- **Project WordPro API Dashboard**
- The project was created using Microsoft Visual Studio Community 2019 on Version 16.11.23 and utilizes Microsoft .NET Framework Version 4.8.09037 and
- several nugget packages, including Microsoft.AspNetCore.Http, Microsoft.AspNetCore.Authentication, AspNetCore.Authentication.Cookies, Microsoft.EntityFrameworkCore, System.Linq, System.Threading.Tasks, AspNetCoreHero.ToastNotification, HttpContext.Session, Identity, System.Net.Http, System.Security.Authentication, and CookieAuthenticationDefaults.
- The project fetches data from a third-party API and displays it on the MVC core view. INotyfService and TempData are used for login validation. The project also features an action to display the user's log activity, including their username, IP address, login time, and visited action.
- The project uses a SQL Server database to save user profile data, and a data table script is utilized for exporting table data to various formats. Other features include printing, copying to the clipboard, and pagination.

- ASP.NET Core is a popular web development framework used to build web applications, web APIs, and mobile backbends.

- It is an open-source, cross-platform framework developed by Microsoft, designed to create modern web applications that can run on Windows, Linux.

- **ASP.NET Core MVC is** the web application framework part of ASP.NET Core, which focuses on building web applications using the Model-View-Controller (MVC) pattern. On the other hand,

- ASP.NET Core Web API is a lightweight, highly scalable, and fast framework used to build RESTful web services, which can be consumed by different client-side applications.

- **As an ASP.NET Core MVC and Web API developer,** involve developing, testing, and deploying web applications and web APIs. I'm responsible for designing and implementing the user interface using Razor views, creating and maintaining RESTful web APIs, implementing security measures, database modeling, and data access using Entity Framework Core.

- work with front-end technologies such as HTML, CSS, and JavaScript to create responsive and user-friendly web applications.

- Additionally, you would need to write and execute unit tests and integration tests to ensure the quality and functionality of the code. You may also be required to collaborate with other developers, designers, and stakeholders to deliver projects on time and within budget.
-
-
-
-
-
-

- 1. .**NET full-stack developer, goals**

- .NET developer looking to become a .NET full-stack developer, my long-term goal would be to expand my skill set to become proficient in all aspects of web development, from front-end to back-end.

- To achieve this goal, I would focus on developing expertise in front-end technologies such as HTML, CSS, and JavaScript, as well as popular front-end frameworks like React, Angular, and Vue. I would also aim to become skilled in back-end technologies such as C#, ASP.NET, and SQL Server, and become familiar with cloud platforms like Azure and AWS.

- To accomplish this, I would seek out learning resources such as online tutorials, courses, and certifications, as well as work on real-world projects and collaborate with other developers to gain practical experience.

- Overall, my long-term goal as a .NET developer looking to become a full-stack developer would be to continually improve my skills and knowledge to become a well-rounded and versatile developer capable of delivering high-quality web applications from start to finish.

2. **What programming languages and frameworks are you proficient in?**

As a .NET developer, I am proficient in C#, ASP.NET, and SQL Server database. I also have experience with front-end technologies such as HTML, CSS, and JavaScript. In addition, I have experience with popular frameworks like MVC, MVC Core, and Web API.

3. **How do you stay current with the latest technology trends and developments?**

I stay up-to-date with the latest technology trends and developments by attending conferences, following industry blogs, and participating in online communities such as Stack Overflow and GitHub. I also like to experiment with new technologies and frameworks in my personal projects to gain hands-on experience.

5. **How do you approach testing and debugging your code?**

I approach testing and debugging my code by writing automated unit tests and integration tests. I also use debugging tools such as Visual Studio debugger to step through the code and identify issues. I follow a systematic approach to troubleshooting and log all errors to aid in debugging.

6. **What is your experience with software design patterns and principles?**

I have extensive experience working with software design patterns and principles such as SOLID, DRY, and KISS. I use design patterns such as the Repository pattern, Singleton pattern, and Factory pattern in my projects to improve code maintainability and scalability.

1. **Can you walk me through a project you have worked on from start to finish?**

Sure, I worked on a web application using ASP.NET MVC framework, CSS, HTML, and SQL Server database. The project involved creating a web-based platform for a company that manages customer feedback. We started by gathering requirements from stakeholders and creating wireframes for the user interface. Then we developed the front-end using HTML, CSS, and JavaScript. We also developed the back-end using ASP.NET MVC and SQL Server for data storage. We used Entity Framework to manage database connections and migrations. We also created a RESTful API using ASP.NET Web API for the mobile app that would connect to the system. We used Git for version control and followed Agile methodology throughout the project. After completing the development, we deployed the application on a server and conducted user acceptance testing. Finally, we launched the application and provided maintenance and support to the client.

2. **Have you ever had to work with legacy code? How did you approach this?**

Yes, I have worked with legacy code in the past. The first thing I did was to read through the code to understand its structure and logic. I also documented any issues or bugs that I found. Then, I created a plan to refactor and modernize the code. I would start by writing unit tests to ensure that any changes I made did not break the existing functionality. I also tried to separate the concerns of the code by breaking it down into smaller, more manageable parts. Finally, I would test the changes thoroughly before pushing the code to the repository.

2. **Have you ever had to work with legacy code? How did you approach this?**

Yes, I have worked with legacy code in the past. The first thing I did was to read through the code to understand its structure and logic. I also documented any issues or bugs that I found. Then, I created a plan to refactor and modernize the code. I would start by writing unit tests to ensure that any changes I made did not break the existing functionality. I also tried to separate the concerns of the code by breaking it down into smaller, more manageable parts. Finally, I would test the changes thoroughly before pushing the code to the repository.

3. **How do you prioritize tasks and manage your workload as a developer?** As a developer, I prioritize tasks based on their level of urgency and importance. I use a project management tool to create a list of tasks and assign them to specific deadlines. I also communicate with my team members to ensure that we are all on the same page regarding priorities. I break down larger tasks into smaller, more manageable ones to help me stay on track. I also set aside time for debugging and testing to ensure that the code is of high quality before submitting it.
4. **Can you explain how you have used version control systems such as Git in your projects**?

As a .NET developer, I use Git as the primary version control system in my projects. I create a repository for each project, and then I commit and push changes to the repository as I work on the project. I also create branches for specific features or bug fixes, and merge them back to the main branch once they are completed and tested.

1. **Have you worked in an Agile development environment?** Can you give an example of how you contributed to a sprint? Yes, I have worked in an Agile development environment. In one project, I contributed to a sprint by developing a new feature for the web application using the MVC framework. I collaborated with the product owner to understand the requirements and with other developers to ensure the code was well-structured and maintainable. I also participated in daily stand-up meetings to provide updates on my progress and discuss any blockers.

2. **How do you approach collaborating with other developers on a project?**

When collaborating with other developers, I first make sure we have a shared understanding of the project goals and requirements. I use communication tools like Slack, Microsoft Teams, and email to stay in touch with my teammates and provide regular updates on my progress. I also make sure to write clean and well-documented code that others can easily understand and maintain.

1. **How do you handle and respond to feedback from code reviews and peer reviews**?

As a professional developer, I always welcome feedback on my work, and I see code reviews and peer reviews as an opportunity to learn and improve. When I receive feedback, I carefully review and consider it, and I try to understand the reviewer's perspective. I ask questions if I'm unsure about anything, and I make changes to my code as needed. I also strive to give constructive feedback when I'm reviewing someone else's code, and I focus on providing specific suggestions for improvement rather than just pointing out problems.

2. What do you see as the biggest challenges facing software development in the next 5-10 years?

There are several significant challenges facing software development in the next few years. One of the biggest is keeping up with rapidly changing technologies and frameworks. As new technologies emerge and evolve, developers must continually learn and adapt to stay relevant. Another challenge is ensuring security and privacy in the face of increasing cyber threats. As more and more data is stored online, it's essential to take steps to protect that data from malicious actors. Finally, there is the challenge of scaling and optimizing software to handle increasing amounts of data and traffic. As businesses and organizations rely more heavily on software systems, developers must find ways to make those systems faster, more efficient,