

A Project Report on
Online Backup System

Submitted as partial fulfillment of the requirement

for M.Sc.(IT) - Semester III

Prepared for Subject: **PROJECT III (ITCC13)**

Prepared By

Name of Student : Gaurav Zalavadiya

Enrollment No. : 22MSCIT22015

Name of Student : Kotharia Samir

Enrollment No. : 22MSCIT22016

Name of Student : Deep Patel

Enrollment No. : 22MSCIT22018



Shri Maneklal M. Patel Institute of Sciences and Research,

Department of M.Sc. (IT), Gandhinagar.

Shri Maneklal M. Patel Institute of Sciences and Research,
Department of M.Sc. (IT), Gandhinagar.



Date:

PROJECT WORK CERTIFICATE

This is to certify that the project report submitted by **Gaurav Zalavadiya,**
Deep Patel, Kotharia Samir having enrollment no. **22MSCIT22015,**
22MSCIT22016, 22MSCIT22018, a student of **M.Sc.(IT) Semester- III** has
completed his/her project titled “**Online Backup System**” satisfactorily during the
academic year 2022.

We appreciate the enthusiasm & dedication towards work submitted.

Internal Project Guide

Ms. Snehal Mam

M.Sc. IT, Gandhinagar

Kadi Sarva Vishwavidyalaya

Prof. B. R. Pandya

Head of Department, M.Sc. IT, Gandhinagar

Kadi Sarva Vishwavidyalaya

ACKNOWLEDGMENT

I am grateful to my internal guide **Ms. Snehal Rindani** for providing me valuable guidance for the PROJECT III and other domain related information.

Also my hearty gratitude to Head of Department, **Prof. (Dr.) B. R. Pandya** for giving me an opportunity for learning through the process of Project III by which I can improve my technical skills along with practical exposure and its applicability in the industry.

Name of Student:

Signature of Student:

Gaurav Zalavadiya

Kotharia Samir

Deep Patel

Enrollment No.: 22mscit22015, 22mscit22016, 22mscit22018

INDEX

Sr. No.	Topic	Page No.
1.	Introduction	5
2.	Scope	6
3.	Objective	7
4.	Functional Requirement	8
5.	System Architecture, Tools &Technology	10
6.	Entity Relationship Diagram Data Dictionary	11
7.	coding	15
8.	Testing	22
9.	Screen layouts	23
10.	Future Enhancement	27

Online Backup system

Introduction

An online backup system, also commonly referred to as cloud backup or remote backup, is a critical component of modern data management and protection strategies. It provides individuals and organizations with a secure and convenient method for safeguarding their digital assets, ensuring data availability, and recovering information in case of data loss or disasters.

This data can include files, documents, photos, videos, databases, and more. Online backup systems offer a reliable and convenient way to safeguard important information from various risks, including hardware failures, data corruption, accidental deletion, and disasters such as fires or floods.

In the digital age, where data plays an integral role in personal and business operations, protecting valuable information from loss or corruption is of paramount importance. Online backup systems offer a robust solution to address this need.

Online backup systems are designed to automatically and securely copy data from local devices, such as computers, servers, or mobile devices, to remote data centers hosted by third-party providers. This redundancy ensures that data remains accessible even if the original device experiences hardware failures, theft, accidental deletion, or other catastrophic events.

Scope

The scope of online backup systems is continually expanding due to the increasing importance of data in both personal and professional contexts. As technology evolves and data volumes continue to grow, the scope of online backup systems widens to encompass a range of applications and industries.

Here's an overview of the broad scope and potential areas of growth for online backup systems:

- The scope of an online backup system involves providing a secure and reliable method to store and restore digital data.
- This includes backing up various types of files, ensuring data, offering user-friendly interfaces, supporting backups, and implementing measures to protect against data breaches and loss.
- The system's scope may also extend to managing different backup versions, offering scalability, and facilitating efficient data recovery.
- The system should support a wide range of data, including documents, images, code files, and databases.
- The system should be designed to accommodate the growth of data over time, allowing users to upgrade storage space as needed.
- An easy-to-use interface is crucial for managing backups, restoring data.

Objective

The primary objectives of an online backup system, also known as a cloud backup system, are to ensure the protection, availability, and recoverability of digital data. These systems aim to address various challenges and risks associated with data management, including data loss, data corruption, hardware failures, and disasters.

Here are the key objectives of an online backup system:

- The objective of an online backup system project is to develop a reliable platform that allows users to back up their digital data (such as files, documents, photos) to servers over the internet.
- This ensures data preservation, disaster recovery, and easy access from server.
- The system should provide features like data backups, efficient storage utilization, and user-friendly interfaces.
- Ensure the ability to recover lost or corrupted data quickly and efficiently, minimizing downtime and disruptions.
- Ensure the ability to preserve project-related data for future reference.

Functional Requirements

1. User Authentication and Authorization:

- Users should be able to create accounts, log in, and manage their profiles.
- Different levels of access should be implemented based on user roles

2. Project Backup and Restore:

- Users should be able to initiate manual backups of their projects.
- Backups should be available such as files, documents, photos.

3. Storage Management:

- Storage space should be allocated for each user's backups.
- Storage quotas should be enforced to prevent abuse.

4. User-Friendly Interface:

- The user interface should be intuitive and easy to navigate.
- Clear instructions should be provided for each functionality.

5. Notification System:

- Users should receive notifications about the status of backups (successful, failed, etc.).

- Alerts should be sent for critical issues like failed backups or storage nearing capacity.

6. Documentation and Help:

- Provide user documentation and help resources (e.g., FAQs, tutorials).

7. Security Measures:

- Implement security features like input validation, authentication, authorization, and protection against common web vulnerabilities (e.g., XSS, CSRF, SQL injection).

System Architecture,Tools & Technology

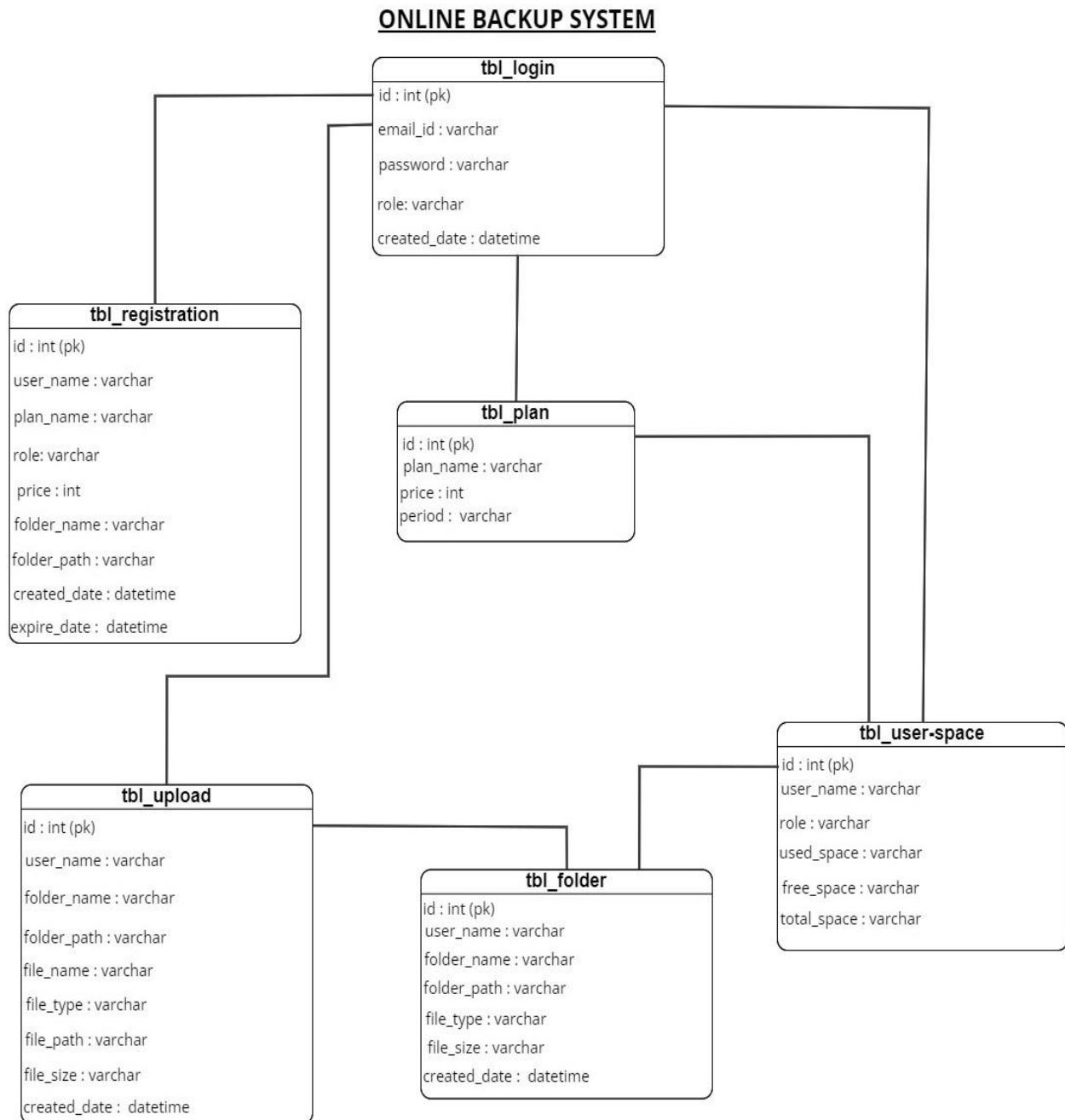
Hardware Requirements:

- ❖ Intel(R) Core(TM) i5-3360M CPU @ 2.80GHz 2.80 GHz
- ❖ Any version of Windows XP or later.
- ❖ Processor speed: 2.0 GHz
- ❖ RAM : 4GB
- ❖ Hard disk: 40GB to 80 GB

Software Requirements:

- ❖ Database : SQL
- ❖ Server : SQL Server
- ❖ Frontend : CSS,HTML,Bootstrap
- ❖ Scripting Language : C#
- ❖ Technology : .Net

Entity Relationship Diagram



Data Dictionary

Table name : Login

No.	Name	Datatype	Null	Constraint
1	Id	int	Primary key	Store user id
2	Email_id	Varchar(40)	Not null	Store email_id
3	Password	Varchar(10)	Not null	Store password

Table name : Registraion

No.	Name	Datatype	Null	Constraint
1	Id	int	Primary key	Store user id
2	User_name	Varchar(15)	Not null	Store user name
3	Plan_name	Varchar(max)	Not null	Store plan name
4	Role	Varchar(5)	Not null	Define role of user
5	price	int	Null	Price of role
6	Folder_name	Varchar(30)	Not null	Backup foldername
7	Folder_path	Varchar(max)	Not null	Backup folderpath
8	Role_time	Datetime	Not null	Time period of role
9	Expire_date	Datetime	Not null	Expiredate of role

Table name : Folder

No.	Name	Datatype	Null	Constraint
1	Id	int	Primary key	Store user id
2	User_name	Varchar(50)	Not null	Store user name
3	File_type	Varchar(50)	Null	Define type of file
4	File_size	Varchar(50)	Null	Size of file
5	Folder_name	Varchar(30)	Not null	Backup foldername
6	Folder_path	Varchar(max)	Not null	Backup folderpath
7	Created_date	Datetime	Not null	Time period of role

Table name : Fileupload

No.	Name	Datatype	Null	Constraint
1	Id	int	Primary key	Store user id
2	User_name	Varchar(50)	Not null	Store user name
3	File_type	Varchar(50)	Null	Define type of file
4	File_size	Varchar(50)	Null	Size of file
5	Folder_name	Varchar(30)	Not null	Backup foldername
6	Folder_path	Varchar(max)	Not null	Backup folderpath
7	File_name	Varchar(50)	Not null	Upload filename
8	Created_date	Datetime	Not null	Time period of role

Table name : plan

No.	Name	Datatype	Null	Constraint
1	Id	int	Primary key	Store user id
2	Plan_name	Varchar(5)	Not null	Define plan package
3	Price	int	Not null	Price of plan
4	Period	Varchar(10)	Not null	Timeperiod role of user

Table name : Storage

No.	Name	Datatype	Null	Constraint
1	Id	int	Primary key	Store user id
2	User_name	Varchar(20)	Not null	Store user name
3	role	Varchar(10)	Not null	Role type
4	Used_space	Varchar(MAX)	Not null	User storage
5	Free_space	Varchar(MAX)	Not null	Define free space
6	Total_space	Varchar(MAX)	Not null	Total space

Coding

File Uploading code :

```
protected void lb_upload_file_Click(object sender, EventArgs e)
{
    div_setting.Visible = false;
    if (div_upload.Visible == true)
    {
        div_upload.Visible = false;
        lb_upload_file.CssClass = "upload_current";

    }
    else
    {
        txt_folder_current.Text =
Request.QueryString["folder_name"].ToString();
        div_upload.Visible = true;
        lb_upload_file.CssClass = "upload_current";
        lb_dashbord.CssClass = "menu1";
        lb_download.CssClass = "menu4";
        lb_setting.CssClass = "menu5";

    }
}
```

File Downloading code :

```
protected void lb_download_Click(object sender, EventArgs e)
{
    lb_download.CssClass = "download_current";
    lb_upload_file.CssClass = "menu3";
    lb_dashbord.CssClass = "menu1";
}
```

```

lb_setting.CssClass = "menu5";
div_upload.Visible = false;
div_setting.Visible = false;

if (GridView1.Visible == true)
{
    using (ZipFile zip = new ZipFile())
    {
        zip.AlternateEncodingUsage = ZipOption.AsNecessary;
        zip.AddDirectoryByName(fn);
        int i=0;
        foreach (GridViewRow row in GridView1.Rows)
        {

            if ((row.FindControl("chkSelect") as CheckBox).Checked)
            {

                string filePath =
((Label)(row.FindControl("Label1"))).Text;
                f_path = Server.MapPath(filePath);
                zip.AddFile(f_path, fn);
                i = i + 1;
            }

        }
        if(i>1)
        {
            Response.Clear();
            Response.BufferOutput = false;
            string zipName = String.Format("Zip_{0}.zip",
DateTime.Now.ToString("yyyy-MMM-dd-HHmmss"));
            Response.ContentType = "application/zip";

```



```

        Response.AddHeader("content-disposition", "attachment;
filename=" + zipName);
        zip.Save(Response.OutputStream);
        Response.End();
    }
    else if(i==1)
    {
        string attch_filepath = f_path;
        string attch_filename = Path.GetFileName(attch_filepath);
        Response.ContentType = "application/octet-stream";
        Response.AppendHeader("Content-Disposition",
"attachment; filename=" + attch_filename);
        Response.TransmitFile(attch_filepath);
        Response.End();
    }
    else
    {
        Response.Write("<script>alert('Please Select any File or
folder')</script>");
    }
}
}
else
{
    using (ZipFile zip = new ZipFile())
    {
        int i=0;
        zip.AlternateEncodingUsage = ZipOption.AsNecessary;
        zip.AddDirectoryByName(fn);
        foreach (DataListItem item in DataList1.Items)
        {
            if ((item.FindControl("chkSelect") as CheckBox).Checked)
            {

```

```

        string filePath =
((Label)(item.FindControl("folder_name"))).Text;
        f_path = Server.MapPath(filePath);
        zip.AddFile(f_path, fn);
        i=i+1;
    }
}
if (i > 1)
{
    Response.Clear();
    Response.BufferOutput = false;
    string zipName = String.Format("Zip_{0}.zip",
DateTime.Now.ToString("yyyy-MMM-dd-HH:mm:ss"));
    Response.ContentType = "application/zip";
    Response.AddHeader("content-disposition", "attachment;
filename=" + zipName);
    zip.Save(Response.OutputStream);
    Response.End();
}
else if(i==1)
{
    string attch_filepath = f_path;
    string attch_filename = Path.GetFileName(attch_filepath);
    Response.ContentType = "application/octet-stream";
    Response.AppendHeader("Content-Disposition",
"attachment; filename=" + attch_filename);
    Response.TransmitFile(attch_filepath);
    Response.End();
}
else
{
    Response.Write("<script>alert('Please Select any File or
folder')</script>");
}
}
}

```

```

    }
  }
}

```

Plan Payment code :

```

if (Request.QueryString["folder_name"].ToString() != null)
{

}
else
{

    TimeZoneInfo INDIAN_ZONE =
    TimeZoneInfo.FindSystemTimeZoneById("India Standard Time");
    DateTime current_time =
    TimeZoneInfo.ConvertTimeFromUtc(DateTime.UtcNow, INDIAN_ZONE);

    DateTime expire_date = current_time.AddDays(30);

    string dt, role, user, pass;
    dt = DateTime.Now.ToString();
    role = "Paid";
    user = Session["Email"].ToString();
    pass = Session["Password"].ToString();
    string str = "Insert into tbl_login values('" + user.ToString() + "','"
+ pass.ToString() + "','" + role.ToString() + "','" + current_time.ToString()
+ "')";
    cmd = new SqlCommand(str, cn);

```

```

cmd.ExecuteNonQuery();

if (rb5gb.Checked == true)
{
    p = "5 GB";
    r = "100";
    f_space = 5;
}
if (rb10gb.Checked == true)
{
    p = "10 GB";
    r = "200";
    f_space = 10;
}
if (rb15gb.Checked == true)
{
    p = "15 GB";
    r = "300";
    f_space = 15;
}

string _allowedChars =
"0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
WXYZ";
Random randNum = new Random();
char[] chars = new char[8];
string folder_name = "";
int i;
int allowedCharCount = _allowedChars.Length;
for (i = 0; i < 8; i++)
{
    chars[i] = _allowedChars[(int)((_allowedChars.Length) *
randNum.NextDouble())];
}

```

```

        folder_name = folder_name + Convert.ToString(chars[i]);

    }
    string path = @"~/ " + folder_name;
    Directory.CreateDirectory(Server.MapPath(path));

    string str1 = "Insert into tbl_registration values('" +
user.ToString() + "','" + p.ToString() + "','" + role.ToString() + "','" +
r.ToString() + "','" + folder_name.ToString() + "','" + path.ToString() +
"', '" + dt.ToString() + "','" + expire_date.ToString() + "')";
    cmd1 = new SqlCommand(str1, cn);
    cmd1.ExecuteNonQuery();

    int use_space = 0;
    string str3 = "Insert into tbl_user_space values('" +
user.ToString() + "','" + role.ToString() + "','" + use_space.ToString() +
"', '" + f_space.ToString() + "','" + p.ToString() + "')";
    cmd2 = new SqlCommand(str3, cn);
    cmd2.ExecuteNonQuery();
    Response.Redirect("home.aspx");
}

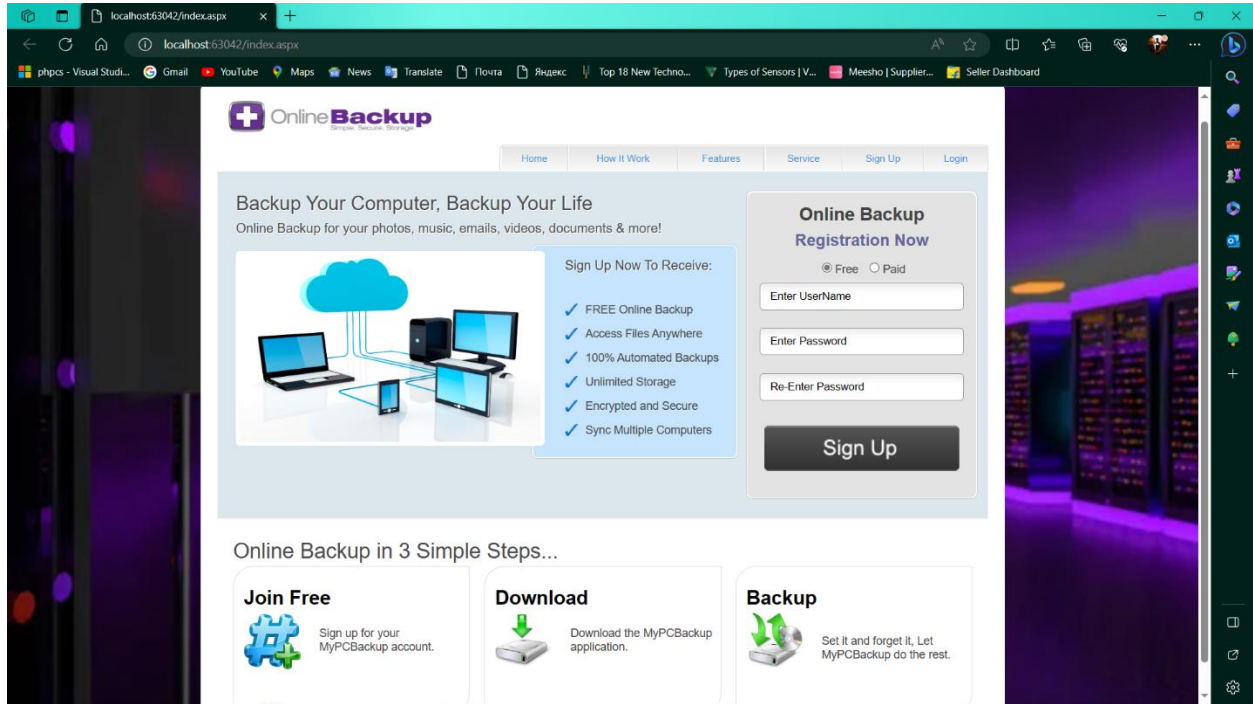
```

Testing

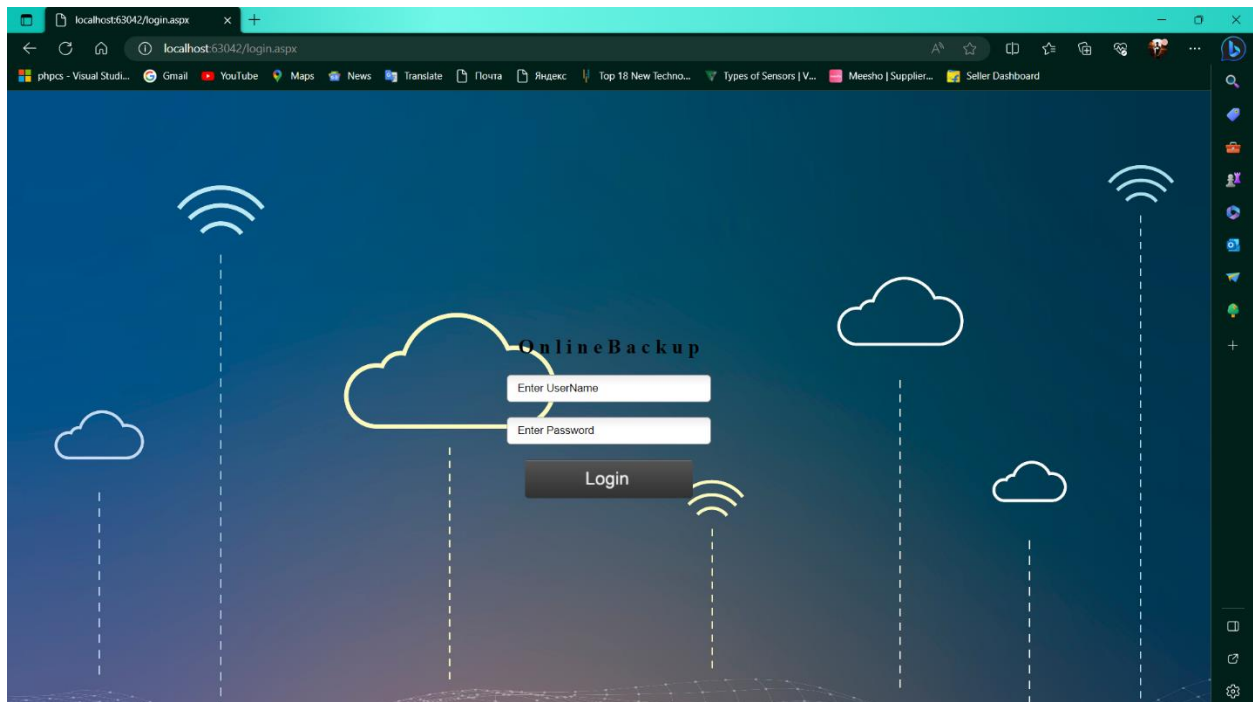
ONLINE BACKUP SYSTEM					
Test_ID	Test Page	Test scenario	Expected Result	Actual Result	Type of Case
TC_01	Index	1. Design of page 2. Page Randerling Time 3. Performance	Satisfy user Requirement and understandable Fast randerling page and time saving Better Performance	Some Information and Design Page Overloading Less performance	Enhancement
TC_02	Login	1.Usermname and Password 2.Required Field and Data	Username and Password must be valid Show Detailed and Instruction	Login into the website Data Override and write again	Bug
TC_03	Registration	1. Username 2. Password 3. Required Field and Data	Show Username instruction Show Password instruction(alphanumeric) Show Detailed and Instruction	Only Email username valid Number password is valid Data Override and write again	Enhancement Bug
TC_04	Home	1. Design of page 2. Submenu 3. File Upload 4. Management	User satisfaction and understandable Submenu Define with proper Name and Detail Define file type and size Proper Files structure	Some Information and Design Only Name and vertical Structure Can't upload big files(Video,movie) No more Details shown	Enhancement Bug Enhancement

Screen Layouts

Index Page :



Login page :



Registration Page :

OnlineBackup

Email ID

☐ Free ☒ Paid

Sign Up

Home Page :

Online Backup

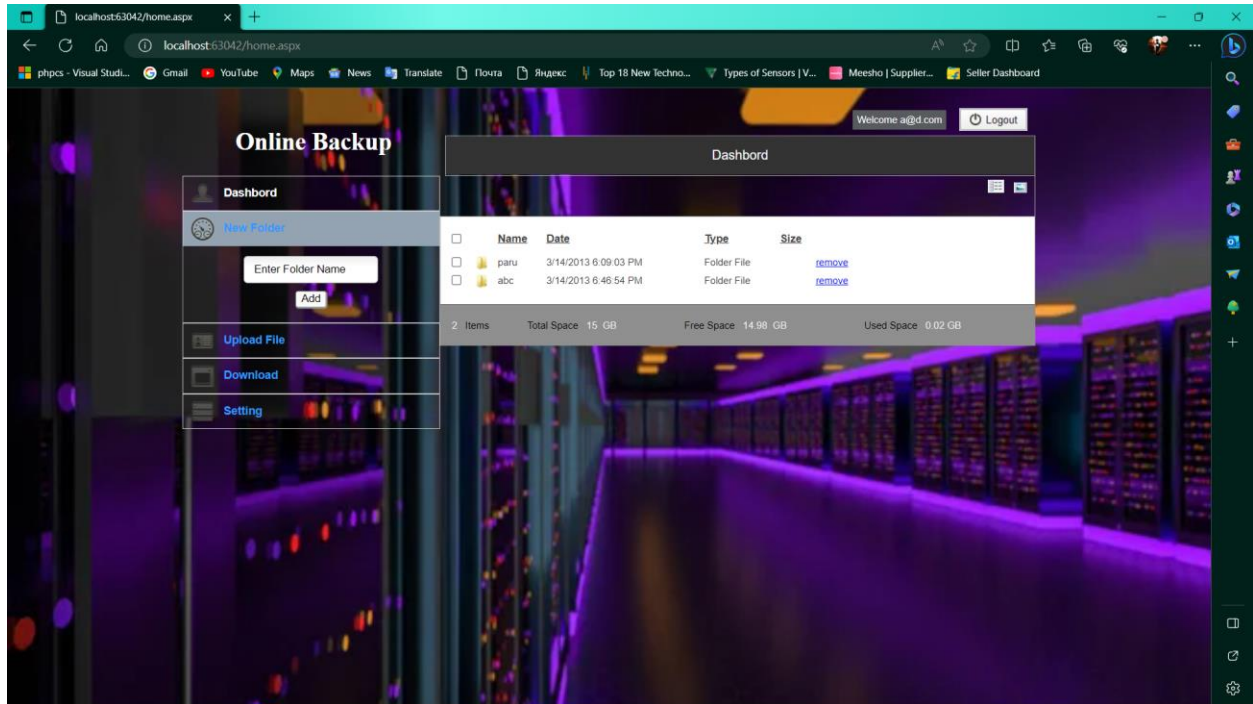
Welcome a@d.com Logout

Dashboard

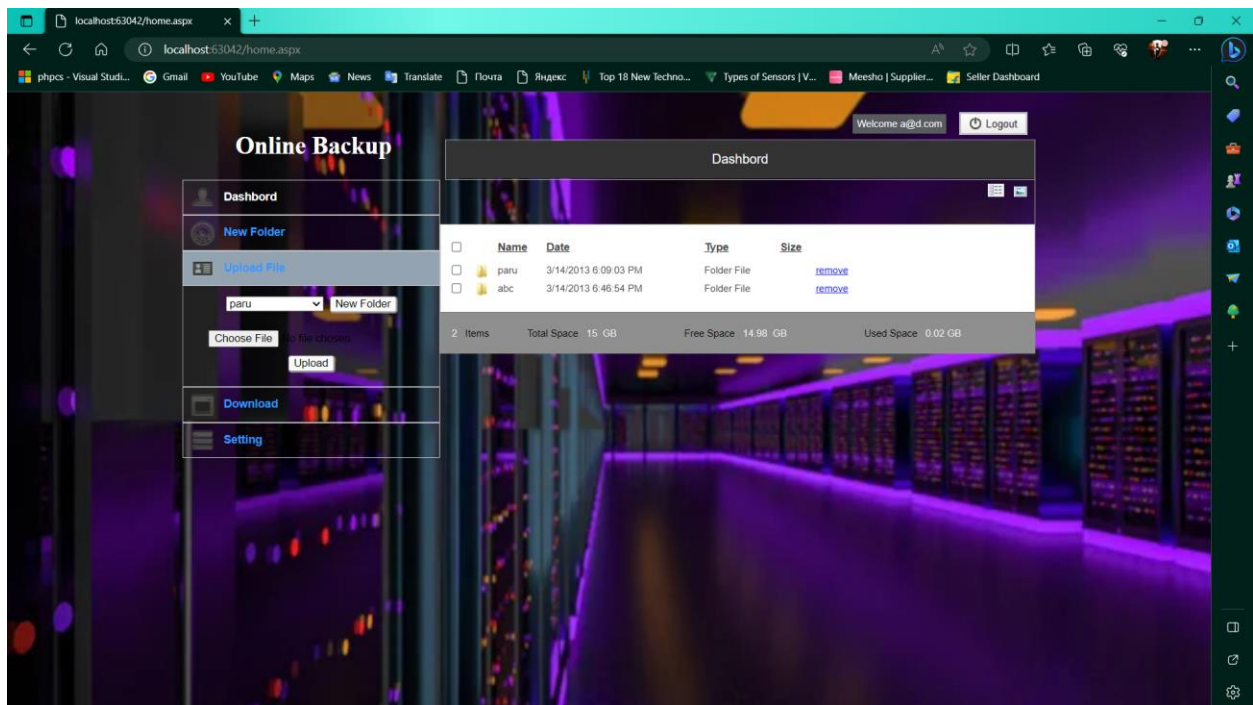
	Name	Date	Type	Size
<input type="checkbox"/>	paru	3/14/2013 6:09:03 PM	Folder File	remove
<input type="checkbox"/>	abc	3/14/2013 6:46:54 PM	Folder File	remove

2 Items Total Space: 15 GB Free Space: 14.99 GB Used Space: 0.02 GB

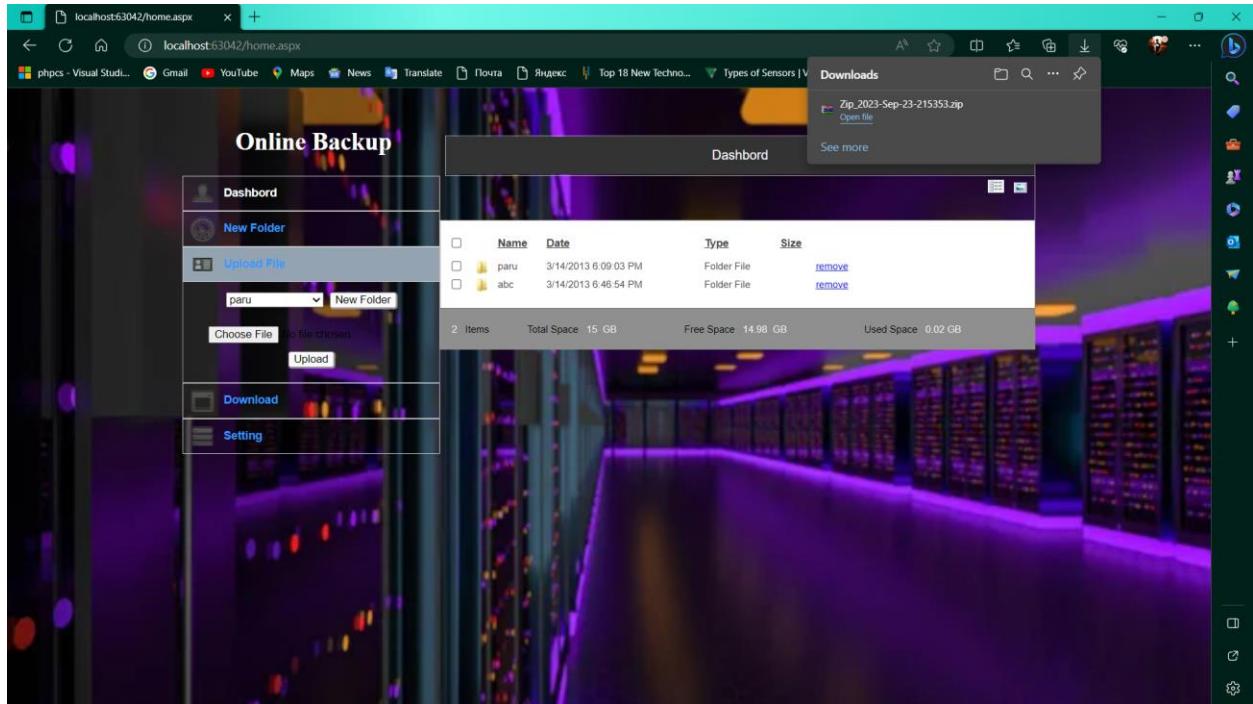
Folder Page :



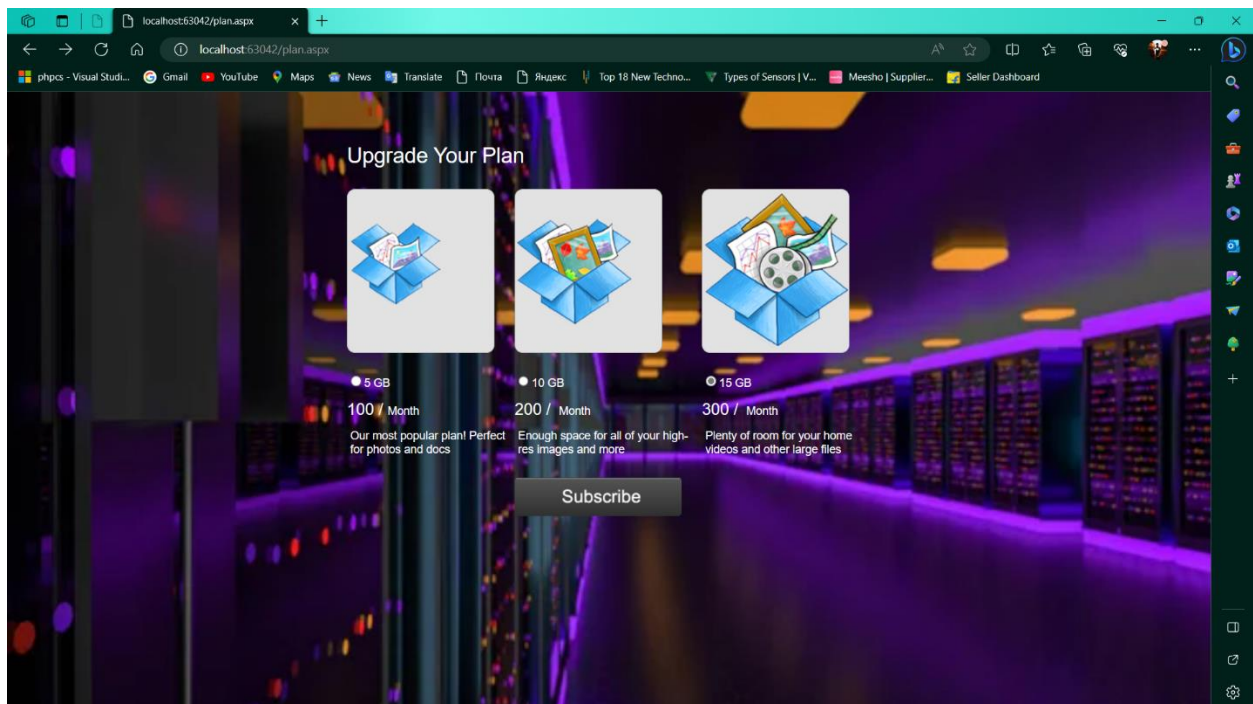
File Upload :



File Download :



Plan Page :



Future Enhancement

Enhancements for an online backup system can help improve its reliability, performance, and user experience. Here are some future enhancements to consider:

1. Artificial Intelligence (AI) and Machine Learning Integration
2. Blockchain for Data Integrity
3. Distributed and Decentralized Backups
4. Zero-Knowledge Encryption
5. Multi-Cloud Support
6. Continuous Data Protection
7. Hybrid Backup Solutions
8. Immutable Backups