

FINAL PROJECT BDAT 1001

Information Encoding Standards



Project done by:

Rimi Mondal

INTRODUCTION



+

○

●

BACKGROUND

- The project involves creating an ASP.NET Core web application that implements user authentication and authorization to protect user data.
- The application will allow users to create and manage contact information, which will be accessible based on the user's role.
- The application will have three security groups: registered users, managers, and administrators, each with specific access permissions to the data.



WORKFLOW



+

○

●

WORKFLOW

1

APP SETUP

- Set up the ASP.NET Core app by following the steps in the tutorial from the link provided.

2

AUTHORIZATION SYSTEM

- Implement the authorization system for the three security groups: registered users, managers, and administrators. Use policies and roles to restrict access to certain actions and views based on the user's role.

3

USER DATA FUNCTIONALITY

- Implement the functionality for registered users to view and edit/delete their own data. Use authentication to ensure that only the authorized user can access their own data.

4

MANAGER APPROVAL SYSTEM

- Implement the functionality for managers to approve or reject contact data. Use a database or file storage to store the approved contacts and ensure that only approved contacts are visible to users.

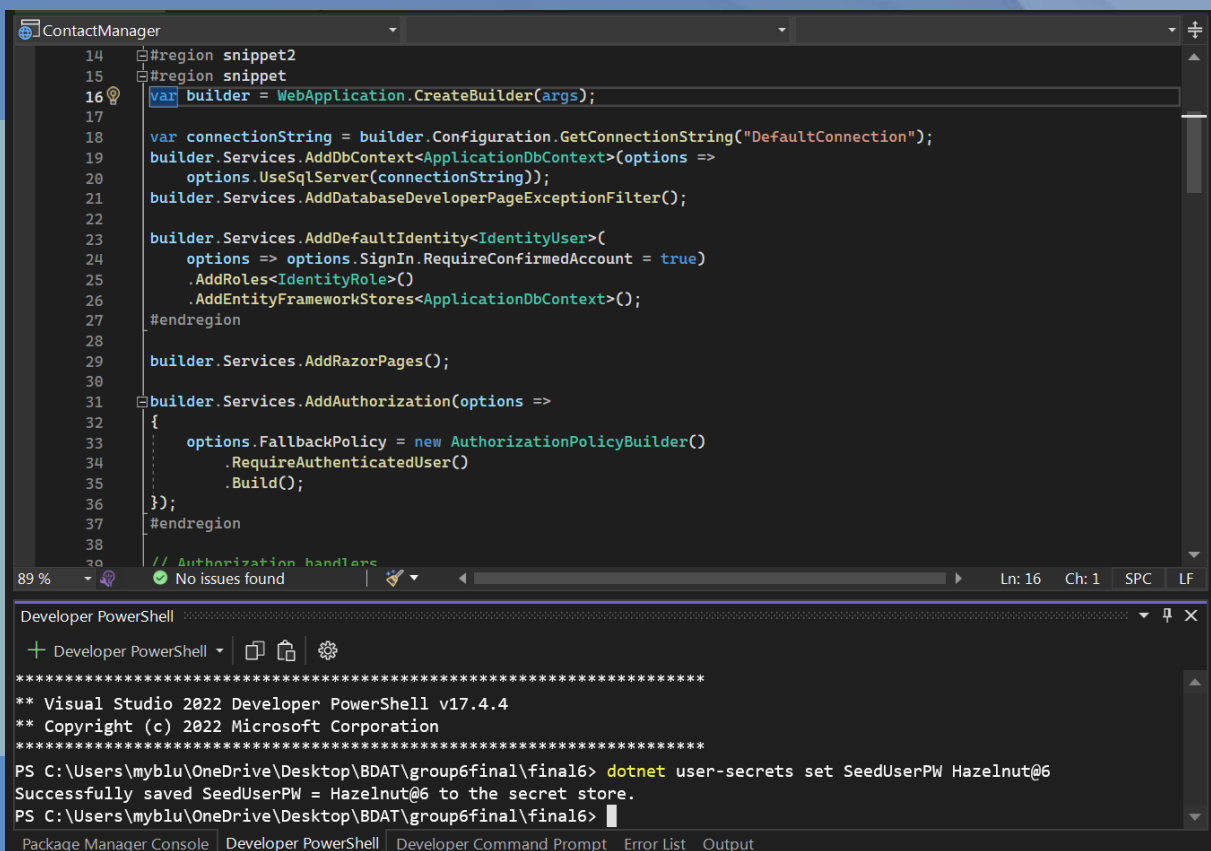
5

ADMINISTRATOR FUNCTIONALITY

- Implement the functionality for administrators to approve/reject and edit/delete any data. Ensure that administrators have the necessary permissions to access and modify any data in the system.

STEP 1: APP SETUP

The password for the user in the ASP.NET Core app can be set using the command-line interface (CLI).

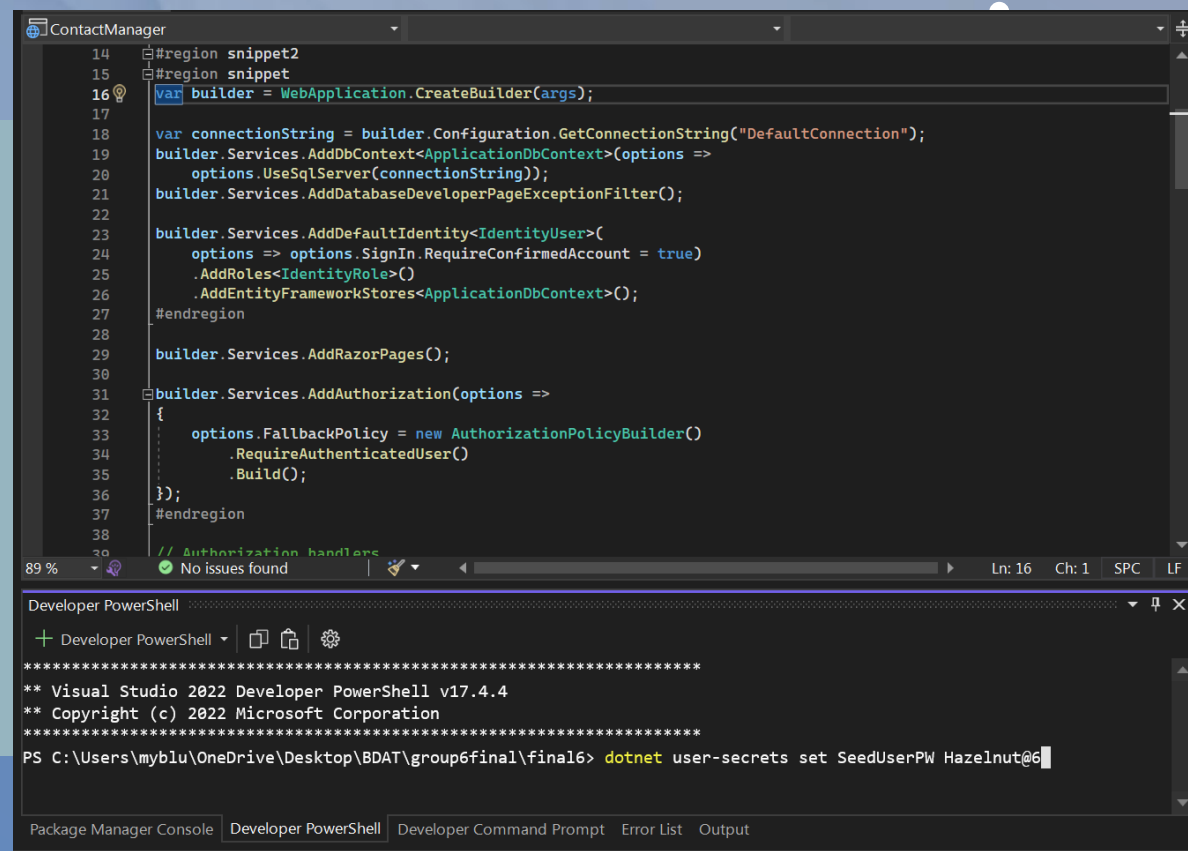


The screenshot shows the Visual Studio IDE with the `ContactManager` project open. The `Program.cs` file is displayed, showing the configuration of the application. The code includes setting up the database context, adding services, and configuring authentication. The `builder.Services.AddAuthorization` method is used to configure the authorization policy.

```
14 #region snippet2
15 #region snippet
16 var builder = WebApplication.CreateBuilder(args);
17
18 var connectionString = builder.Configuration.GetConnectionString("DefaultConnection");
19 builder.Services.AddDbContext<ApplicationDbContext>(options =>
20     options.UseSqlServer(connectionString));
21 builder.Services.AddDatabaseDeveloperPageExceptionFilter();
22
23 builder.Services.AddDefaultIdentity<IdentityUser>(
24     options => options.SignIn.RequireConfirmedAccount = true)
25     .AddRoles<IdentityRole>()
26     .AddEntityFrameworkStores<ApplicationDbContext>();
27 #endregion
28
29 builder.Services.AddRazorPages();
30
31 builder.Services.AddAuthorization(options =>
32 {
33     options.FallbackPolicy = new AuthorizationPolicyBuilder()
34         .RequireAuthenticatedUser()
35         .Build();
36 });
37 #endregion
38
39 // Authorization handlers
```

The Developer PowerShell terminal at the bottom shows the command to set the seed user password:

```
** Visual Studio 2022 Developer PowerShell v17.4.4
** Copyright (c) 2022 Microsoft Corporation
*****
PS C:\Users\myblu\OneDrive\Desktop\BDAT\group6final\final6> dotnet user-secrets set SeedUserPW Hazelnut@6
Successfully saved SeedUserPW = Hazelnut@6 to the secret store.
PS C:\Users\myblu\OneDrive\Desktop\BDAT\group6final\final6>
```



The screenshot shows the Visual Studio IDE with the `ContactManager` project open. The `Program.cs` file is displayed, showing the configuration of the application. The code includes setting up the database context, adding services, and configuring authentication. The `builder.Services.AddAuthorization` method is used to configure the authorization policy.

```
14 #region snippet2
15 #region snippet
16 var builder = WebApplication.CreateBuilder(args);
17
18 var connectionString = builder.Configuration.GetConnectionString("DefaultConnection");
19 builder.Services.AddDbContext<ApplicationDbContext>(options =>
20     options.UseSqlServer(connectionString));
21 builder.Services.AddDatabaseDeveloperPageExceptionFilter();
22
23 builder.Services.AddDefaultIdentity<IdentityUser>(
24     options => options.SignIn.RequireConfirmedAccount = true)
25     .AddRoles<IdentityRole>()
26     .AddEntityFrameworkStores<ApplicationDbContext>();
27 #endregion
28
29 builder.Services.AddRazorPages();
30
31 builder.Services.AddAuthorization(options =>
32 {
33     options.FallbackPolicy = new AuthorizationPolicyBuilder()
34         .RequireAuthenticatedUser()
35         .Build();
36 });
37 #endregion
38
39 // Authorization handlers
```

The Developer PowerShell terminal at the bottom shows the command to set the seed user password:

```
** Visual Studio 2022 Developer PowerShell v17.4.4
** Copyright (c) 2022 Microsoft Corporation
*****
PS C:\Users\myblu\OneDrive\Desktop\BDAT\group6final\final6> dotnet user-secrets set SeedUserPW Hazelnut@6
```


DEMONSTRATION

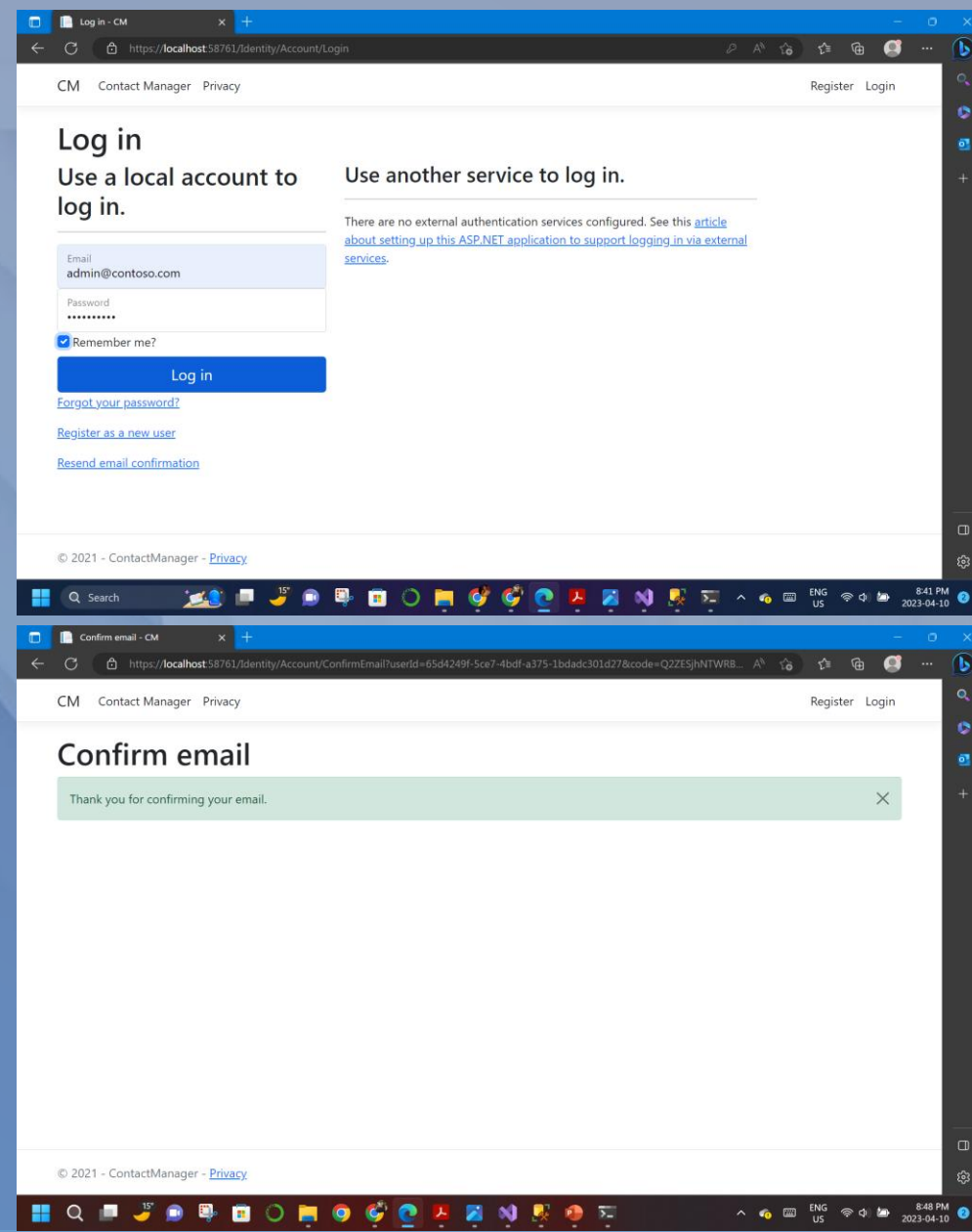
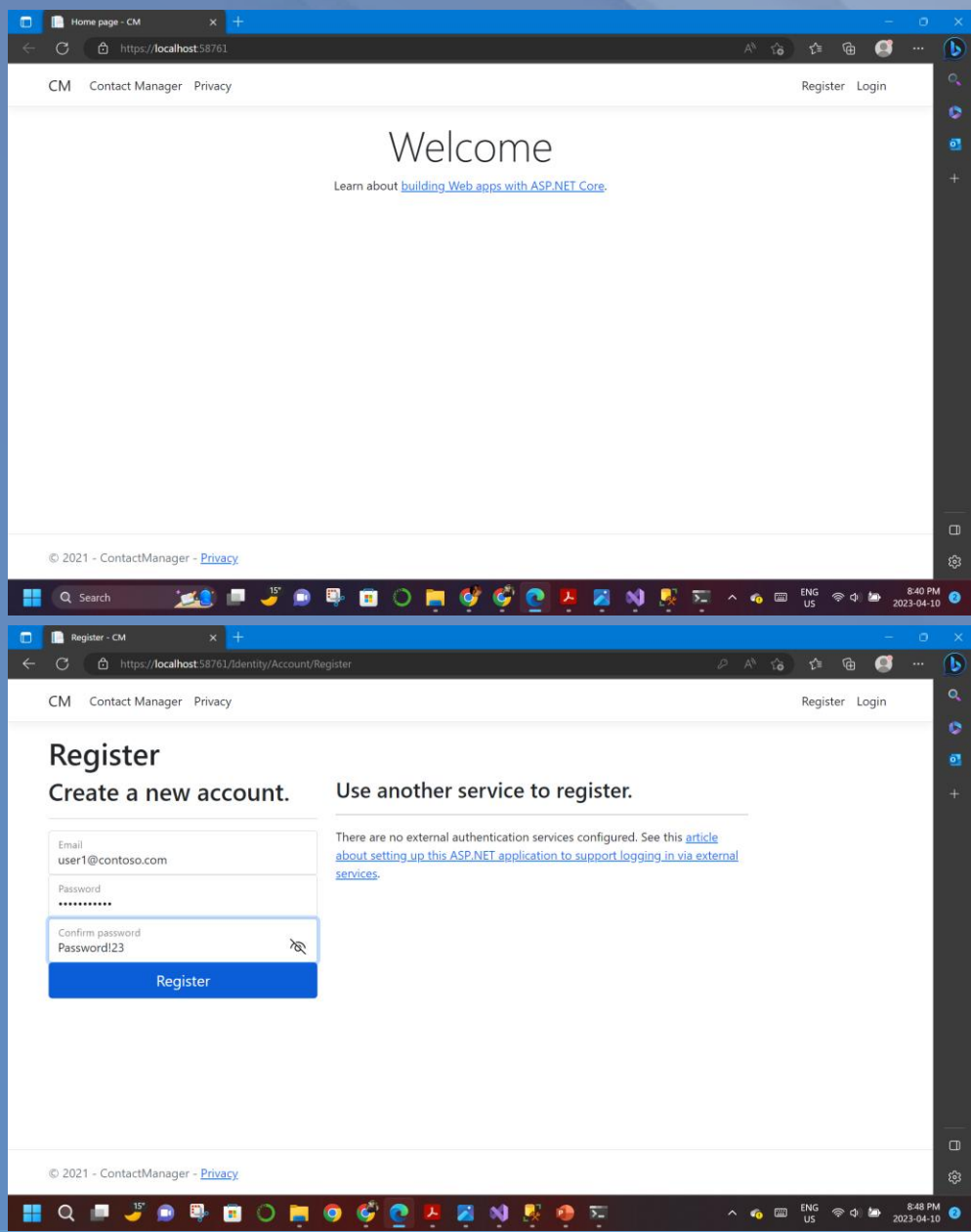


+

○

●

STEP 2: AUTHORIZATION SYSTEM



STEP 2: AUTHORIZATION SYSTEM

CM Contact Manager Privacy Hello admin@contoso.com! Logout

Index

[Create New](#)

Name	Address	City	State	Zip	Email	Status
Debra Garcia	1234 Main St	Redmond	WA	10999	debra@example.com	Approved Edit Details Delete
Thorsten Weinrich	5678 1st Ave W	Redmond	WA	10999	thorsten@example.com	Submitted Edit Details Delete
Yuhong Li	9012 State st	Redmond	WA	10999	yuhong@example.com	Rejected Edit Details Delete
Jon Orton	3456 Maple St	Redmond	WA	10999	jon@example.com	Submitted Edit Details Delete
Diliana Alexieva-Bosseva	7890 2nd Ave E	Redmond	WA	10999	diliana@example.com	Submitted Edit Details Delete
contact1	gugg	ggijhj	hjhjh	777777	yhfghghj@mail.com	Submitted Edit Details Delete
contact1byuser1	jhgjhghj	fhjfhf	fgjhfg	00008	hghjghghj@mail.com	Submitted Edit Details Delete
contact123byuser3	gyg	yfgyfgf	hgfhghg	7886	ghghjy@mail.com	Submitted Edit Details Delete

© 2021 - ContactManager - [Privacy](#)

CM Contact Manager Privacy Hello user1@contoso.com! Logout

Index

[Create New](#)

Name	Address	City	State	Zip	Email	Status
Debra Garcia	1234 Main St	Redmond	WA	10999	debra@example.com	Approved Details
contact1byuser1	jhgjhghj	fhjfhf	fgjhfg	00008	hghjghghj@mail.com	Submitted Edit Details Delete

© 2021 - ContactManager - [Privacy](#)

CM Contact Manager Privacy Hello manager@contoso.com! Logout

Index

[Create New](#)

Name	Address	City	State	Zip	Email	Status
Debra Garcia	1234 Main St	Redmond	WA	10999	debra@example.com	Approved Details
Thorsten Weinrich	5678 1st Ave W	Redmond	WA	10999	thorsten@example.com	Submitted Details
Yuhong Li	9012 State st	Redmond	WA	10999	yuhong@example.com	Rejected Details
Jon Orton	3456 Maple St	Redmond	WA	10999	jon@example.com	Submitted Details
Diliana Alexieva-Bosseva	7890 2nd Ave E	Redmond	WA	10999	diliana@example.com	Submitted Details
contact123byuser3	gyg	yfgyfgf	hgfhghg	7886	ghghjy@mail.com	Rejected Details
Rimi	15 Catherine Dr	Barrie	ON	L4N0Y5	mailrimi2662@gmail.com	Approved Details
Rahul	Homeless	No	No	00000	mailrahul@gmail.com	Rejected Details
Ashutosh	28 Seline Dr	Barrie	ON	L4M3Y7	mailashul@gmail.com	Approved Details

© 2021 - ContactManager - [Privacy](#)

CM Contact Manager Privacy Hello user3@contoso.com! Logout

Index

[Create New](#)

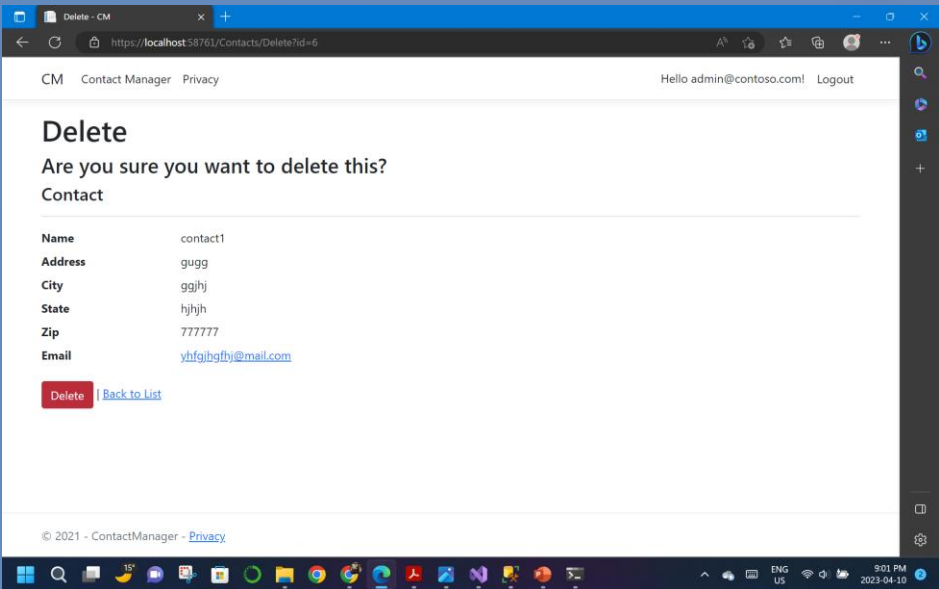
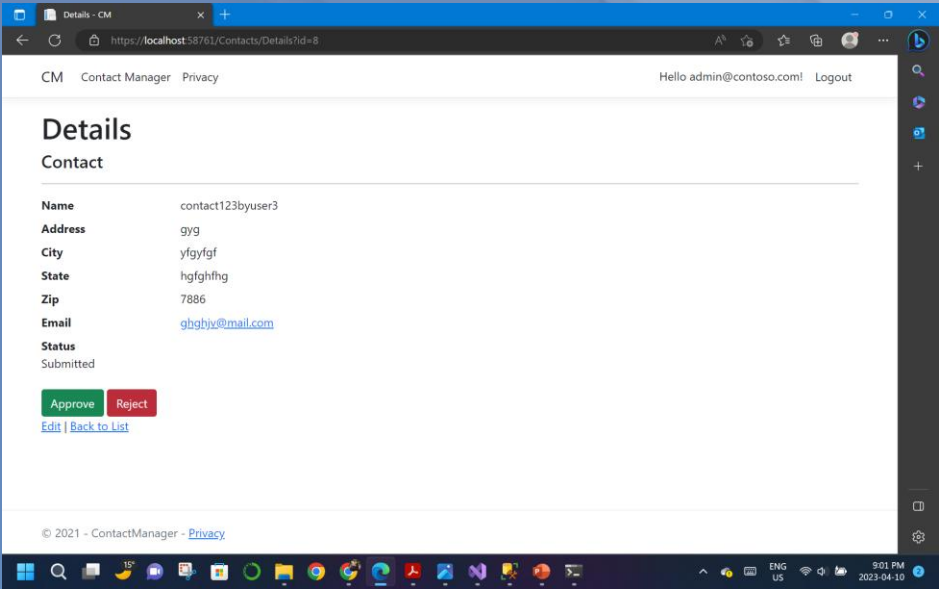
Name	Address	City	State	Zip	Email	Status
Debra Garcia	1234 Main St	Redmond	WA	10999	debra@example.com	Approved Details

© 2021 - ContactManager - [Privacy](#)

STEP 3: USER DATA FUNCTIONALITY

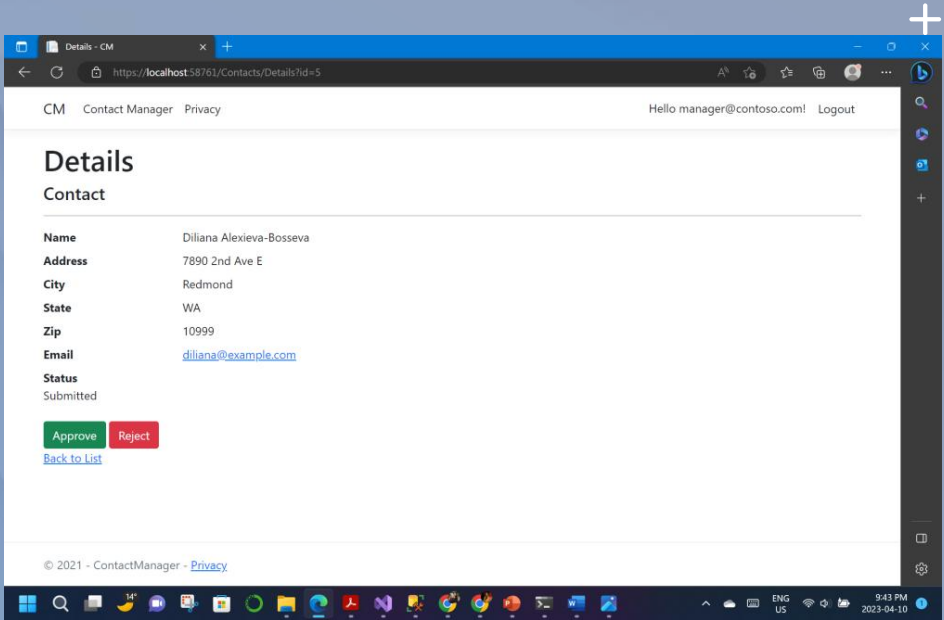
A
D
M
I
N

A
C
C
E
S
S



M
A
N
A
G
E
R

A
C
C
E
S
S



STEP 3: USER DATA FUNCTIONALITY

Create - CM

https://localhost:58761/Contacts/Create

CM Contact Manager Privacy Hello user1@contoso.com! Logout

Create Contact

Name
Rimi

Address
15 Catherine Dr

City
Barrie

State
ON

Zip
L4N0Y5

Email
mailrimi2662@gmail.com

Create

[Back to List](#)

© 2021 - ContactManager - Privacy

Create - CM

https://localhost:58761/Contacts/Create

CM Contact Manager Privacy Hello user1@contoso.com! Logout

Create Contact

Name
Rahul

Address
Homeless

City
No

State
No

Zip
00000

Email
mailrahul@gmail.com

Create

[Back to List](#)

© 2021 - ContactManager - Privacy

Create - CM

https://localhost:58761/Contacts/Create

CM Contact Manager Privacy Hello user1@contoso.com! Logout

Create Contact

Name
Ashutosh

Address
28 Seline Dr

City
Barrie

State
ON

Zip
L4M3Y7

Email
mailashul@gmail.com

Create

[Back to List](#)

© 2021 - ContactManager - Privacy

Index - CM

https://localhost:58761/Contacts

CM Contact Manager Privacy Hello user1@contoso.com! Logout

Index

[Create New](#)

Name	Address	City	State	Zip	Email	Status
Debra Garcia	1234 Main St	Redmond	WA	10999	debra@example.com	Approved Details
Rimi	15 Catherine Dr	Barrie	ON	L4N0Y5	mailrimi2662@gmail.com	Submitted Edit Details Delete
Rahul	Homeless	No	No	00000	mailrahul@gmail.com	Submitted Edit Details Delete
Ashutosh	28 Seline Dr	Barrie	ON	L4M3Y7	mailashul@gmail.com	Submitted Edit Details Delete

© 2021 - ContactManager - Privacy

STEP 4: MANAGER APPROVAL SYSTEM

Index - CM

https://localhost:58761/Contacts

CM Contact Manager Privacy

Hello manager@contoso.com! Logout

Index

[Create New](#)

Name	Address	City	State	Zip	Email	Status
Debra Garcia	1234 Main St	Redmond	WA	10999	debra@example.com	Approved Details
Thorsten Weinrich	5678 1st Ave W	Redmond	WA	10999	thorsten@example.com	Submitted Details
Yuhong Li	9012 State st	Redmond	WA	10999	yuhong@example.com	Rejected Details
Jon Orton	3456 Maple St	Redmond	WA	10999	jon@example.com	Submitted Details
Diliana Alexieva-Bosseva	7890 2nd Ave E	Redmond	WA	10999	diliana@example.com	Submitted Details
contact123byuser3	gyg	yfygyfgf	hgfhghfg	7886	gbghjv@mail.com	Rejected Details
Rimi	15 Catherine Dr	Barrie	ON	L4N0Y5	mailrim2662@gmail.com	Approved Details
Rahul	Homeless	No	No	00000	mailrahul@gmail.com	Rejected Details
Ashutosh	28 Seline Dr	Barrie	ON	L4M3Y7	mailashul@gmail.com	Approved Details

© 2021 - ContactManager - [Privacy](#)

Details - CM

https://localhost:58761/Contacts/Details?id=5

CM Contact Manager Privacy

Hello manager@contoso.com! Logout

Details

Contact

Name

Diliana Alexieva-Bosseva

Address

7890 2nd Ave E

City

Redmond

State

WA

Zip

10999

Email

[diliana@example.com](#)

Status

Submitted

Approve

Reject

[Back to List](#)

© 2021 - ContactManager - [Privacy](#)

Details - CM

https://localhost:58761/Contacts/Details?id=10

CM Contact Manager Privacy

Hello manager@contoso.com! Logout

Details

Contact

Name

Rahul

Address

Homeless

City

No

State

No

Zip

00000

Email

[mailrahul@gmail.com](#)

Status

Rejected

Approve

[Back to List](#)

© 2021 - ContactManager - [Privacy](#)

STEP 5: ADMINISTRATOR FUNCTIONALITY

Details - CM

https://localhost:58761/Contacts/Details?id=9

CM Contact Manager Privacy Hello admin@contoso.com! Logout

Details

Contact

Name

Address

City

State

Zip

Email

Status

Rimi

15 Catherine Dr

Barrie

ON

L4N0Y5

mailrimi2662@gmail.com

Submitted

Approve

Reject

[Edit](#)

[Back to List](#)

© 2021 - ContactManager - Privacy

Index - CM

https://localhost:58761/Contacts

CM Contact Manager Privacy Hello user1@contoso.com! Logout

Index

[Create New](#)

Name	Address	City	State	Zip	Email	Status
Debra Garcia	1234 Main St	Redmond	WA	10999	debra@example.com	Approved Details
Rimi	15 Catherine Dr	Barrie	ON	L4N0Y5	mailrimi2662@gmail.com	Approved Edit Details Delete
Rahul	Homeless	No	No	00000	mailrahul@gmail.com	Rejected Edit Details Delete
Ashutosh	28 Seline Dr	Barrie	ON	L4M3Y7	mailashul@gmail.com	Approved Edit Details Delete

© 2021 - ContactManager - Privacy

Index - CM

https://localhost:58761/Contacts

CM Contact Manager Privacy Hello admin@contoso.com! Logout

Index

[Create New](#)

Name	Address	City	State	Zip	Email	Status
Debra Garcia	1234 Main St	Redmond	WA	10999	debra@example.com	Approved Edit Details Delete
Thorsten Weinrich	5678 1st Ave W	Redmond	WA	10999	thorsten@example.com	Submitted Edit Details Delete
Yuhong Li	9012 State st	Redmond	WA	10999	yuhong@example.com	Rejected Edit Details Delete
Jon Orton	3456 Maple St	Redmond	WA	10999	jon@example.com	Submitted Edit Details Delete
Diliana Alexieva-Bosseva	7890 2nd Ave E	Redmond	WA	10999	diliana@example.com	Submitted Edit Details Delete
contact123byuser3	gyg	yfgyfgf	hgfhghg	7886	gghghiv@mail.com	Rejected Edit Details Delete
Rimi	15 Catherine Dr	Barrie	ON	L4N0Y5	mailrimi2662@gmail.com	Approved Edit Details Delete
Rahul	Homeless	No	No	00000	mailrahul@gmail.com	Rejected Edit Details Delete
Ashutosh	28 Seline Dr	Barrie	ON	L4M3Y7	mailashul@gmail.com	Approved Edit Details Delete

© 2021 - ContactManager - Privacy

Index - CM

https://localhost:58761/Contacts

CM Contact Manager Privacy Hello user3@contoso.com! Logout

Index

[Create New](#)

Name	Address	City	State	Zip	Email	Status
Debra Garcia	1234 Main St	Redmond	WA	10999	debra@example.com	Approved Details
contact123byuser3	gyg	yfgyfgf	hgfhghg	7886	gghghiv@mail.com	Rejected Edit Details Delete
Rimi	15 Catherine Dr	Barrie	ON	L4N0Y5	mailrimi2662@gmail.com	Approved Details
Ashutosh	28 Seline Dr	Barrie	ON	L4M3Y7	mailashul@gmail.com	Approved Details

© 2021 - ContactManager - Privacy

PART 2



SECURITY TECHNOLOGIES RECOMMENDATIONS

1. How can we transfer personal data securely within their network?

To securely transfer personal data within a network, there are several security technologies that can be considered. One such technology is a Virtual Private Network (VPN), which provides a secure encrypted connection between two points. This technology ensures that the data is protected from unauthorized access and interception.

Another technology that can be used is Secure Socket Layer (SSL) or Transport Layer Security (TLS), which encrypts data while it is in transit between the client and the server. Additionally, Secure File Transfer Protocol (SFTP) can be used to transfer files securely between servers by using encryption to protect the data while in transit. Encrypted email is another way to secure data in transit, which can be achieved by using technologies like Pretty Good Privacy (PGP) or Secure/Multipurpose Internet Mail Extensions (S/MIME).

It is important to note that the selection of the most suitable technology will depend on the specific requirements of the company and the nature of the data that needs to be transferred. A thorough risk assessment should be conducted to determine the most appropriate security measures to implement.

SECURITY TECHNOLOGIES RECOMMENDATIONS

2. What security protocol is best for transferring personal files?

Our recommendation is for the company to first assess the risks involved in transferring personal files. Identifying specific security needs will be essential before evaluating which security protocol to use. While Secure File Transfer Protocol (SFTP) is a popular choice for securely transferring files over the internet, it may not be the best solution for every project. The choice of protocol will depend on the specific requirements of the company.

For instance, if the company requires a more user-friendly solution that is accessible via a web interface, HTTPS or WebDAV may be more appropriate. Conversely, if the company requires additional security features like two-factor authentication or end-to-end encryption, FTPS or AS2 may be more suitable.

The selection of the protocol will also depend on other factors such as the sensitivity of the data being transferred, the available resources and infrastructure, and the specific needs of the company. Ultimately, a comprehensive risk assessment followed by an informed protocol selection process is key to ensuring that personal files are transferred securely.

SECURITY TECHNOLOGIES RECOMMENDATIONS

3. Can we encode and encrypt images?

As a consultant for this project, we can confirm that images can be both encoded and encrypted. Encoding involves transforming data into a specific format that can be easily interpreted by a computer, while encryption involves the process of transforming data so that it can only be deciphered by someone with the appropriate decryption key. There are several encoding and encryption methods available for images, each with their own set of advantages and disadvantages.

One common method of encoding and encrypting images is using the Advanced Encryption Standard (AES). AES is a widely used symmetric-key encryption algorithm that can encrypt data in blocks of 128 bits. Another popular method is using the Secure Sockets Layer (SSL) or Transport Layer Security (TLS) protocols to encrypt images while they are in transit between a client and a server.

It is worth noting that encoding techniques like Base64 can also be used to convert binary data into a text format that can be transmitted easily via email or messaging applications. However, encoding techniques like Base64 do not offer encryption and should not be used as a substitute for encryption when transmitting sensitive images.

In conclusion, encoding and encryption methods can be used to secure images depending on the specific requirements of the project. As a consultant, we would recommend conducting a thorough risk assessment to determine the most appropriate method for encoding and encrypting images based on the level of risk involved.

SECURITY TECHNOLOGIES RECOMMENDATIONS

4. Our database cannot be moved from the site and we need to be able to access it externally using a secure API. Can you explain the architecture of a secure API?

As a consultant on this project, we can explain the architecture of a secure API in a way that is easy to understand. Think of an API as a gatekeeper that allows different applications to talk to each other. A secure API architecture involves building a strong and secure gate that can only be accessed by authorized users.

To achieve this, we need to make sure that only the right users are allowed through the gate. This is done by using authentication protocols that verify the identity of the user. Once the user is authenticated, we need to make sure that they are only allowed to access the resources or perform the actions they are authorized to. This is done using access control mechanisms that ensure that users are authorized based on their role or other attributes.

In addition to controlling access, we also need to make sure that the data being transmitted through the gate is secure. This is done by encrypting the data using protocols like HTTPS or SSL, which ensure that the data is only readable by authorized users.

Finally, we need to keep an eye on what's going on at the gate to make sure that everything is secure. This is done by using security tools like firewalls and intrusion detection systems that monitor and log API activities. This allows us to quickly identify and respond to any security threats.

Overall, a secure API architecture involves building a strong and secure gate that only allows authorized users to access the resources they need. This is done by using authentication and access control mechanisms, encrypting data, and monitoring and logging API activities to ensure security.

SECURITY TECHNOLOGIES RECOMMENDATIONS

5. Can you recommend a secure framework for coding an API?

We can recommend a secure framework for coding an API that can help ensure the security of your data. There are several frameworks available that can help you develop secure APIs. One popular framework is ASP.NET Core, which has built-in security features such as authentication and authorization middleware. It also supports industry-standard security protocols like OAuth 2.0 and JWT, which can be used to protect your API against unauthorized access.

Another popular framework is Spring, which provides similar security features to ASP.NET Core, including support for OAuth 2.0 and JWT. Spring also has a range of security features such as secure password hashing, access control, and secure session management.

If you require the highest level of security for your API, then the Open Web Application Security Project (OWASP) is a good place to start. They offer a list of secure coding frameworks and libraries, which can help you protect your API against common security vulnerabilities such as injection attacks and cross-site scripting (XSS).

Overall, the choice of framework depends on your specific needs and requirements. We would recommend conducting a thorough analysis of your security requirements and choosing a framework that has the right set of security features to meet your needs.

SECURITY TECHNOLOGIES RECOMMENDATIONS

6. What data interchange format should we use while transferring data between locations?

As a consultant, we would suggest using a data interchange format that balances between security, interoperability, and efficiency while transferring data between locations. Based on our experience, JSON and XML are two popular formats that offer such benefits.

JSON is lightweight, flexible, and commonly used for transferring data between different applications. It's easy to read and write and is supported by most modern programming languages. Besides, JSON can be easily converted to and from other data formats, which makes it highly interoperable.

On the other hand, XML is another widely used data interchange format that provides a reasonable balance between security and interoperability. It's more verbose than JSON, but it has inbuilt support for advanced features such as data validation and encryption, making it a good choice for sensitive data. Additionally, XML is also supported by most modern programming languages, which makes it highly interoperable.

However, the choice of data interchange format should be made based on your specific needs and requirements. If you're looking for a format that is lightweight and flexible and easy to work with, JSON may be a good choice. On the other hand, if you need advanced features like data validation and encryption, XML may be a better fit. We would suggest evaluating your needs and requirements carefully and choosing a format that is best suited for your use case.

SECURITY TECHNOLOGIES RECOMMENDATIONS

7. How should we store our data in our many locations?

○

Our recommendation would be to use a distributed database system to store your data across multiple locations. This approach can offer a range of benefits, such as improved performance, increased availability, and better fault tolerance.

One option is a **sharded database system**, which splits the data across multiple servers or data centers. This can help to optimize performance by minimizing the amount of data that needs to be transferred between locations. Another option is a **replicated database system**, which stores multiple copies of the same data in different locations. This can enhance availability and fault tolerance since if one location goes down, the data can still be accessed from another location.

When implementing a distributed database system, it's crucial to ensure that the data is stored securely and that proper access controls are in place. You should also consider the cost and complexity of managing and maintaining the system, as this can be a significant factor in the long run.

Ultimately, the choice of distributed database system should be based on your specific needs and requirements. It's important to carefully evaluate the available options and choose the one that best fits your use case.

SECURITY TECHNOLOGIES RECOMMENDATIONS

8. What are the ethical concerns related to the transmission of personal data?

○

As a consultant, we would like to highlight that the transmission of personal data can raise serious ethical concerns. One of the main concerns is the risk of data breaches, where personal data can be accessed or stolen by unauthorized parties. This can cause significant harm to individuals, ranging from identity theft to financial losses and reputational damage.

Another ethical issue is the potential misuse of personal data, where it can be used for purposes other than what it was collected for. For instance, if personal data is collected for research purposes, it should not be used for commercial purposes without obtaining the explicit consent of the individual.

Moreover, individuals have the right to know what personal data is being collected about them and how it is being used. They should also have control over their data and the ability to access, correct, or delete it when necessary. Additionally, there are ethical concerns related to collecting personal data from vulnerable populations such as children or individuals with disabilities. Their data should be collected and used with utmost care, ensuring that their privacy and rights are protected.

Organizations need to take a responsible and ethical approach when it comes to the collection, transmission, and use of personal data. This requires implementing appropriate security measures, obtaining informed consent from individuals, and respecting their rights and privacy. By doing so, organizations can establish trust with their customers and stakeholders and demonstrate their commitment to ethical business practices.



THANK YOU

+

•

○