

Веб-инфраструктура Blazor

Научный руководитель – Магомедов Ислам Арбиевич

Тахаев Усам Русланович

Студент (бакалавр)

ФГБОУ ВО "Чеченский государственный университет Грозный, Россия

E-mail: takhaev.usam@bk.ru

Первая реализация проекта Blazor была датирована 5 декабря, 2017 года. В дальнейшем эта технология активно развивалась компанией Microsoft. Сами представители Microsoft изначально не верили в эту идею: слабая оптимизация, малая поддержка платформ, сырость технологии SPA. В конечном итоге, Microsoft, выпустили полноценную релиз версию данной технологии в сентябре 2019 под лицензией Apache-2.0. Их выпуск был громким, их настрой и упорность показали результаты, с выходом новой .NET Core 3.1, который включает в себя Blazor, популярность компании возросла. Blazor-это компонентоориентированный (компонентом в Blazor, называют любой элемент пользовательского интерфейса, страница, элемент ввода данных и.т.д.) фреймворк позволяющий строить архитектуру своего клиентского приложения на языке программирования C#, тем самым позволяя полностью заменить JavaScript. Данный фреймворк работает в бинарном формате WebAssembly. Он основан на .NET core, который работает с razor разметкой, которая в свою очередь использует html\css. Полная интеграция Blazor с JavaScript (не исключая открытые веб-стандарты) дают возможность разработчикам не задумываться о поддержке платформ. Ведь Blazor работает во всех современных веб-браузерах, включая мобильные браузеры. Он позволяет работать с JavaScript, с помощью абстракции IJSRuntime. Благодаря такой тесной дружбе с JavaScript, не исключена возможность использовать большую, проверенную временем, экосистему мира JavaScript.

У технологии Blazor, есть две реализации:

1. Blazor server- это модель при использовании которой обработка «frontend» части все ещё выполняется на сервере, а именно рендерится вебсайт в режиме реального времени. Это происходит путем обновления компонентов с помощью SignalR, а именно интерфейса, обработка события и т.п..

2. Blazor WebAssembly- это модель уже является полностью частью frontend, позволяя выполняться коду непосредственно на стороне клиента. При первой загрузки сервиса с этой моделью, все нужные зависимости, и среда разработки .NET скачиваются на клиентский браузер. Сам процесс выполнения, следующий: C# компилируется в IL код (IL или CIL — это промежуточный код, в который транслируется C#, F# и другие языки на платформе .NET), а далее из IL преобразовываются команды WebAssembly, которые выполняются в одном процессе, т.е. все команды по работе с интерфейсом, обработкой события и т.п

Источники и литература

- 1) Robert C. Martin Series. Clean code. United States at Courier in Stoughton, Massachusetts. 2008.
- 2) Джеффи Рихтер. CLR via C#. Программирование на платформе Microsoft .NET FRAMEWORK 4.5 на языке C#. 4-е изд.: Питер, 2014 – 896 с. – (Серия мастер-класс)
- 3) Марк Лутц. Python. Карманный справочник. ВИЛЬЯМС 2018г. 5-е издание