Software Platforms Research

Individual Project: FestivalConnect

Name: Lucas Jacobs
Class: S-A-RB06
PCN: 490692
Student number: 4607368
Technical teachers: Felipe Ebert, Bartosz Paszkowski
Semester coach: Gerard Elbers

Table of Contents

Glossary

Term	Definition
client-server interactions in websites	Client-server interactions in websites involve communication between a user's browser (client) and a remote server, exchanging requests and responses.
sever-side website programming	Server-side website programming involves writing code that runs on the server to generate dynamic content for web pages.
Programming Paradigms	A programming paradigm is an approach to solving problems using some programming language or also we can say it is a method to solve a problem using tools and techniques that are available to us following some approach.
Aspect-Oriented Programming	programming paradigm that enables the modularization of crosscutting concerns in software applications
XSS (Cross-Site Scripting)	Cross-site scripting (XSS) is a common vulnerability that allows hackers to inject malicious scripts into web pages viewed by other users.
CSRF (Cross-Site Request Forgery)	CSRF attacks occur when malicious websites trick users into taking unauthorized actions on other sites where they are authenticated. Django offers built-in protection against CSRF attacks using CSRF tokens.
Open-Source	Open source software is software with source code that anyone can inspect, modify, and enhance.
Single-page application	A website or web application that dynamically rewrites a current web page with new data from the web server.
Virtual DOM	A programming concept where an ideal, or “virtual”, representation of a UI is kept in memory and synced with the “real” DOM by a library such as ReactDOM.

Introduction

FestivalConnect needs to have a proper back-end framework to work with. It needs to be able to support various aspects such as maintainable, easy-to-use, scalable, etc. The important parts of the functional and non-functional requirements of FestivalConnect are listed in the (Jacobs, 2024, Software Requirements Specification), this needs to be taken into account when deciding the software frameworks.

What frameworks can fit FestivalConnect?

Before we dive into specific frameworks, first there needs to be a set some requirements that determine what best fits the FestivalConnect application. For picking a framework we need to make a trade-off including a range of application development, development speed, manageability, and flexibility, this also further specifies what needs are important for FestivalConnect (Jacobs, 2024, Software Requirements Specification). the following things:

1. **Range of Applicability**: The backend framework, will be used for handling all kinds of things such as authentication and community business logic within FestivalConnect. For FestivalConnect it is important to have easy interaction with the frontend components, to try and improve the user experience. Moreover, the framework needs to also handle the potential growth in user base, support scalability, and keeping mind that it needs to be able to be extendible for when in the feature we for instance want a payment method for premium accounts.
2. **Development Speed**: For this project, sprints contain three weeks, meaning that FestivalConnect must have proper deadlines to keep an eye on. Also, development speed is important when there needs to be an update rollout, especially when it is peak festival season. Therefore, a rapid development pace is of a need, so the framework needs to offer built-in features, and documentation of the framework to have an easy guideline to follow. Furthermore, a community that supports the framework is important with also libraries that can accelerate the process of developing.
3. **Manageability & Flexibility**: In FestivalConnect it should be easy to maintain and flexible so that we can have performance optimization and scalability during peak usage in the festival season. Also, the framework needs to support internationalization and localization for the user, since a lot of different people from other countries go to the same festival, such as Tomorrowland having 400000 people from over 200 countries. (Celebrate 20 years of Tomorrowland in 2024 with the new theme 'LIFE', 2023)

Furthermore, frameworks can also be divided into multiple different kinds. We need to find a middle that can support the three things mentioned above. See the following:

- **Low/No Code Framework**: This is used for low-management projects and is a fast-paced speed framework to develop. It has predefined points to customize and beyond this, it is impossible. So these frameworks do not fit FestivalConnect, since this is going to be a big project with also a future look for extendibility and it needs to be flexible with this, which this framework is not. Also, right now it is not a problem that it has problems for collaborative development, but in the future this project needs to be hand-over and ready to be used by another development team, making this framework not a good pick.
- **High-level Full-stack Frameworks:** With this framework, we already have an entire infrastructure to build full-scale enterprise applications faster. This can be a fit for FestivalConnect since it can make a custom application with typical enterprise requirements. This can be taken into account when comparing the framework to see if one of these frameworks can outperform the needs of other frameworks.
- **Narrow-focused Frameworks:** These frameworks are made to develop with a rise of the level of abstraction, it will be a good mix for development speed with a high-level flexibility. Therefore, these frameworks can be a good fit for FestivalConnect, supporting potential architecture structures and having a good fit for custom back ends.
- **"No Frameworks" Approach:** This will give the widest approach to manageability & Flexibility. This comes with a slow way of developing enterprise applications. This is not a fit for FestivalConnect since we need to have high-speed development which will be hard

considering all the things to keep into account such as handling multiple languages, module architecture, etc. (Stukalov, 2018)

Back-end Framework Choices

Now that we know what framework categories we can look at, it is a good way to pick one high-level framework, and three narrow-focused frameworks to look at their pros and cons. Considering the high level, it is good to look at Django frameworks. The web framework is highly recommended for real-time apps and can make a big difference if you want an enterprise app with consistent updates and rapid delivery, which can fit the FestivalConnect requirements. (Binmile, 2021)

When the Django framework outperforms the narrow-focussed frameworks, we can then further look for the best high-level full-stack frameworks, but since narrow-focussed has more benefits we now only pick one high-level framework.

For the narrow-focussed frameworks, the frameworks .Net API Core, Spring framework, and Jakarta EE can be a good fit. To quickly look at the frameworks, they have a supporting community, flexible to architecture and the developer of FestivalConnect has experience in the coding languages used in these frameworks. Of course, we go further in-depth to finally choose the best-fitting framework.

Front-end Frameworks Choices

To also research the front-end framework, we also want to pick a high-level framework, and two narrow-focused frameworks to look for the pros and cons. For the high-level framework, we want to look for Blazor, which is part of the .NET framework. It can for example fit with the backend when it is also C#, supporting productivity in app development and maintenance, but also the use of the same libraries in the same existing framework. (ASP.NET Core Blazor, 2024)

For the narrow-focussed frameworks, the framework React and Angular are a good fit to look at. They both are flexible and offer the right functionality when it comes to the scaling of this project. FestivalConnect is an enterprise application, which React and Angular both support. (React vs Angular: What to Choose for Enterprise Application in 2024, 2024)

What enterprise software platform in regards to the back-end best fits this project?

For the back end, several criteria are important. Therefore, we are going over the following criteria, and comparing the frameworks with each other, to eventually evaluate them in an evaluation matrix. First, let's introduce all the frameworks and what they are mostly used for.

- **Django**: This framework is a high-level Python Framework, that lets you rapidly develop websites that are both secure and maintainable. You need to have a base knowledge of server-side website programming, and some standard mechanisms of client-server interactions in websites. (Django introduction, n.d.)
- **.NET Core API**: Framework that is used for building server-side applications that can run on multiple platforms (Windows, Linux, Mac). It supports languages like C#, VB.NET, and F# for creating web services and API's. (.NET Core vs .NET Framework: How to Pick a .NET Runtime for an Application, 2024)
- **Spring:** The framework simplifies enterprise Java development, supporting built-in features such as Dependency injection and aspect-oriented programming (AOP). (Wickramasinghe, 2021)
- **Jakarta EE:** Previously named Java Enterprise Edition (Java EE). It extends the Java Standard Edition (Java SE), with technologies, standards, and APIs, trying to address and fix challenges in enterprise applications, supporting industry technologies such as cloud-native, microservices, and web apps. (What is Jakarta EE?, n.d.)

Do the frameworks support the expected load?

As mentioned in the introduction, there is a (Jacobs, 2024, Software Specification Document), that will state how many concurrent users FestivalConnect needs to handle. It came down to having a handled user system of fifty million concurrent users.

Django

It is a powerful web framework that can handle large volumes of traffic, but to do this, it requires careful planning and optimization to have optimal performance. So you need to take into account that every little thing can count, and extra milliseconds, when this adds on to the millions of requests, it can lead to a lot more consumption of resources. (Godina, 2020), (Adelli, 2023)

.NET Core

This framework can be one of the best for enterprise applications because it is open source (everyone can use it for free), it runs on multiple platforms, and it is fast and scalable because it is built for performance that can handle large-scale web applications. (Panchal, 2022)

Spring

In spring, Java's performance is highly optimized. With Spring it will enhance this. It can make low-latency, and high throughput, making it suitable for enterprise applications. (Lebbos, 2023)
Furthermore, a big priority of Spring is performance, and is constantly updated based on feedback from the community. Spring will be a lightweight solution, and will improve performance, in terms of start-up time, memory usage, and runtime cpu. (SYER, 2018)

Jakarta EE

The framework can be used to make scalable applications. It can scale because it uses automating much of the necessary infrastructure work, and it distributes the load across hardware. Giving it the possibility to handle such high user numbers.

Conclusion

For each framework, the same thing comes back, and it is important to look at the details since this can significantly decrease the performance when users increase. They are all useable for FestivalConnect, but Sprint Boot is least recommended.

Do the frameworks have a cost of the product or any licensing?

For developing and for future maintenance of the product, it is important if there can be costs for using the frameworks.

Django
It is a free, open-source framework that can be used for web applications, both for personal and business use. (Pricing for Django, 2023)

.NET Core
This is also free to use. Furthermore, there are options that Microsoft offers and recommends when you have a business of more than five people working there ranging between a license of $45-250 per user/per month. Also for large organizations with 500-plus users, there are more licenses to choose from. (Buy Visual Studio, n.d.)

Spring
Free to use.

Jakarta EE
Free to use.

Conclusion
All options are a good fit for FestivalConnect. .Net Core can become a bit expensive when more developers are needed or it will be handed over to another person/business. But for this project, these are not relevant and are only add-ons that can be extra helpful.

Does the platform have an active community and support options to fall back on?

When looking at a framework, you want to know if it has an active community. It can be beneficial when you want to know certain best practices or you're stuck during development. Also, frequent updates are crucial when keeping up with the best practices of for instance security and preventing it from being outdated.

Django
Django has official documentation about the functionalities in the framework. There is also some knowledge on Django Forums, Stackoverflow, and knowledge on GitHub to find certain tutorials or information. It also has still frequent updates and is supported.

.**Net Core**
It has a lot of official documentation covering all the aspects of .NET Core development. There are also a lot of questions on Stack Overflow, and tutorials on YouTube to learn C#. But also a lot of other technologies are covered on how to properly implement in .NET. The community is really big on the communities on both the forums and StackOverflow.

Spring
Spring also has a great and active community, with official documentation on the Spring website, with also ranging topics about covering several technologies in regards to the Spring framework. The official forums of Spring are not supported anymore, but this only makes channels such as Stackoverflow more popular for searching for answers.

Jakarta EE
For this framework, there are some basic tutorials on how to set up certain things for a web service in the framework in the official documentation. Also, it has a Jakarta EE Community, with
Conclusion
All the platforms offer good guidance on their official documents, their communities are all strong. To conclude, Microsoft offers the most resources and guidance throughout the whole framework, with great guidance and enough references.

Does the platform have enough security options to properly secure FestivalConnect?

For FestivalConnect, security is a crucial part because it needs to preserve its business reputation. Therefore, making sure the whole application is safe against intruders.

Django

This framework has powerful security features, including password hashing with great industry standards, secure session management, and authentication protocols. Also with session data encryption, we will make sure to prevent unauthorized access, and authentication will make sure of controlled user access. Also, it has built-in protection against XSS and CSRF attacks. Also, Django's ORM mitigates SQL injection risks by sanitizing user inputs. With all these security features, Django can be a good choice for developing a secure web application. (Isah, 2023)

.NET Core

.NET has a lot of security features and libraries that make it an attractive platform for enterprise development, with examples such as code access security and Windows authentication. Also, it has really strong password rules and prevents brute-force hacking attempts or even attacks with the use of predictable or dictionary words as passwords. (Cuthbert, 2022)

Additionally, the strong community and widely expanded documentation also contribute to the maintenance of a secure development environment.

Spring

Makes use of Spring Security, which is a framework that provides authentication, authorization, and protection against common attacks. It is a standard for securing Spring-based applications. It is really easy to implement and easy to extend on custom requirements. (Spring Security, n.d.)

Jakarta EE

Uses the Jakarta EE Security API for handling security. It is used by specifying the dependency of the Maven for the Security API. For authentication, it can use Jakarta Authentication. It is easy to implement, by using the annotations on top of for instance a class or property. You can add custom security on the same level, meaning that the security is the same for the deployment on the server. (baeldung, 2023)

Conclusion

For security, you must keep it well-maintained. All platforms have great solutions for security vulnerabilities. Due to the fact that .NET Core also has a strong community and guidelines, while also being a big company making trust for upcoming support even stronger, the favor goes to .NET Core

How easy is the ease of use of the platforms?

The developers need to have the feeling and confidence to properly develop, making the ease of use important. The developer of FestivalConnect has experience with C# and JAVA when it comes to building websites, with some experience in Python for writing research documents for data science courses. Also, the Spring and .NET are familiar and there is no experience in both Django and Jakarta EE.

Django
Learning and mastering Django takes time and practice. When knowing Python and familiar with industry standards it is within three weeks doable to master. Moreover, rushing the process needs to be avoided at all times, since it is all about understanding the principles. (Learn Django, n.d.)
To further note, Django also provides shortcuts and tools to create complex applications quickly and easily (Murimi, 2024)

.NET Core
The framework is really easy to learn and use, even when you are not familiar with it. The platforms use a range of tools and libraries to simplify the development process, making it easier for developers to write and maintain code. Also with the great community support, it is easy to find answers to your questions. (Kamal, 2023)

Spring
Spring is easy to use because it simplifies development with features like fast setup without an XML configuration, customizable tools, easy to understand with the use of annotations, and therefore letting developers focus on core project goals. (what's the difference between Spring and Spring Boot in Java?, 2024)

Jakarta EE
It can be quite complex to work with, especially when you have never used the framework. It has regulations that are strict and can result in a more difficult learning curve. (LLC, 2023)

Conclusion
When starting this project, we want smooth implementation with rapid development. Since I don't have some experience with Jakarta EE and Django, it can become complex to set it up. Even though the frameworks all have their shortcuts and offer complex things to narrow down, Spring and .NET Core will be the easiest to use due to both experience and the way they are built.

SWOT-Analysis

To give a clear overview of what the strengths, weaknesses, opportunities, and threads are for these frameworks, I will show a SWOT analysis to get a final overview of choosing the framework that fits FestivalConnect.

Django

	Beneficial	Harmful
Internal	Strengths 1. Mature, widely used 2. A lot of documentation, good community 3. Built-in security features	Weaknesses 1. Learning curve 2. Careful optimization for handling larger volumes of traffic.
External	Opportunities 1. Continuous improvement and updates from the community. 2. New technologies integration and libraries	Threads 1. A lot of alternatives 2. Potential scalability challenges without proper optimization.

.NET Core

	Beneficial	Harmful
Internal	Strengths 1. Cross-platform compatibility, supporting multiple operating systems. 2. Fast and scalable performance, suitable for enterprise applications. 3. Strong security features. 4. Great community and large base of documentation	Weaknesses 1. Limited to languages like C# or F#. 2. Can bring license costs for certain enterprise features.
External	Opportunities 1. Growing community and ecosystem 2. Integration with cloud services and modern development practices	Threads 1. Changes in the future plan of Microsoft can affect .NET Core.

Spring

	Beneficial	Harmful
Internal	Strengths 1. Simplified Java Development with built-in features such as dependency injection (DI). 2. Active community and enough documentation. 3. Customizable and supports various architectural patterns.	Weaknesses 1. Configuration can become messy, especially for beginners. 2. Potential performance issues DI.
External	Opportunities 1. Integration with cloud-native.	Threads 1. Competition to other Java frameworks and alternative languages. 2. Changes in Java ecosystem or enterprise development trends.

Jakarta EE

	Beneficial	Harmful
Internal	Strengths 1. Great APIs and technology. 2. Scalable architecture and security support. 3. Active Community.	Weaknesses 1. Complex learning curve, for beginners. 2. Limited documentation.
External	Opportunities 1. Integration with industry technology such as microservices and cloud-native. 2. Adoption of Jakarta EE by more organizations.	Threads 1. Competition from other platforms. 2. Lack of updates may impact adoption.

Evaluation matrix

The evaluation matrix will be based on several important points that have been discussed in the previously asked questions. Each point will weigh how important that specific part is in comparison to the FestivalConnect application.

Evaluation Matrix	**Weight**	**Django**	**.NET Core API**	**Spring**	**Jakarta EE**
Scalability	0.20	++	++	++	++
Performance	0.20	+	++	+	+
Cost/License	0.20	++	+	++	++
Support & Community	0.20	+	++	++	++
Personal Skills	0.15	-	++	++	-
Security	0.25	+	++	+	+
Flexibility (ease of use)	0.20	+	++	++	+
Total:	N/A	1,65	2,70	2,35	1,95

- = Not preferred (0 * Weight score)
+ = General (1 * Weight score)
++ = Preference (2 * Weight score)

Conclusion

To conclude the choice of the back-end, all the frameworks can fit in enterprise software. Also since Django did not score the highest in the evaluation matrix, we will not further dive into High-level Full-stack Frameworks. With the experience of the developer of FestivalConnect and the research that has been done, it concludes that the .NET Core API is the most suitable choice. It scored great on each metric, making it a great fit to start developing for FestivalConnect.

What enterprise software platforms in regards to the front-end best fit this project?

Deciding the best front-end framework, there are several criteria points to consider in the final decision, when making an evaluation matrix. First, let's go over the frameworks and give them a brief introduction.

- **Blazor:** Modern front-end web framework that is based on HTML, CSS, and C# that helps you build web apps faster. With this framework, you can build web apps with components that can reused., that can be run from both the client and the server, so you can deliver a great web experience. (Build beautiful web apps with Blazor, n.d.)
- **React:** An open-source JavaScript library that is used for front-end development. It is component-based and lets you build high-quality user interfaces for web apps, Facebook has been developed with it. (React for Front-End Development – Features & Benefits [2024], 2024)
- **Angular:** Angular is an open-source software, which is a JavaScript framework written in TypeScript. Google supports the system, and it is a way to develop single-page applications. Also, it comes with providing a clear structure for developers to work with. (What is Angular?: Architecture, Features, and Advantages, 2023)

Can the frameworks handle the expected user load?

The frameworks need to handle at most for FestivalConnect 50 million concurrent users, this is also stated in (Jacobs, 2024, Software Requirements Specification). It needs to handle this without any performance downgrades.

Blazor

It uses WebAssembly and .NET integration, supporting the client and server-side development for high-performance applications. Server-side hosting provides hosting several performance benefits, especially for compute-intensive tasks. (Panchasara, 2023) Furthermore, it uses one-way-, event-, and two-way data binding, giving variety to choose the best option in certain areas (Blazor Data Binding : One-way, Two-way and Event Binding, 2022)

React

React has a really good performance due to the virtual DOM, making it efficient in UI updates since it minimizes manipulations on the actual DOM. Client-side rendering and a component-based architecture make sure that it limits the loading time and promotes scalability. The libraries and functions that are available in the React ecosystem, make sure that there are enough tools for optimizations. (Panchasara, 2023)

React uses One-Way Data Binding, which renders the modifications only after the mode state is updated. This gives more control to the developers over developing FestivalConnect. (Dhaduk, 2024)

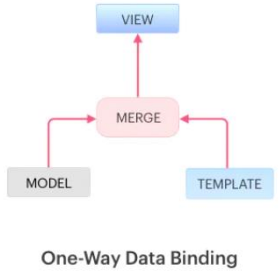

Figure 1: One-Way Data Binding

Angular
With Angular you can quickly load applications, it offers developers-friendly features. It is a component-based architecture. It uses a two-way data binding, keeping the model and view in sync all the time. (Dhaduk, 2024)

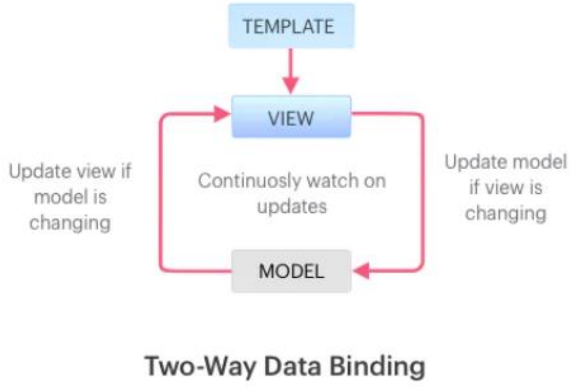
TEMPLATE
VIEW
Update view if model is changing
Continuosly watch on updates
Update model if view is changing
MODEL
Two-Way Data Binding

Figure 2: Two-Way Data Binding

Conclusion
The performance of each platform is in the state to handle the performance that FestivalConnect expects. Of course, keeping in mind fine optimizations for structure and using the best practices.

Do the frameworks have a cost of the product or any licensing?

The budget of this project is zero, with only the help of the licensing offered by Fontys.

Blazor
Free to use, has some libraries that may be paid for commercial use, such as DevExpress or Telerik.

React
Both React and JavaScript are free to use.

Angular
Free to use.

Conclusion
They all fit the requirements of FestivalConnect.

What is the experience of the development of FestivalConnect?

The developer has experience gained during the third semester at the University of Fontys. Also throughout my university journey until now I had a lot of experience in .NET, and during the internship, I gained a lot of experience using Blazor. Additionally, I have never worked with Angular. Therefore both React and Blazor are at minimal risk of wasting time when getting used to the system and setting things up.

Does the platform have an active community and support options to fall back on?

For each framework, it is important to have a strong community, that way it improves user experience and ongoing development. Also with a community, due to sharing content, you can easily find your answers, by looking at forums, tutorials, official documentation, and community pages.

Blazor

Blazor has a growing community that is relatively small in comparison to React and Angular. Nonetheless, it has a lot of supporting libraries, forums, code projects on GitHub, StackOverflow pages, and its large documentation to get started with Blazor. Also, it is supported by most browsers (Apple Safari, Google Chrome, Microsoft Edge, and Mozilla Firefox).

React

React has a community of millions of developers. There are a lot of questions found on Stackoverflow, Discussion Forums such as Reddit's React Community or Reactifflus online chat, and also their documentation with the latest news updates on X and the official blogs. (Where To Get Support, n.d.)

Angular

It is supported by all the popular browsers. It also has platforms such as Reddit and Stackoverflow to find the needed resources and official documentation. It is supported by Google, which maintains the framework.

Conclusion

All the frameworks are well-supported, and maintained by their creators. They are supported by the popular browsers. Even though Blazor has a smaller community, they are growing and rapidly creating a big community. Also, the big documentation of Microsoft on Blazor is a big plus to compensate for the potential size of the community.

Does the platform have enough security options to properly secure FestivalConnect?

The frameworks can offer various security options that can be beneficial for the website to be safe.

Blazor

Blazor Web Applications prioritize security with mechanisms such as authentication, input validation, session management, error handling, data protection, and denial of service prevention (DDoS). With built-in mechanisms, this thing will have a secure user interaction, and it is easy to use for developers. Also, Blazor implements limits and restrictions to prevent attacks and to get better overall security and trust in the system. (Nelson, 2023)

React

React has several security measures against common vulnerabilities like Cross-Site Scripting (XSS), Broken authentication, SQL injection, XEE, and other kinds. It has best practices using authorization that even makes it possible for multi-factor authentication, see the following picture.

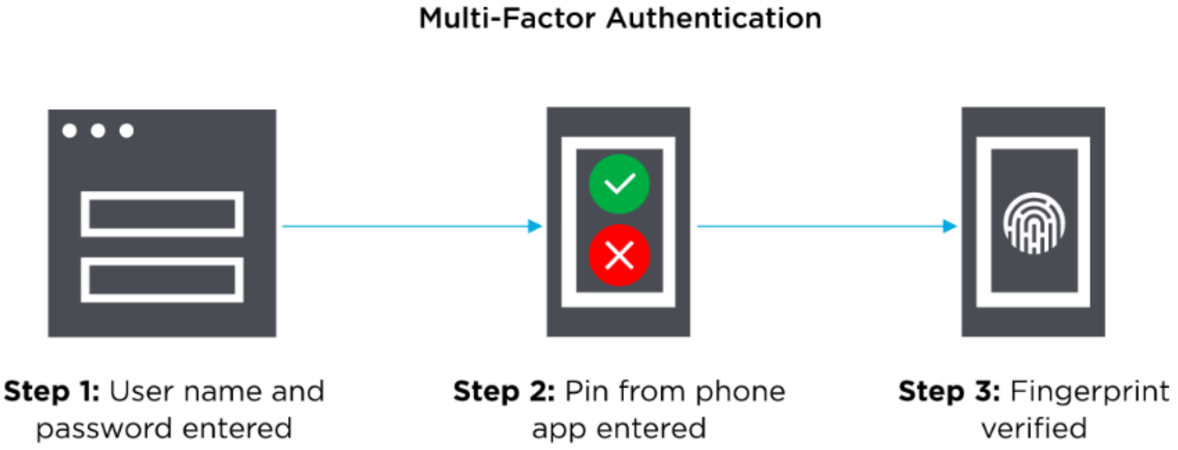
Multi-Factor Authentication
Step 1: User name and password entered
Step 2: Pin from phone app entered
Step 3: Fingerprint verified

Figure 3: Multi-factor authentication example

In React there are more secure coding practices and React's built-in features that developers can use. With the right implementation, developers can create a safe environment, preventing potential threats. (Swadia, 2021)

Angular

Angular also supports a lot of built-in features for security. It has features against common vulnerabilities like cross-site scripting (XSS). Angular is treating all values as untrusted by default, making it so that it will sanitize and escape potential harmful content. Also, it states that you keep current with the latest releases of Angular. Also don't make custom versions, since these tend to fall behind. (Security, n.d.)

Conclusion

All frameworks provide secure environments with built-in mechanisms to use, for an easy-to-protect to protect and secure development process.

How easy is the ease of use of the platforms?

Blazor

To begin with, Blazor is written with HTML, CSS, and C# meaning it is consistent with the languages on the front-end and back-end, making understanding code better. Also in the .Net environment, the front-end and back-end can both use a shared library of models, improving consistency throughout development. Moreover, it is a component-based architecture with promotes code reuse and maintenance. Also, it has great functionality for debugging with Visual Studio to better streamline the development workflows.

React

Easy to use and learn, brings concepts of HTML and JavaScript together with beneficial additions. It also has components that can be reusable. (Deshpande, 2023) Additionally, React has stable code, they are predictable in how it is designed, and it uses a technique that the data flow is always in a single direction, meaning parent to child, when a child changes the parent will keep the default properties. (Chernysh, 2023)

Angular

Also designed to be user-friendly, modular, and extensible. For separation of concern, it uses a Model-View-Controller architecture. Angular has a component-based use, that can be reused. (Onome, 2024)

Conclusion

All the frameworks are easy and friendly to use. The edge goes to Blazor since the consistency of the same language, with also having the option to share a library of exchange models.

SWOT-Analysis

To give a clear overview of what the strengths, weaknesses, opportunities, and threads are for these frameworks, I will show a SWOT analysis to get a final overview of choosing the framework that fits FestivalConnect.

Blazor

	Beneficial	**Harmful**
Internal	**Strengths** **1. Based on HTML, CSS, and C#, offering consistency with backend language.** **2. Component-based, promotes code reuse and maintenance.** **3. Good security features (built-in)** **4. Growing community** **5. A lot of official documentation.**	**Weaknesses** **1. smaller community compared to React and Angular** **2. Limited third-party libraries compared to other frameworks**
External	**Opportunities** **1. Rapid growing community and ecosystem.** **2. Integration with .NET environment for shared libraries.**	**Threads** **1. Potential challenges in finding specific third-party solutions or support.**

React

	Beneficial	Harmful
Internal	Strengths 1. Large community, millions of developers 2. Efficient DOM for improved performance 3. Storing security measures against common vulnerabilities. 4. Predictable and stable codebase. 5. Component-based architecture for code reusability	Weaknesses 1. Moderate learning curve for beginners 2. Limited support for TypeScript compared to Angular
External	Opportunities 1. A lot of third-party libraries and tools 2. Continuous updates and improvements from the community	Threads 1. Complexity in managing state and data flow in larger applications

Angular

	Beneficial	Harmful
Internal	Strengths 1. Supported by Google with a large community 2. Two-way data binding for efficient data sync. 3. Strong security with built-in mechanisms. 4. Component-based architecture for modular design.	Weaknesses 1. Harder to learn, compared to React. 2. Limited support for third-party libraries.
External	Opportunities 1. Integration with Google's ecosystem and tools. 2. Continuous updates.	Threads 1. Potential challenges in managing a larger application.

Evaluation Matrix

To conclude with the points that are discussed between the font-end frameworks, we will use an evaluation matrix to conclude.

Evaluation Matrix	Weight	Blazor Web Assembly	React	Angular
Scalability	0.20	++	++	++
Performance	0.20	+	+	+
Cost/License	0.20	+	+	+
Support & Community	0.20	+	++	++
Personal Skills	0.15	++	+	-
Security	0.25	++	++	++
Flexibility (ease of use)	0.20	++	++	+
Total:	N/A	2,20	2,25	1,90

- = Not preferred (0 * Weight score)
+ = General (1 * Weight score)
++ = Preference (2 * Weight score)

Conclusion

Even though React is not that known to the development of FestivalConnect, it does have great support and community and is easy to use, has good security practices, which will therefore be the choice of framework for the front end. Also from the conducted research, it barely won, meaning that all the frameworks are suitable for the FestivalConnect front-end choice, but due to the slight edge, it will get the preference.

References

.NET Core vs .NET Framework: How to Pick a .NET Runtime for an Application. (2024, 03 05). Retrieved from stackify: https://stackify.com/net-core-vs-net-framework/#:~:text=NET%20Core%20is%20used%20to,and%20F%23%20in%20both%20runtimes.

Adelli, A. (2023, 11 13). *Efficiently Scaling Django for Millions*. Retrieved from linkedin: https://www.linkedin.com/pulse/efficiently-scaling-django-millions-adarshreddy-adelli-mptoc/?trk=articles_directory

ASP.NET Core Blazor. (2024, 09 02). Retrieved from learn.microsoft: https://learn.microsoft.com/en-us/aspnet/core/blazor/?view=aspnetcore-8.0

baeldung. (2023, 07 25). *Jakarta EE 8 Security API*. Retrieved from baeldung: https://www.baeldung.com/java-ee-8-security

Binmile. (2021, 10 12). *How Django Apps Help Business Enterprises Simplify Processes and Increase Revenue*. Retrieved from linkedin: https://www.linkedin.com/pulse/how-django-apps-help-business-enterprises-simplify-/

Blazor Data Binding : One-way, Two-way and Event Binding. (2022, 08 21). Retrieved from dotnettricks: https://www.dotnettricks.com/learn/blazor/blazor-data-binding-one-way-two-way-and-event-binding

Build beautiful web apps with Blazor. (n.d.). Retrieved from dotnet.microsoft: https://dotnet.microsoft.com/en-us/apps/aspnet/web-apps/blazor#:~:text=Blazor%20is%20a%20modern%20front,can%20deliver%20great%20web%20experiences.

Buy Visual Studio. (n.d.). Retrieved from visualstudio.microsoft: https://visualstudio.microsoft.com/vs/pricing/?tab=enterprise

Celebrate 20 years of Tomorrowland in 2024 with the new theme 'LIFE'. (2023, 11 15). Retrieved from tomorrowland: https://www.tomorrowland.com/home/article/tomorrowland-belgium-2024-theme-and-dates

Chernysh, A. (2023, 12 14). *When And Why You Should Use React?* Retrieved from uptech: https://www.uptech.team/blog/why-use-react

Cuthbert, E. (2022, 05 25). *.Net Core Benefits: Why It is the most Desirable in 2022*. Retrieved from medium: https://medium.com/geekculture/net-core-benefits-why-it-is-the-most-desirable-in-2022-73a90d7ab2a6

Deshpande, C. (2023, 10 05). *The Best Guide to Know What Is React*. Retrieved from simplilearn: https://www.simplilearn.com/tutorials/reactjs-tutorial/what-is-reactjs#:~:text=Small%20learning%20curve%3A%20React%20is,proper%20understanding%20of%20React's%20library.

Dhaduk, H. (2024, 02 02). *Angular vs React: Which to Choose for Your Front End in 2024?* Retrieved from simform: https://www.simform.com/blog/angular-vs-react/#:~:text=In%20terms%20of%20performance%2C%20bundle,optimizations%20and%20Virtual%20DOM%20implementation.

Django introduction. (n.d.). Retrieved from developer.mozilla: https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Introduction

Godina, N. (2020, 07 21). *How Django can handle 100 millions of requests per day*. Retrieved from medium: https://medium.com/ebs-integrator/how-django-can-handle-100-millions-of-requests-per-day-c4cdbf48639e

Isah, J. (2023, 08 31). *Web Security in Django – How to Build a Secure Web Application*. Retrieved from freecodecamp: https://www.freecodecamp.org/news/how-to-build-a-secure-django-web-app/#:~:text=Django%20uses%20a%20robust%20and,hashing%20mechanism%20consists%20of%20PBKDF2.

Kamal, A. (2023, 03 06). *Reasons Behind The Popularity Of .NET Core*. Retrieved from linkedin: https://www.linkedin.com/pulse/reasons-behind-popularity-net-core-ahmed-kamal/

Learn Django. (n.d.). Retrieved from bootcamp.berkeley: https://bootcamp.berkeley.edu/resources/coding/learn-django/#:~:text=As%20with%20any%20skill%2C%20learning,as%20two%20to%20three%20weeks.

LLC, D. T. (2023, 08 05). *Comparing Java Frameworks: Spring vs. Java EE (Jakarta EE)*. Retrieved from linkedin: https://www.linkedin.com/pulse/comparing-java-frameworks-spring-vs-ee-jakarta/

Murimi, A. (2024, 02 06). *Why Django Is Your Best Bet for Web Development*. Retrieved from linkedin: https://www.linkedin.com/pulse/why-django-your-best-bet-web-development-alex-murimi-4qwdf/

Nelson, J. C. (2023, 10 23). *[Blazor] Docs on Blazor Web security #51577*. Retrieved from github: https://github.com/dotnet/aspnetcore/issues/51577

Onome, D. (2024, 01 01). *Angular vs. React: The Difference and Which Framework to Choose*. Retrieved from ninetailed: https://ninetailed.io/blog/angular-vs-react/

Panchal, K. (2022, 09 30). *Why ASP.NET Core is best for Enterprise Web Application Development?* Retrieved from ifourtechnolab: https://www.ifourtechnolab.com/blog/why-asp-net-core-is-best-for-enterprise-web-application-development#:~:text=ASP.NET%20Core%20is%20the%20best%20framework%20for%20enterprise%20web,Windows%2C%20macOS%2C%20or%20Linux.

Panchasara, M. (2023, 10 26). *Blazor vs React: Detailed Comparison*. Retrieved from radixweb: https://radixweb.com/blog/blazor-vs-react#What

Pricing for Django. (2023, 06 26). Retrieved from capterra: https://www.capterra.com/p/234557/Django/pricing/

React for Front-End Development – Features & Benefits [2024]. (2024, 01 11). Retrieved from brainhub: https://brainhub.eu/library/reasons-to-choose-react#:~:text=To%20give%20you%20a%20gentle,it%20works%20with%20Virtual%20DOM.

React vs Angular: What to Choose for Enterprise Application in 2024. (2024, 01 13). Retrieved from program-ace: https://program-ace.com/blog/react-vs-angular/#:~:text=However%2C%20both%20Angular%20and%20React,choice%20for%20this%20particular%20project.

Security. (n.d.). Retrieved from angular: https://angular.io/guide/security

Spring Security. (n.d.). Retrieved from docs.spring: https://docs.spring.io/spring-security/reference/index.html#:~:text=Spring%20Security%20is%20a%20framework,for%20securing%20Spring%2Dbased%20applications.

Stukalov, A. (2018, 07 2). *Classification of Development Frameworks for Enterprise Applications*. Retrieved from javacodegeeks: https://www.javacodegeeks.com/2018/06/classification-development-framework.html

Swadia, S. (2021, 10 28). *How to Secure Your React.js Application*. Retrieved from freecodecamp: https://www.freecodecamp.org/news/best-practices-for-security-of-your-react-js-application/

What is Angular?: Architecture, Features, and Advantages. (2023, 07 24). Retrieved from simplilearn: https://www.simplilearn.com/tutorials/angular-tutorial/what-is-angular#:~:text=Angular%20is%20an%20open%2Dsource,for%20developers%20to%20work%20with.

What is Jakarta EE? (n.d.). Retrieved from educative: https://educative.io/answers/what-is-jakarta-ee

what's the difference between Spring and Spring Boot in Java? (2024, 02 24). Retrieved from anywhere.epam: https://anywhere.epam.com/en/blog/spring-boot-vs-spring

what's the difference between Spring and Spring Boot in Java? (2024, 02 24). Retrieved from anywhere: https://anywhere.epam.com/en/blog/spring-boot-vs-spring

Where To Get Support. (n.d.). Retrieved from legacy.reactjs: https://legacy.reactjs.org/community/support.html

Wickramasinghe, S. (2021, 10 07). *The Spring Framework Beginner's Guide: Features, Architecture & Getting Started*. Retrieved from bmc: https://www.bmc.com/blogs/spring-framework/#:~:text=The%20Spring%20Framework%20is%20an,Aspect%2Doriented%20programming%20(AOP)

Jacobs, L. (2024). Software Requirements Specification (Unpublished manuscript), FontysICT.