) AND  
    (cartelle.ID_Proprietario=1)  
);
```

Selecting all user's 1 favorite files

```
SELECT * FROM file, opzionifile  
WHERE (  
    (file.ID_File = opzionifile.ID_File) AND  
    (opzionifile.CodiceDettaglio="PREF") AND  
    (opzionifile.ID_Utente=1)  
);
```

2.b Website

The website consists of eight php pages consisting of an html structure and a CSS style sheet, as well as the PHP code for the Apache server.

The site can be reached from the internet and, if you are not logged in, the first page to be displayed is the home page which shows the general characteristics of the service with an infographic and invites the visitor to register or log in.

To access the service there is a button with the wording "Login" which redirects to the login page. If the user is not yet registered, he will instead have to register by accessing the dedicated page by pressing the "Test MicroDrive" button.

It is then possible to make changes to your account at a later time through the dedicated page.

After logging in, the main screen of the service is displayed, showing the files and folders in the "Home" directory. By selecting one of the folders presents, the page will update, displaying the files contained therein.

By right-clicking on files and folders, you can perform various options such as view file information (last modified date, size, owner) and share the file or folder in question with other registered users.

There is then another page of the site where all the files and folders shared with the user are visible. To access it, press the "Shared with me" button. Another section that can be consulted is that of favorites, also accessible by pressing the relevant button. Finally, there is the "Trash" button which shows the relative page with the deleted files inside.

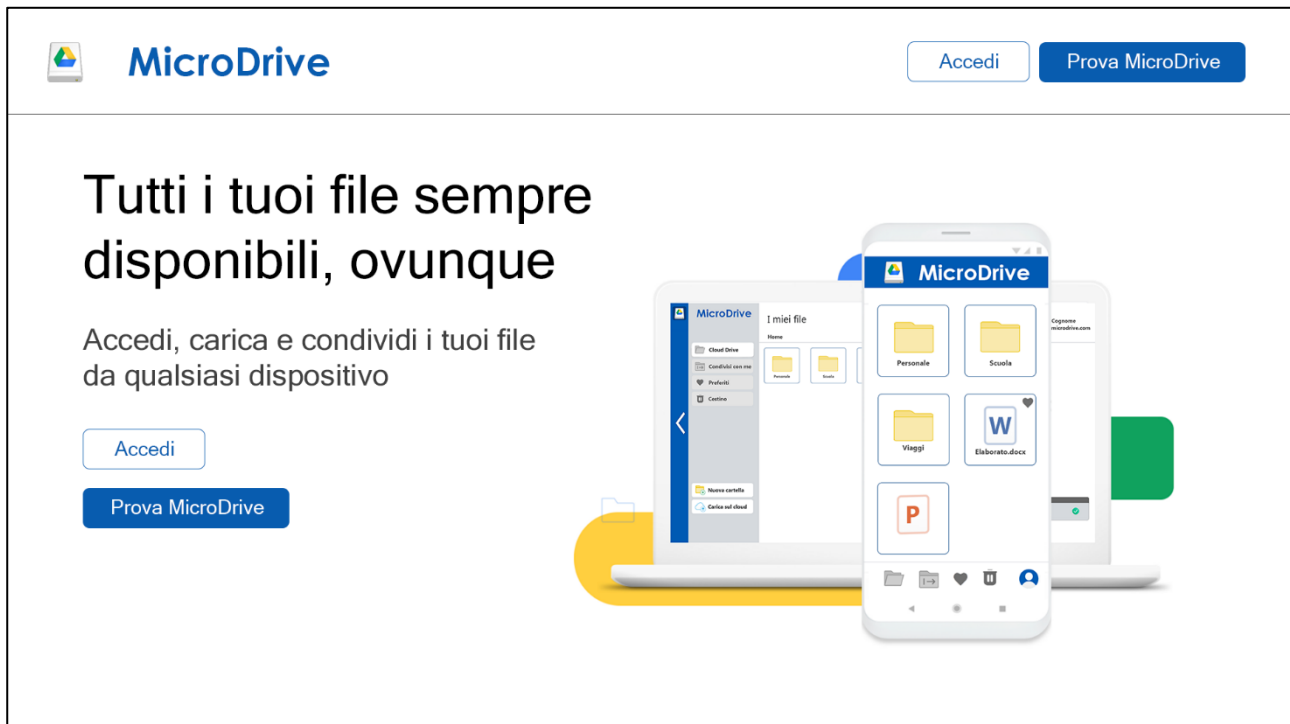
Site map

- Home page (www.microdrive.com)
 - Cloud
 - Home Cloud (cloud.microdrive.com/homeCloud)
 - Shared with me (cloud.microdrive.com/shared)
 - Favorites (cloud.microdrive.com/preferred)
 - Recycle Bin(cloud.microdrive.com/bin)
 - account
 - Login (account.microdrive.com/login)
 - Logging (account.microdrive.com/signup)
 - Edit Profile (account.microdrive.com/editProfile)

WEB pages

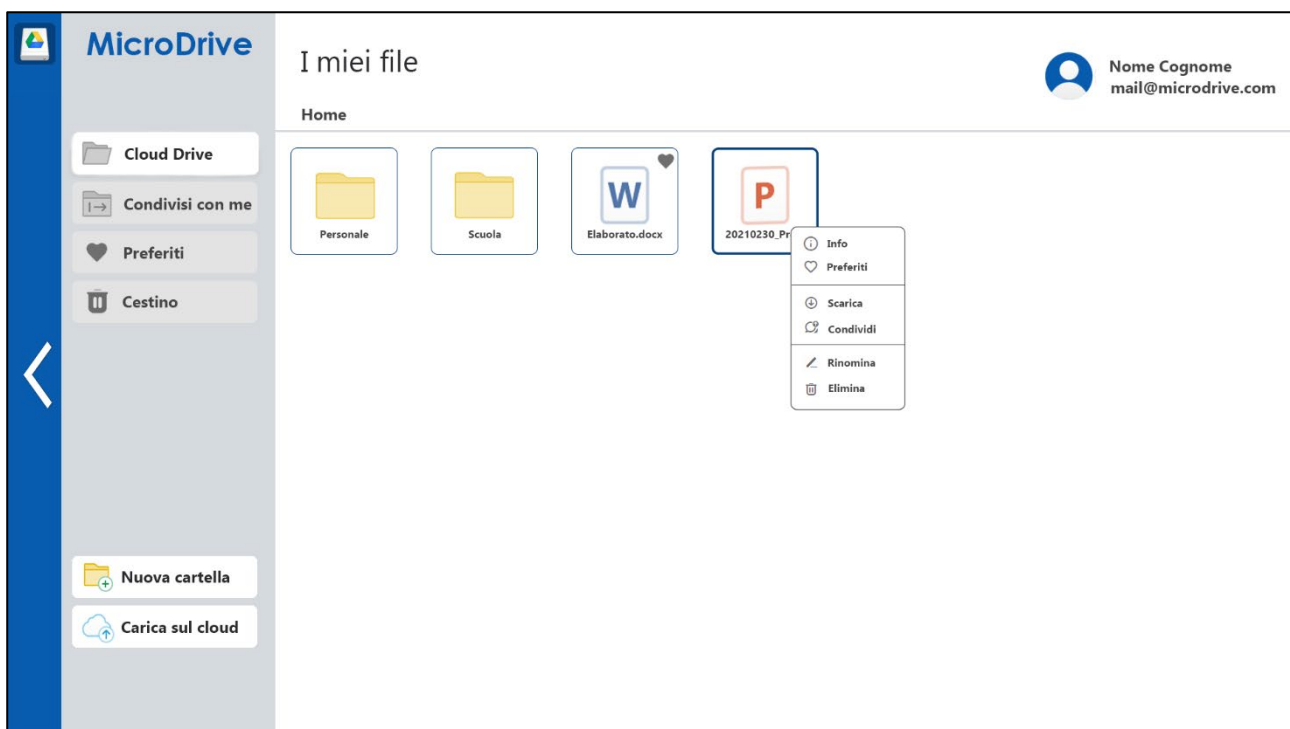
Home

www.microdrive.com

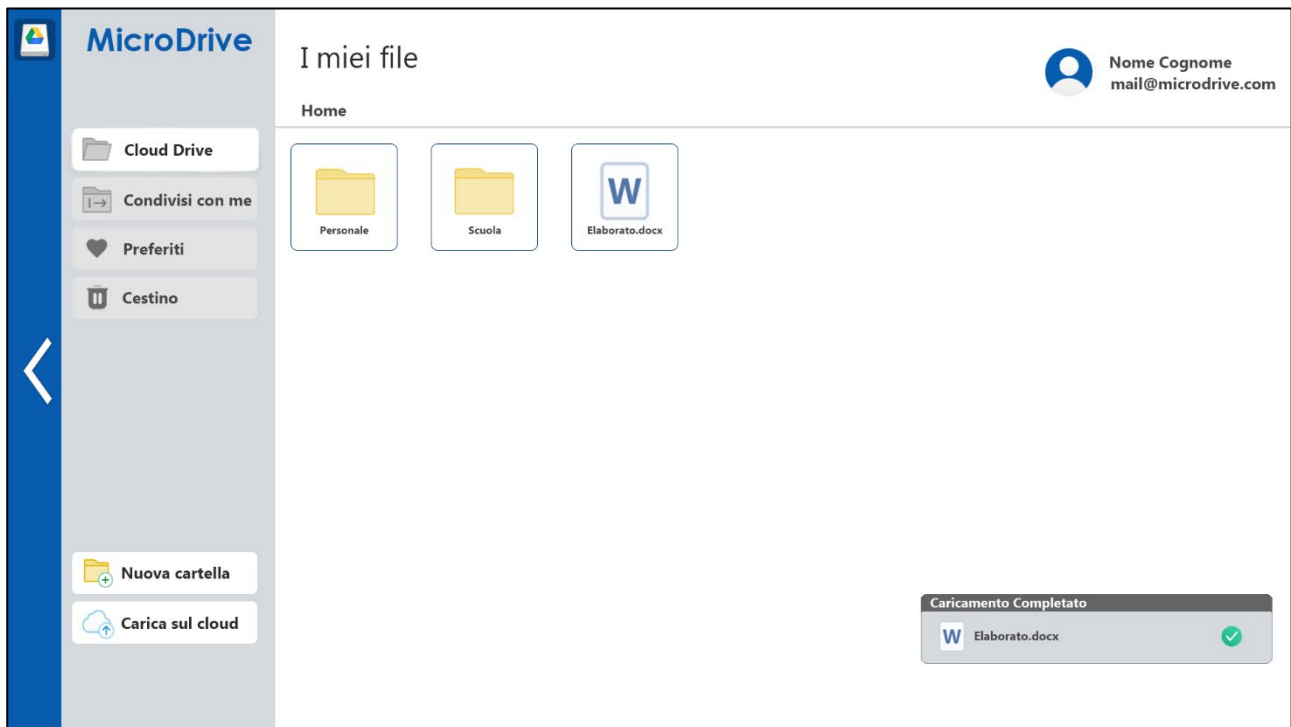


Home cloud

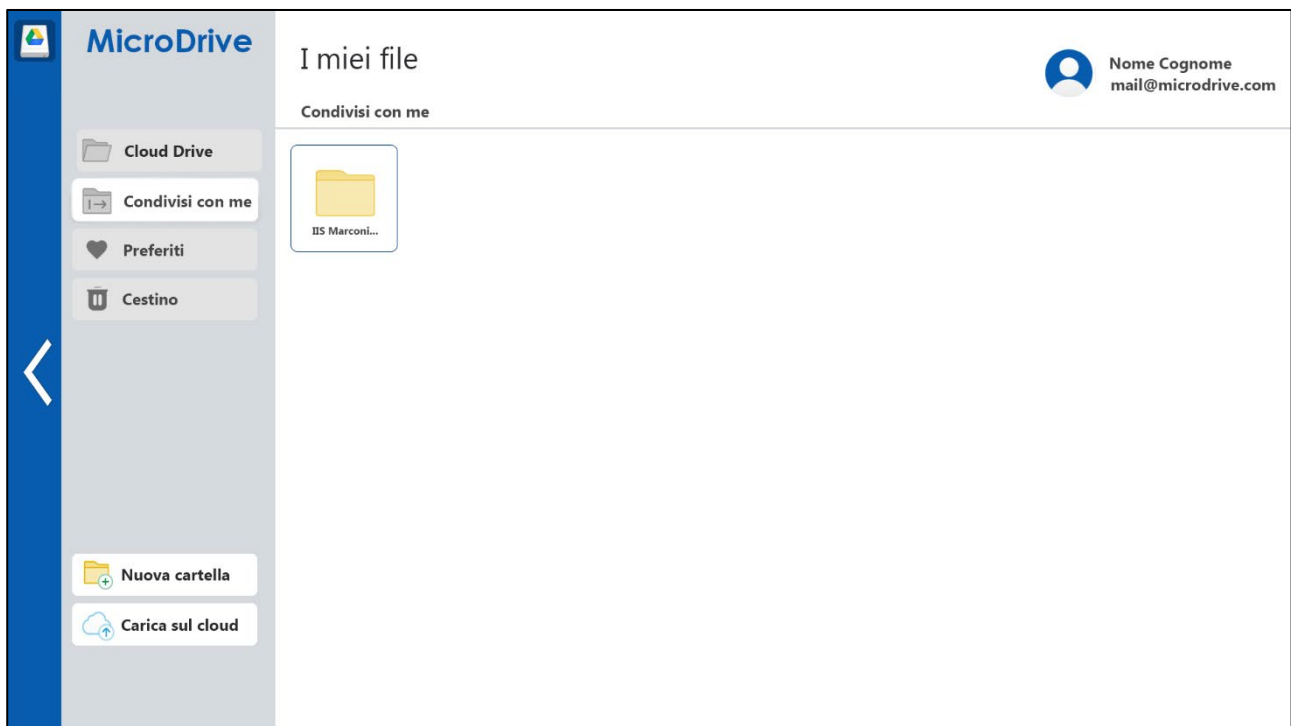
cloud.microdrive.com/homeCloud

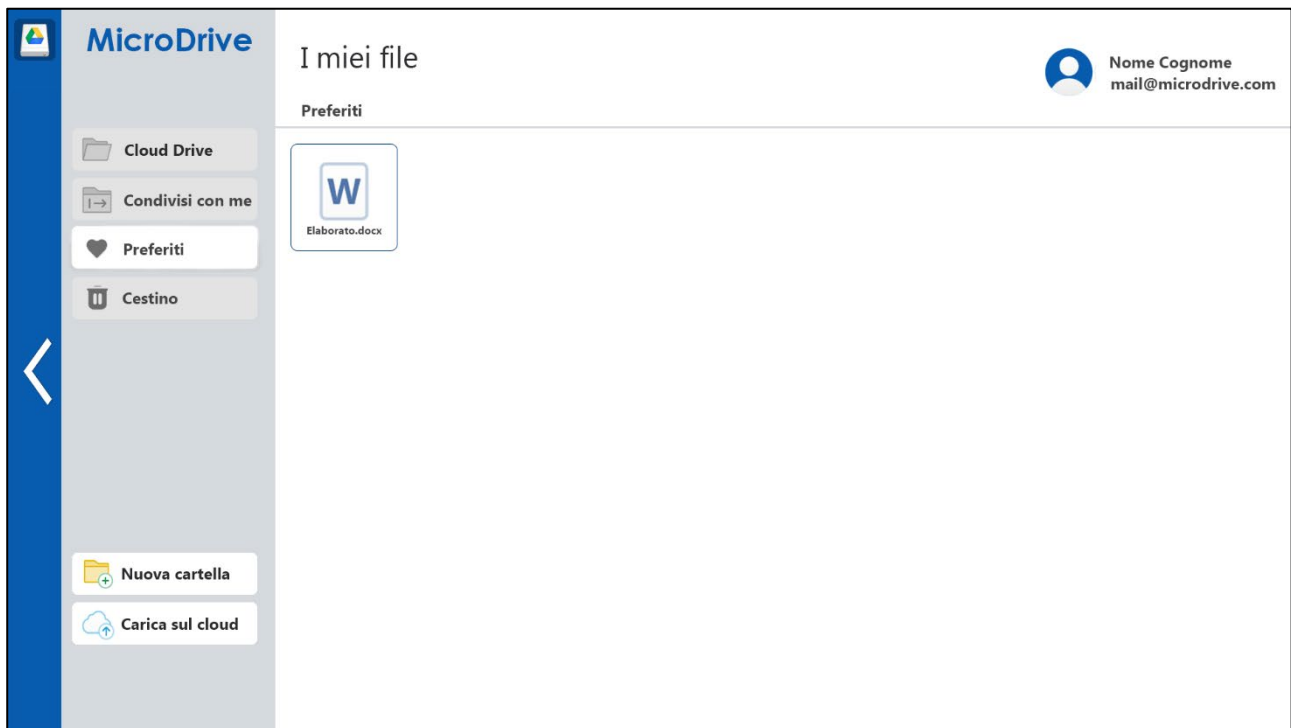
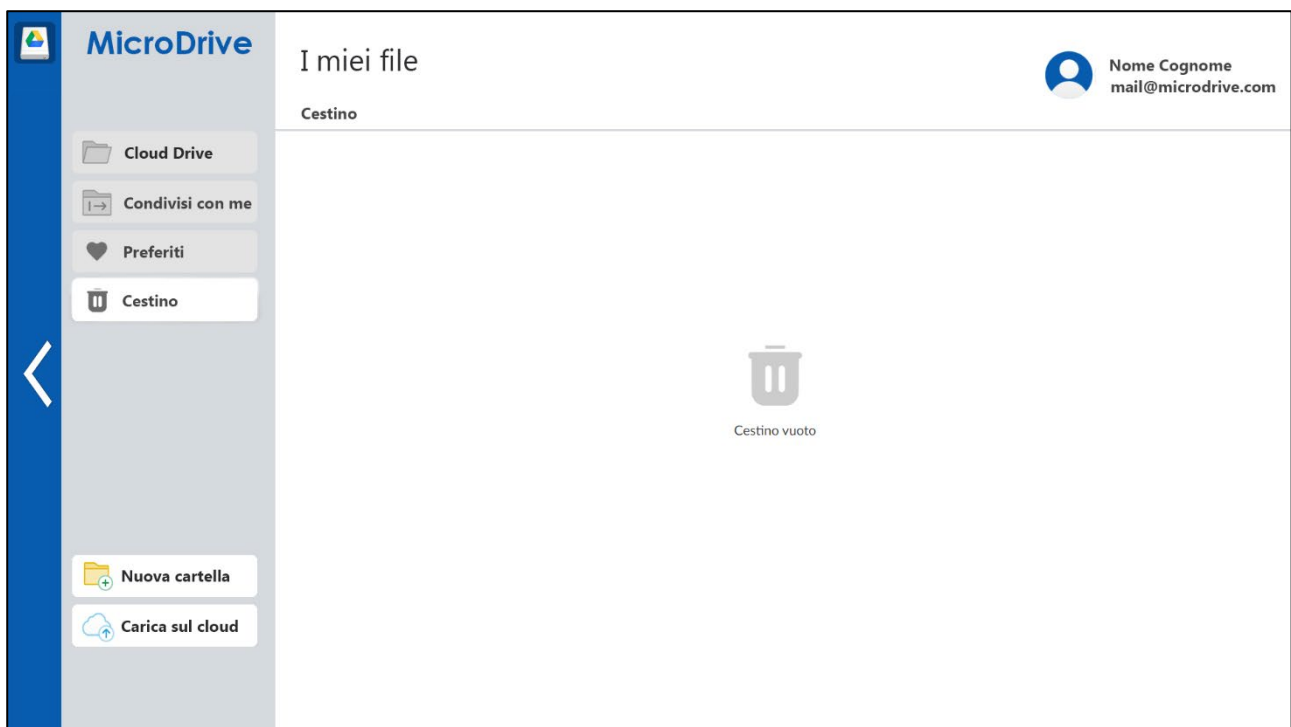


Upload a file



Shared with me

cloud.microdrive.com/shared

Preferredcloud.microdrive.com/preferred**Bin**cloud.microdrive.com/bin

Registration

account.microdrive.com/signup

 **MicroDrive**

Accedi

Crea il tuo account MicroDrive

Nome

Cognome

email

@microdrive.com

Indirizzo email di recupero

Data di nascita

Sesso

gg

mm

aaaa

M

Paese

Numero di telefono

Password

Conferma password

Registrazione

Login

account.microdrive.com/login

 **MicroDrive**

Accedi

Accedi al tuo account MicroDrive

email

@microdrive.com

Password

Accedi

Edit Profile

account.microdrive.com/editProfile

 **MicroDrive**

Accedi

Modifica il tuo account MicroDrive

Nome1

Cognome1

utente1

@microdrive.com

emailrecupero1@mail.com

Data di nascita

Sesso

01

01

2000

M

Italia

+39 3336669999

Password

Conferma password

Modifica

PHP code

Php code is required to be able to do this, and it interfaces with databases in order to work with the user's data and files. Each page of the website has its own code.

The following code is a part of what is required for logging, updating account, login, and uploading files.

Registration

```

1  <?php
2  //inizializzazione delle variabili
3      $password = "";
4      $errore="";
5      //email e tipodip in sessione dato che dobbiamo portarlo anche in gestione
       paziente
6          $nome = $email = $emailRrecupero= $password = $passwordC = $sesso =
           $data = $paese = $NumTel= $cognome= $giorno=$mese=$anno='';
7      $erroriR=false;
8
9  //funzione controllo data
10 function ControlloData($data){
11     $arrayData = explode("-", $data);
12     $Anno = $arrayData[0];
13     $Mese = $arrayData[1];
14     $Giorno = $arrayData[2];
15     if(is_numeric($Anno))
16     {
17         if(!checkdate($Mese, $Giorno, $Anno)){
18             return false;
19         }else{
20             if($data<date("Y-m-d")){
21                 return true;
22             }else
23             {
24                 return false;
25             }
26         }
27     }
28 }
29
30 $opzioniPaesi = " "; //menu tendina paese
31 $myfile = fopen("paesi.csv", "r") or die ("Lista dei paesi non trovato.");
32 //scorro il file per trovare il paese giusto
33 while (!feof($myfile)) {
34     $linea = fgets($myfile);
35     if (!feof($myfile)) {
36         $array = explode(",", $linea);
37         $paese = trim($array[1]);
38         $opzioniPaesi = $opzioniPaesi . "<option value=" . $array[1] . ">" .
           $paese . "</option>";
39     }
40 }
41
42 if(isset($_POST['Registrazione'])) /*INIZIO REGISTRAZIONE*/
43 {
44     require("datidb.php");
45     if (isset($_POST['nome'])) $nome = trim($_POST['nome']);
46     if (isset($_POST['cognome'])) $cognome = trim($_POST['cognome']);
47     if (isset($_POST['email'])) $email= trim($_POST['email']);
48     $email.="microdrive.com";
49
50     if (isset($_POST['emailRecupero'])) $emailRecupero=
       trim($_POST['emailRecupero']);
51
52     if (isset($_POST['giorno'])) $giorno = trim($_POST['giorno']);
53     if (isset($_POST['mese'])) $mese = trim($_POST['mese']);
54     if (isset($_POST['anno'])) $anno = trim($_POST['anno']);
55     $data=$anno; $data.="-"; $data.=$mese;$data.="-";$data.=$giorno;
56
57     if (isset($_POST['paese'])) $paese = trim($_POST['paese']);
58
59     if (isset($_POST['password'])) $password = trim($_POST['password']);
60     if (isset($_POST['Cpassword'])) $passwordC = trim($_POST['Cpassword']);
61     if (isset($_POST['NumTel'])) $NumTel = trim($_POST['NumTel']);
62
63
64     $datavalida=ControlloData($data); //funzione per il controllo della data
65     if(!$datavalida){
66         $erroriR=true;

```

```

67         $errore= "Data non valida";
68     }
69
70     //verifico che le due password corrispondano
71     if($Cpassword!=$password){
72         $erroriR=true;
73         $errore= "Le password non corrispondono";
74     }
75
76     //verifico che il numero di telefono abbia la giusta lunghezza
77     if(strlen($NumTel)<10 || strlen($NumTel)>14){
78         $erroriR=true;
79         $errore= "Numero di telefono errato";
80     }
81     if (!$erroriR)
82     {
83         // non ci sono errori salvo dati su database
84         // connessione libreria mysqli
85         $conn = new mysqli($DB_HOST, $DB_USER, $DB_PASS, $DB_NAME);
86         // controllo se la connessione ha avuto successo
87         if ($conn->connect_error) {
88             // errore di connessione
89             $errore= "ERRORE CONNESSIONE DATABASE";
90             die("ERRORE CONNESSIONE DATABASE"); // termina script
91         } else {
92             //Query per vedere se l'utente è già registrato
93             $sql="SELECT Email FROM Credenziali WHERE Email='" . $Email .
94             "'";
95             $pQuery = $conn->query($sql);
96             if ($pQuery->num_rows != 1) {
97                 //Inserimento utente nella tabella credenziali
98                 $pQuery = $conn->prepare("INSERT INTO Credenziali (Email,
99                 NumeroTel, emailRecupero, PW) VALUES (?, ?, ?, ?);");
100                 $pQuery->bind_param("ssss", $email, $NumTel,
101                 $emailRecupero, $password);
102                 $pQuery->execute();
103                 var_dump($pQuery->error);
104                 $righeins = $pQuery->affected_rows;
105                 if ($righeins==1)
106                 {
107                     //Inserimento utente nella tabella utenti
108                     $pQuery = $conn->prepare("INSERT INTO Utenti (Email,
109                     Nome, Cognome, DataNascita, Paese, Sesso) VALUES
110                     (?, ?, ?, ?, ?, ?);");
111                     $pQuery->bind_param("ssssss", $email, $nome, $cognome,
112                     $data, $paese, $NumTel);
113                     $pQuery->execute();
114                     var_dump($pQuery->error);
115                     $righeins = $pQuery->affected_rows;
116                     //Controllo che l'inserimento si andato a buon fine
117                     if($righeins>0)
118                     {
119                         $errore= "Inserimento avvenuto correttamente";
120                     }else
121                     {
122                         $errore= "Qualcosa è andato storto, riprova";
123                     }
124                     header("location:
125                     cloud.microdrive.com/home-cloud.php");
126                 }
127             } else
128             {
129                 // errore di esecuzione
130                 $errore= "MEMORIZZAZIONE SU DATABASE FALLITA!";
131             }
132             $pQuery->close();
133             $conn->close();
134         } else{
135             $errore= "Sei già registrato";
136         }
137     }

```

```

130         }
131     }
132 }
133 ?>

```

Login

```

1  <?php
2  //inizializzazione delle variabili
3      $errore= $nome = $email = $password = $cognome='';
4      $erroriR=false;
5
6  if(isset($_POST['Accedi'])){ //clicco su login
7      require("datidb.php"); //pagina richiesta
8      // controllo se esistono dati in post
9      if (isset($_POST['email'])){
10         $email=trim($_POST['email']);
11     }
12     if (isset($_POST['password'])) $password = trim($_POST['password']);
13     // connessione libreria mysqli
14     $conn = new mysqli($DB_HOST, $DB_USER, $DB_PASS, $DB_NAME);
15     // controllo se la connessione ha avuto successo
16     if ($conn->connect_error)
17     {
18         // stampa errore
19         $errore= "ERRORE CONNESSIONE DATABASE";
20         // termina script
21         die("ERRORE CONNESSIONE DATABASE");
22     } else {
23         // connessione riuscita
24         // preparazione della query
25         //seleziona la password associata all'email
26         $sql="SELECT Password FROM credenzili WHERE Email='" . $email
27             . "'";
28         $pQuery = $conn->query($sql);
29
30         //Se trova l'utente nelle credenzili recupera i dati
31         if ($pQuery->num_rows > 0) {
32             $row = $pQuery->fetch_assoc();
33             if($row["Password"]== $password){
34                 $pQuery->close();
35                 //seleziona l'utente associata all'email
36                 $sql="SELECT ID_Utente, Nome, Cognome FROM utenti
37                     WHERE Email='" . $email . "'";
38                 $pQuery = $conn->query($sql);
39                 if($pQuery->num_rows > 0) {
40                     $row = $pQuery->fetch_assoc();
41                     $_SESSION['ID_Utente']=$row["ID_Utente"];
42                     $_SESSION['Nome']=$row["Nome"];
43                     $_SESSION['Cognome']=$row["Cognome"];
44                     //indirizzamento alla pagina Home Cloud
45                     header("location:
46                         cloud.microdrive.com/home-cloud.php");
47                 }
48                 $errore="Errore DB";
49             }
50             else{
51                 // errore di esecuzione
52                 $errore="Password errata";
53             }
54         } else {
55             $errore="Utente non trovato";
56         }
57         $conn->close();
58     }
59 }
60 ?>

```

Edit Profile

```

1  <?php
2  session_start();
3
4  if (isset($_SESSION["ID_Utente"])) $ID_Utente = $_SESSION["ID_Utente"];
5  //dichiarazione errori
6  $texterrori= $Dati= $PasswordC= $Password= $Nome= $Cognome= $DataNascita=
   $Paese= $EmailRecupero= $NumeroTel= $Sesso= " ";
7  $errori = false;
8
9  /*INIZIO Modifica*/
10     require("datidb.php");
11     // connessione libreria mysqli
12     $conn = new mysqli($DB_HOST, $DB_USER, $DB_PASS, $DB_NAME);
13     // controllo se la connessione ha avuto successo
14     if ($conn->connect_error)
15     {
16         // errore di connessione
17         $texterrori= "ERRORE CONNESSIONE DATABASE";
18         die("ERRORE CONNESSIONE DATABASE"); // termina script
19     } else {
20         //Query per la password dell'utente
21         $sql="SELECT * FROM utenti WHERE ID_Utente=' " .
           $_SESSION['ID_Utente'] . "'";
22         $pQuery = $conn->query($sql);
23         if ($pQuery->num_rows > 0) {
24             while($Dati = mysqli_fetch_assoc($pQuery)) {
25                 $Email= $Dati["Email"];
26                 $Nome= $Dati["Nome"];
27                 $Cognome= $Dati["Cognome"];
28                 $DataNascita= $Dati["DataNascita"];
29                 $Paese= $Dati["Paese"];
30                 $Sesso= $Dati["Sesso"];
31             }
32         } else {
33             $texterrori= "Qualcosa è andato storto";
34         }
35         //Leggo e mi salvo i valori dei campi tramite la query
36         $sql="SELECT * FROM credenziali WHERE Email=' " . $Email .
           "'";
37         $pQuery = $conn->query($sql);
38         if ($pQuery->num_rows > 0) {
39             while($Dati = mysqli_fetch_assoc($pQuery)) {
40                 $EmailRecupero= $Dati["EmailRecupero"];
41                 $NumeroTel= $Dati["NumeroTel"];
42                 $Password= $Dati["Password"];
43             }
44             $pQuery->close();
45         } else {
46             $texterrori= "Qualcosa è andato storto";
47         }
48         $conn->close();
49     }
50     /*MODIFICA*/
51     if(isset($_POST['Modifica'])) {
52         if (isset($_POST['Password'])) $Password = trim($_POST['Password']);
53         if (isset($_POST['PasswordC'])) $PasswordC= trim($_POST['PasswordC']);
54         if (isset($_POST['NumeroTel'])) $NumeroTel = trim($_POST['NumeroTel']);
55         if (isset($_POST['Nome'])) $Nome = trim($_POST['Nome']);
56         if (isset($_POST['Cognome'])) $Cognome = trim($_POST['Cognome']);
57         if (isset($_POST['Sesso'])) $Sesso = trim($_POST['Sesso']);
58         if (isset($_POST['DataNascita'])) $DataNascita =
           trim($_POST['DataNascita']);
59         //Verifico che le password corrispondano
60         if($Password!= $PasswordC) {
61             $errori=true;
62             $texterrori= "Le password non corrispondono";
63         }
64         //Verifico che il numero di telefono abbia la lunghezza giusta
65         if(strlen($NumeroTel)<10 || strlen($NumeroTel)>10) {
66             $errori=true;

```

```

67         $texterrori= "Numero di telefono errato";
68     }
69     //Se non ci sono errori procedo
70     if (!$errori)
71     {
72         // connessione libreria mysqli
73         $conn = new mysqli($DB_HOST, $DB_USER, $DB_PASS, $DB_NAME);
74         if ($conn->connect_error)
75         {
76             // errore di connessione
77             $texterrori= "ERRORE CONNESSIONE DATABASE";
78             die("ERRORE CONNESSIONE DATABASE"); // termina script
79         } else {
80             //Aggiorno i dati nella tabella utenti
81             $pQuery = $conn->prepare("UPDATE utenti SET Nome = ?,
            Cognome = ?, DataNascita = ?, Sesso = ?, Paese = ? WHERE
            ID_Utente=?");
82             // query parametrica
83             $pQuery->bind_param('sssss', $Nome, $Cognome,
            $DataNascita, $Sesso, $Paese, $_SESSION['ID_Utente']);
84             $pQuery->execute();
85             var_dump($pQuery->error);
86             $righeins = $pQuery->affected_rows;
87             if ($righeins != 1) { //Verifico che la modifica sulla
            tabella utenti sia andata a buon fine
88                 //Aggiorno i dati nella tabella credenziali
89                 $pQuery = $conn->prepare("UPDATE credenziali SET
            Password = ?, NumeroTel = ? WHERE Email=?");
90                 $pQuery->bind_param('sss', $Password, $NumeroTel,
            $_SESSION['email']);
91                 // query parametrica
92                 $pQuery->execute();
93                 var_dump($pQuery->error);
94                 $righeins = $pQuery->affected_rows;
95                 $pQuery->close();
96                 $conn->close();
97             } else {
98                 $texterrori= "Errore";
99             }
100         }
101     }
102 }
103 }
104 }
105 ?>

```

Upload file

```

1  <?php
2  session_start();
3
4  if (isset($_SESSION["ID_Utente"])) $ID_Utente = $_SESSION["ID_Utente"];
5  if (isset($_SESSION["ID_Cartella"])) $ID_Cartella = $_SESSION["ID_Cartella"];
6  if (isset($_SESSION["PathCartella"])) $PathCartella = $_SESSION["PathCartella"];
7
8  include 'datidb.php';
9
10 $StatoUpload = '';
11 // Lista dei formati disponibili
12 $myfile = fopen("estensioni.csv", "r") or die ("File estensioni non trovato.");
13 //scorro il file
14 while (!feof($myfile)) {
15     $linea = fgets($myfile);
16     if (!feof($myfile)) {
17         $array = explode(",", $linea);
18     }
19     $EstensioniPermesse = $array;
20     // Percorso file
21     $Directory = "user".$ID_Utente."/". $PathCartella;
22     $NomeFile = basename($_FILES["file"]["name"]);
23     $PercorsoFile = $Directory . $NomeFile;
24     $TipoFile = pathinfo($PercorsoFile,PATHINFO_EXTENSION);
25
26     if(isset($_POST["Carica"]) && !empty($_FILES["file"]["name"])){
27         if(in_array($TipoFile, $EstensioniPermesse)){
28             // Caricamento del file nel server
29             if(move_uploaded_file($_FILES["file"]["tmp_name"], $PercorsoFile)){
30                 // Inserimento del file nel database
31                 $pQuery = $conn->prepare("INSERT INTO `file` ( `ID_Cartella`,
32                     `PathFile`, `NomeFile`, `Estensione`) VALUES (?, ?, ?, ?);");
33                 $pQuery->bind_param('isssi', $ID_Cartella, $PercorsoFile, $name,
34                     $TipoFile);
35                 $pQuery->execute();
36                 var_dump($pQuery->error);
37                 $righeins = $pQuery->affected_rows;
38                 if($righeins == 1){
39                     $StatoUpload = "Il file ".$NomeFile. " è stato caricato
40                         correttamente.";
41                 }else{
42                     $StatoUpload = "Il caricamento è fallito, riprova.";
43                 }
44             }else{
45                 $StatoUpload = "Ci dispiace, c'è stato un errore durante il
46                     caricamento del tuo file.";
47             }
48         }else{
49             $StatoUpload = "Ci dispiace il tuo formato non è supportato.";
50         }
51     }else{
52         $StatoUpload = "Per favore seleziona un file da caricare.";
53     }
54 }
55 // Mostra lo stato
56 echo $StatoUpload;
57 ?>

```

3. Project management

Medium-long times are required for the realization of the project: the construction of the structures takes place in a limited time by modifying existing buildings; as regards the construction of the data center and office network infrastructure, this takes place quickly: a suitable number of people will be employed in setting up and configuring the equipment.

The budget is not a problem, despite the complexity of the project, the resources deployed by the company are huge. In any case, agreements will be made with suppliers to obtain reasonable prices on the huge amount of equipment needed.

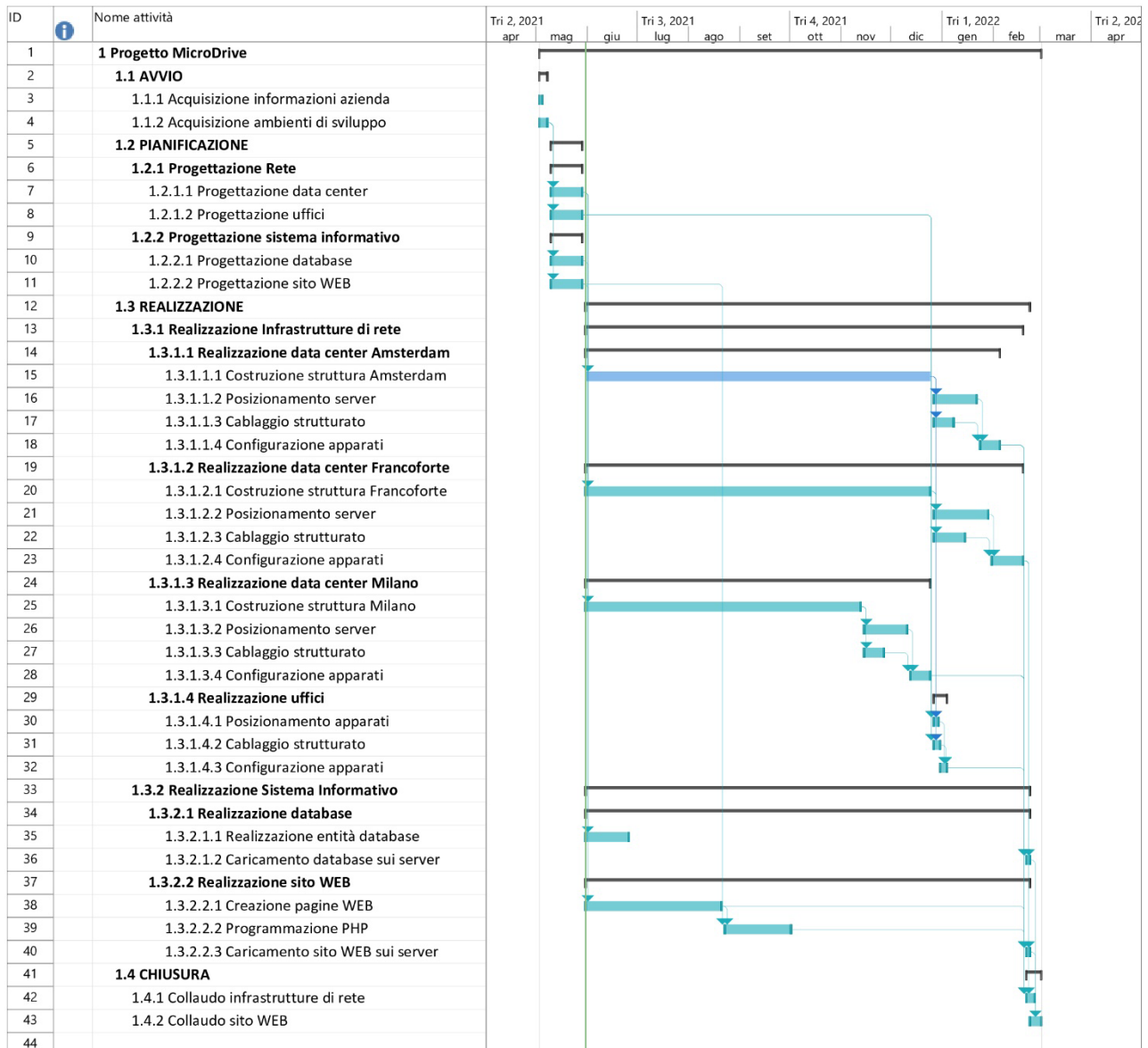
As regards the creation of the infrastructures and the preparation of the same, subcontracting companies will be used, the budget can be increased in the event of delays, increasing the number of workers and work shifts, in order to respect the delivery of the infrastructures. for the end of Q1 of 2022.

The activities and relative dates foreseen for the realization of the project are listed below, also represented in a Gantt chart.

3.a Work Breakdown Structure

ID		Nome attività	Durata	Inizio	Fine	Predecessori
1		1 Progetto MicroDrive	217 g	lun 03/05/21	mar 01/03/22	
2		1.1 AVVIO	5 g	lun 03/05/21	ven 07/05/21	
3		1.1.1 Acquisizione informazioni azienda	2 g	lun 03/05/21	mar 04/05/21	
4		1.1.2 Acquisizione ambienti di sviluppo	5 g	lun 03/05/21	ven 07/05/21	
5		1.2 PIANIFICAZIONE	15 g	lun 10/05/21	ven 28/05/21	
6		1.2.1 Progettazione Rete	15 g	lun 10/05/21	ven 28/05/21	
7		1.2.1.1 Progettazione data center	15 g	lun 10/05/21	ven 28/05/21	4
8		1.2.1.2 Progettazione uffici	15 g	lun 10/05/21	ven 28/05/21	4
9		1.2.2 Progettazione sistema informativo	15 g	lun 10/05/21	ven 28/05/21	
10		1.2.2.1 Progettazione database	15 g	lun 10/05/21	ven 28/05/21	4
11		1.2.2.2 Progettazione sito WEB	15 g	lun 10/05/21	ven 28/05/21	4
12		1.3 REALIZZAZIONE	192 g	lun 31/05/21	mar 22/02/22	
13		1.3.1 Realizzazione Infrastrutture di rete	190 g	lun 31/05/21	ven 18/02/22	
14		1.3.1.1 Realizzazione data center Amsterdam	180 g	lun 31/05/21	ven 04/02/22	
15		1.3.1.1.1 Costruzione struttura Amsterdam	150 g	lun 31/05/21	ven 24/12/21	7
16		1.3.1.1.2 Posizionamento server	20 g	lun 27/12/21	ven 21/01/22	15
17		1.3.1.1.3 Cablaggio strutturato	10 g	lun 27/12/21	ven 07/01/22	15
18		1.3.1.1.4 Configurazione apparati	10 g	lun 24/01/22	ven 04/02/22	17;16
19		1.3.1.2 Realizzazione data center Francoforte	190 g	lun 31/05/21	ven 18/02/22	
20		1.3.1.2.1 Costruzione struttura Francoforte	150 g	lun 31/05/21	ven 24/12/21	7
21		1.3.1.2.2 Posizionamento server	25 g	lun 27/12/21	ven 28/01/22	20
22		1.3.1.2.3 Cablaggio strutturato	15 g	lun 27/12/21	ven 14/01/22	20
23		1.3.1.2.4 Configurazione apparati	15 g	lun 31/01/22	ven 18/02/22	22;21
24		1.3.1.3 Realizzazione data center Milano	150 g	lun 31/05/21	ven 24/12/21	
25		1.3.1.3.1 Costruzione struttura Milano	120 g	lun 31/05/21	ven 12/11/21	7
26		1.3.1.3.2 Posizionamento server	20 g	lun 15/11/21	ven 10/12/21	25
27		1.3.1.3.3 Cablaggio strutturato	10 g	lun 15/11/21	ven 26/11/21	25
28		1.3.1.3.4 Configurazione apparati	10 g	lun 13/12/21	ven 24/12/21	26;27
29		1.3.1.4 Realizzazione uffici	6 g	lun 27/12/21	lun 03/01/22	
30		1.3.1.4.1 Posizionamento apparati	3 g	lun 27/12/21	mer 29/12/21	8;15
31		1.3.1.4.2 Cablaggio strutturato	4 g	lun 27/12/21	gio 30/12/21	8;15
32		1.3.1.4.3 Configurazione apparati	2 g	ven 31/12/21	lun 03/01/22	30;31
33		1.3.2 Realizzazione Sistema Informativo	192 g	lun 31/05/21	mar 22/02/22	
34		1.3.2.1 Realizzazione database	192 g	lun 31/05/21	mar 22/02/22	
35		1.3.2.1.1 Realizzazione entità database	20 g	lun 31/05/21	ven 25/06/21	10
36		1.3.2.1.2 Caricamento database sui server	2 g	lun 21/02/22	mar 22/02/22	18;23;28
37		1.3.2.2 Realizzazione sito WEB	192 g	lun 31/05/21	mar 22/02/22	
38		1.3.2.2.1 Creazione pagine WEB	60 g	lun 31/05/21	ven 20/08/21	11
39		1.3.2.2.2 Programmazione PHP	30 g	lun 23/08/21	ven 01/10/21	11;38
40		1.3.2.2.3 Caricamento sito WEB sui server	2 g	lun 21/02/22	mar 22/02/22	18;23;28;38;39
41		1.4 CHIUSURA	7 g	lun 21/02/22	mar 01/03/22	
42		1.4.1 Collaudo infrastrutture di rete	5 g	lun 21/02/22	ven 25/02/22	18;23;28;32
43		1.4.2 Collaudo sito WEB	5 g	mer 23/02/22	mar 01/03/22	36;40
44						

3.b Gantt chart



4. Conclusions

The project accepts all the client's requests, creating the hardware infrastructure and software for the cloud storage service.

The analysis and design carried out include, in addition to the application, also all the infrastructures necessary for operation. The work carried out takes into consideration only the company, considering users as *Internet users* and not focusing on how they connect to the designed service.

The design carried out allows the company to add new types of services without making changes to the network infrastructure, creating only the new web pages and updating the databases.