6/14/2024

Research Report

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

Introduction 2
Problem 2
Research Questions 3
Research Methods 5
Estimation of Required Time 8
Description of the Process 9
Research Question and Main Purpose 9
DOT Framework Usage With Results 10
Sub-Question 1: What are the key features and functionalities that should be included in FestivalConnect to improve the festival experience for users? 10
Sub-Question 2: What kind of behaviors need to be taken into account, on what the users expect from the FestivalConnect application? 10
Sub-Question 3: Which technologies and architectural frameworks and methods are most suitable for developing FestivalConnect, while having an eye on scalability, security, and maintainability? 11
Sub-Question 4: How can user feedback and community engagement be implemented into FestivalConnect to better the sense of belonging to the festival and improve interaction between the festival-goers and organizers? 12
Sub-Question 5: What strategies and technologies can be implemented to make just that the law, regulations, and industry standards are guaranteed, keeping in mind data privacy and security? 13
Sub-Question 6: What are the most effective methods for testing and validating FestivalConnect to make sure that it will meet the specified quality standards and requirements? 13
Sub-Question 7: How can cloud-native principles and technologies be used to improve the scalability of FestivalConnect, with a particular look at handling peak festival periods that lead to increasing user demand? 14
Sub-Question 8: What measures should be taken into account to minimize the security risks and that FestivalConnect is designed with proper security, considering the trad-offs between performance? 14
Sub-Question 9: How can FestivalConnect be designed and implemented to support continuous software development and deployment? 15
Main Question Conclusion 15
References 16

Introduction

This document will cover how I have conducted all my research. This includes the answers to the research questions with the methods that have been used to do it and the results that are found from this research. The research is made in separate documents, so these documents will give the answers with references to the documents that have the research in them.

Problem

The main problem is that the festival industry doesn't have a digital platform that can effectively connect the festival goers and provide organizations with easy access to feedback and interaction with festival attendees. Therefore, there will be missed opportunities to make a great community and a limited reach to discover new events.

Research Questions

FestivalConnect aims to have a digital platform for festival-goers and organizers to connect and improve the festival experience. However, without proper consideration for maintainability while also keeping in mind the performance, scalability, security, and privacy. The platform can become difficult to update and maintain over time, hindering the ability of FestivalConnect to adapt to the evolving festival industry.

Main research question:

"How can FestivalConnect be developed and maintained to optimize the privacy, performance, security, and scalability throughout its lifecycle?"

Sub-questions:

1. What are the key features and functionalities that should be included in FestivalConnect to improve the festival experience for users?
2. What kind of behaviors need to be taken into account, regarding what the users expect from the FestivalConnect application?
3. Which technologies and architectural frameworks are most suitable for developing FestivalConnect, while having an eye on scalability, security, and maintainability?
    a. What front- and back-end technologies are best fit for handling FestivalConnect, to handle large volumes of user data and make sure of scalability?
    b. Can a certain architecture improve the scalability and maintainability of FestivalConnect?
    c. What database technologies can offer secure and efficient data handling for storing user information and event data?
    d. What is a good framework to make sure that internalization can be used and people from all over the world can use the system?
4. How can user feedback and community engagement be implemented into FestivalConnect to better the sense of belonging to the festival and improve interaction between the festival-goers and organizers?
    a. How can we properly implement the features in a user-friendly way?
5. What strategies and technologies can be implemented to make just that the law, regulations, and industry standards are guaranteed, keeping in mind data privacy and security?
    a. What are the needed strategies to make sure that it complies with data protection laws, such as GDPR?
    b. What legal regulations and industry standards are set for collecting, storing, and use of user data in the festival industry?
    c. How can we assess that these security standards are met and solve potential vulnerabilities or risks?
6. What are the most effective methods for testing and validating FestivalConnect to make sure that it will meet the specified quality standards and requirements?
    a. What is the most suitable testing to use, to verify the functionality and quality of FestivalConnect?
    b. Can automated testing tools and frameworks streamline the testing process and improve the consistent and reliable results?

    c. What metrics can be used to evaluate the effectiveness of testing with effect on the quality?

7. How can cloud-native principles and technologies be used to improve the scalability of FestivalConnect, with a particular look at handling peak festival periods that lead to increasing user demand?

    a. What cloud services can be used by FestivalConnect and offer the necessary infrastructure and tools for deploying and managing the environment?

    b. What are the best architectures to use inside cloud services to improve scalability and is cost-efficient?

    c. Can there be different strategies implemented to minimize latency for users accessing FestivalConnect from different geographic locations?

8. What measures should be taken into account to minimize the security risks and that FestivalConnect is designed with proper security, considering the trade-offs between performance?

    a. What security threats and vulnerabilities are most relevant to FestivalConnect, and how can we overcome those weaknesses?

    b. How can secure coding practices, such as input validation or error handling play a role in integrating secure code?

    c. What authentication and authorization technologies are needed to protect user accounts and protect them from sensitive data from unauthorized access?

9. How can FestivalConnect be designed and implemented to support continuous software development and deployment?

    a. What DevOps practices and tools can be used to support continuous integration and deployment for FestivalConnect?

    b. How can version control systems and branching strategies give an improved environment to develop in?

Research Methods

During this project, I will do research to get to know what the best technologies and practices are for FestivalConnect. The Development Oriented Triangulation Framework (DOT-Framework) will be used. This framework has several strategies such as library, lab, showroom, and workshop, see the following picture.

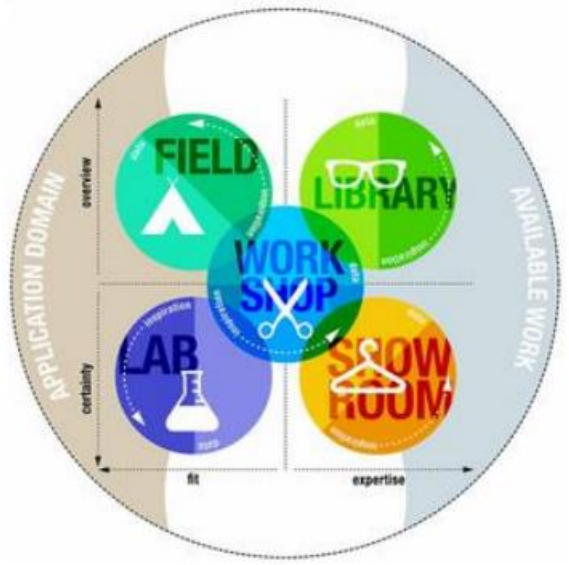


Figure 1: DOT-Framework

To answer the research questions, we need to have a mix of methods to answer these research questions.
See the following strategies:

- **Library**: Getting to know what is already been done, guiding to the best information to help me further design my FestivalConnect.
- **Field**: Understand the context. Getting to know the end user's needs, desires, and limitations.
- **Lab**: Test parts and concepts of the product. Test things to validate if it works as intended.
- **Showroom**: Test the ideas that you have for existing work. Testing your product to general guidelines or show prototypes to your stakeholders.
- **Workshop**: Use this strategy to explore new opportunities.

(The DOT Framework, n.d.)
Let's now list all the research questions with certain research methods to answer them. Note that this is just a guideline, and there can always be used another strategy.

1. What are the key features and functionalities that should be included in FestivalConnect to improve the festival experience for users?
    a. **Literature Study**: Getting to know who my audience is going to be. Based on this adjust my functionalities, keeping in mind and prioritizing what is most important to them.
    b. **Community research**: What are the existing communities like in the festival industry? Are they diverse? Do they have special needs?
    c. **Stakeholder analysis**: Look at what my stakeholders have to say during the talks with them. What are we discussing and prioritizing to have the most suitable functionalities for FestivalConnect?

2. What kind of behaviors need to be taken into account, on what the users expect from the FestivalConnect application?

    a. **Literature Study**: getting to know what the important behaviors for FestivalConnect application needs to be. Also looking at data to get to know what a performance measurements to aim for, what is the expected user base?

    b. **Community research**: Looking at what people have to say about the importance of how an application should behave. What is important for the user, in the context of festivals?

3. Which technologies and architectural frameworks and methods are most suitable for developing FestivalConnect, while having an eye on scalability, security, and maintainability?
    a. **Literature study**: Looking at what frameworks work best in the context of FestivalConnect.
    b. **Community research**: What do trustworthy communities have to say about certain frameworks, and does the information relate to my case?
    c. **SWOT analysis**: what are the strengths, weaknesses, opportunities, and threats when it comes to certain frameworks?
    d. **Design pattern research**: Can apply certain design patterns to the design of the database setup, architecture, etc.
    e. **Pitch**: Justifying to my stakeholders why I have chosen certain technologies.
    f. **Evaluation matrix**: Final conclusion matrix on how good the technology fit the application.

4. How can user feedback and community engagement be implemented into FestivalConnect to better the sense of belonging to the festival and improve interaction between the festival-goers and organizers?
    a. **Community research**: Look for what the special needs are for this sector.
    b. **Design pattern research**: are there needed patterns that can improve the quality and structure of the code when implementing these features?
    c. **Task analysis**: what is the precise flow of these functions, how do these work, and can this be explained with a certain flow diagram on both high-level as well as in-depth details?
    d. **Prototyping**: Giving a visual look at how the system can look and function.

5. What strategies and technologies can be implemented to make just that the law, regulations, and industry standards are guaranteed, keeping in mind data privacy and security?
    a. **Problem analysis**: what are the problems that occur that can lead to weak privacy and security? Getting to know what the real problem is, before implementing and regulating some rules into the application?
    b. **Literature study**: Are there industry standard regulations when it comes to collecting, storing, and use of user data in the festival industry? What can be the best practices and needed requirements to properly cherish the laws?
    c. **Available product analysis**: Look for best practices when it comes to handling data in a software enterprise application.
    d. **Best good and bad practices**: what should be done and don’t to have secure data handling.

6. What are the most effective methods for testing and validating FestivalConnect to make sure that it will meet the specified quality standards and requirements?
    a. **Literature study**: Look for certain patterns to properly test your application.
    b. **Best good and bad practices**: when it comes to testing, what should we do and not?
    c. **System test**: Make test scenarios to properly test the user stories. Make a separate test plan for this.
    d. **Unit test**: test the business logic and if this logic is correct.
    e. **Component test**: testing the flow of the application.
    f. **Benchmark testing**: Testing the system benchmarks and if it complies with the agreed metrics.

7. How can cloud-native principles and technologies be used to improve the scalability of FestivalConnect, with a particular look at handling peak festival periods that lead to increasing user demand?
    a. **Literature study**: Researching what the best practices are for the cloud. How to increase user traffic?
    b. **Available product analysis**: what is currently already available on the market, that can be used and is proven to be efficient and good.
    c. **Problem analysis**: Does the improved scalability of FestivalConnect only depend on cloud-native technologies or how can we make sure that the solution that we want to provide is helping?

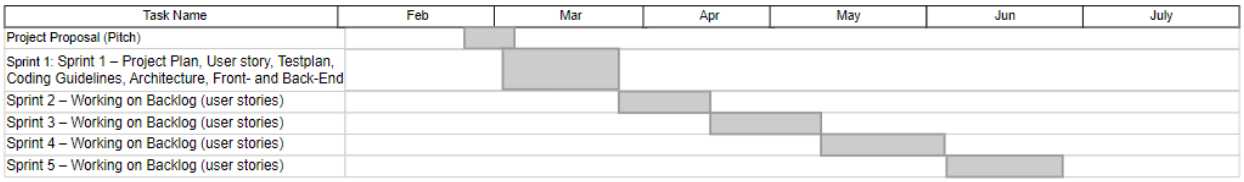
8. What measures should be taken into account to minimize the security risks and that FestivalConnect is designed with proper security, considering the trad-offs between performance?
    a. **Code review**: Regularly check the code of the design security to check for faults.
    b. **Unit test**: is the created code secure and is it logically correct?
    c. **Literature study**: Looking at the most common vulnerabilities of a website. What can be possible outlooks for hackers to hack the system? The needed security for the product.
    d. **Available product analysis**: what are the best practices on the market to make a secure website?

9. How can FestivalConnect be designed and implemented to support continuous software development and deployment?
    a. **Literature study:** Looking at the best software to support CI/CD.
    b. **Available product analysis:** Getting to know what already exists on the market to ensure CI/CD.
    c. **Pitch**: Showing to the stakeholders why FestivalConnect is designed to have CI/CD.
    d. **Community research**: More knowledge of people there opinions if they recommend certain designs and software.

Estimation of Required Time

Phasing	Effort	Start	Ready
1 Pre-project phase – Work on the project proposal, initial ideas	2 weeks	19-02-2024	03-03-2024
2 Sprint 1 – Project Plan, User story, Testplan, Coding Guidelines, Architecture, Front- and Back-End Choice	3 weeks	04-03-2024	24-03-2024
3 Sprint 2 – Working on Backlog (user stories)	3 weeks	25-03-2024	14-04-2024
4 Sprint 3 – Working on Backlog (user stories)	3 weeks	15-04-2024	12-05-2024
5 Sprint 4 – Working on Backlog (user stories)	3 weeks	13-05-2024	02-06-2024
6 Sprint 5 – Working on Backlog (user stories)	3 weeks	03-06-2024	23-06-2024

To visualize it more, here is a Gantt chart of the documents that I want to cover, over the time of this project duration. Note, that this chart may not cover all the documents, with also having that some documents need initial research at the beginning and an update throughout the sprints (such as user stories document).



To further divide the research question on when these are expected to be finished the following division.

- Sprint 1-2: Sub questions 1, 2, 3, and 4.
- Sprint 2-3: Sub questions: 6, 8, and 9.
- Sprint 3-5: Sub questions: 5 and 7.

Description of the Process

Throughout this semester, I used several research methods and tools to guide me on the right track. All these research methods were applied to different research questions that ultimately led to the answer to the main question. The details of the research and how I worked throughout this semester more descriptively, (Jacobs, 2024, Project Plan) explain the problem and way of working clearly.

Research Question and Main Purpose

The main question of this project was:

"How can FestivalConnect be developed and maintained to optimize the privacy, performance, security, and scalability throughout its lifecycle?"

This semester was all around making an enterprise software application, with this question the application could be correctly implemented by doing multiple research to cover the non-functional requirements so that the application will behave as supposed. The non-functional requirements in the main question are important throughout the whole development process and during this research, a constant consideration between these and other non-functionals had to be taken.

DOT Framework Usage With Results

To answer the main research question, several sub-questions need to be answered throughout this semester, making use of DOT-Framework. We will go over each sub-question what methods are used and where I applied them in my application.

Sub-Question 1: What are the key features and functionalities that should be included in FestivalConnect to improve the festival experience for users?

- **Literature Study**: By looking into several festival branches and different applications that have the same kind of style I want to implement, I came to various conclusions on what needs to be implemented and how this can be handled visually. The design can be found back in (Jacobs, 2024, Design Document), and the functionalities are listed in (Jacobs, 2024, Software Requirements Specification, Functional Requirements).
- **Community research**: This one I looked specifically at what can be important in regards to the nonfunctional requirements in regards to privacy, like for example functionalities that can hide your preferences in order to not see that you are interested in festivals that are for example related to LGBTQ+, which can indicate from other people what sexual identification you have. Furthermore, what is also important is what people will expect from a community-based application.
- **Stakeholder analysis**: This semester I had to show several learning outcomes, with the talks with both my semester coach and technical teachers we prioritized functionalities by looking at how to go further in achieving the learning outcomes.

The conclusion on this question was that by looking into what people expect and how applications are built within the same range of functionalities, this listed down functional requirements which are stated inside of the (Jacobs, 2024, Software Requirements Specification, Functional Requirements).

Sub-Question 2: What kind of behaviors need to be taken into account, on what the users expect from the FestivalConnect application?

- **Literature Study**: By studying the importance of non-functional requirements that are expected from the user, I did several kinds of research on what users can expect. This is broadly written down in the in (Jacobs, 2024, Software Requirements Specification, Non-Functional Requirements). This lists down how FestivalConnect is taking into account performance, scalability, security, and privacy by looking at what users expect from an enterprise software application. For example, an important part is taking a look when the festival season starts because this means that a load of applications will increase which also means that the application will decrease in performance, therefore these performances can and need to be taken into account to also have FesivalConnect be able to scale up when needed.
- **Community research**: I have looked into scenarios of what users who visit festivals want from such an application. This includes internalization but also having my application secure and my data private.

This question was answered by taking into account several measurements and perspectives to make the right decisions for both what the client wants and at the same time what this means for the application. This all is concluded into several non-functional requirements that are listed in (Jacobs, 2024, Software Requirements Specification, Non-Functional Requirements FestivalConnect).

Sub-Question 3: Which technologies and architectural frameworks and methods are most suitable for developing FestivalConnect, while having an eye on scalability, security, and maintainability?

- **Literature study**: For the front- and back end but also for architecture, I have looked at several technologies that could fit FestivalConnect. Researching on various topics which technology will stand out for this application. This can be found in (2024, Jacobs, Software Platforms Research) and (2024, Jacobs, Architecture Research).
- **Community research**: Comparing the systems and technologies to each other, to look at what are the benefits and what can be difficult with this technology. This was to get to know how people will think about several technologies and if the communities are big enough to start using them.
- **SWOT analysis**: I have compared the front- and backend technologies to each other for each part and showed the strengths, weaknesses, opportunities, and threats to then have a quick overview of what can be a good solution for FesivalConnect.
- **Design pattern research**: Looking into the application setup of the chosen architecture and how it can benefit the application in terms of the nonfunctional requirements. The results with the explanations can be found in (Jacobs, 2024, Technical Design).
- **Pitch**: I justified to the teachers why I have chosen technologies and certain design patterns by telling how they relate to the non-functionals.
- Evaluation matrix: The conclusion in a concrete overview, which gives a quick overview of how I decided upon the technologies.

For the front end, I had the following conclusion.
Even though React is not that well-known for 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. To fully dive into how I came to this conclusion, see (2024, Jacobs, Software Platforms Research, What enterprise software platforms in regards to the front-end best fit this project?)
To continue for the back end.
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. To see further information (2024, Jacobs, Software Platforms Research, What enterprise software platform in regards to the back-end best fits this project?)
Furthermore the architecture.
For FestivalConnect, Microservices architecture is the one that stands out as the best choice due to the benefits of agility, scalability, and fault isolation. While other architectural styles like N-Tier, Web-Queue-Worker, Event-Driven, and Service-Oriented architectures offer certain advantages, they come with trade-offs.

- N-Tier offers scalability but lacks agility and fault isolation while also not being able to deploy small updates separately from microservices.
- Web-Que-Worker and Event-Driven architectures have great asynchronous tasks and real-time processing but may not offer the flexibility and independence of microservices.
- Service Oriented Architecture (SOA) also provides reusable services but lacks the agility and granularity of microservices.
- Serverless offers cost-efficiency and scalability but risks too much on vendor lock-in and has testing complexity.

Therefore, while each option has great advantages, microservices have a great balance for FestivalConnect's requirements and growth future. To read more (2024, Jacobs, Architecture Research).

This concludes the research while keeping in mind the non-functional requirements, this architecture is further applied (Jacobs, 2024, Technical design).

Sub-Question 4: How can user feedback and community engagement be implemented into FestivalConnect to better the sense of belonging to the festival and improve interaction between the festival-goers and organizers?

- **Community Research**: I have looked and researched if there are needed festival goers that are not heard of or if there is a certain lack of communication level between festival goers and festival organizers for several festivals.
- **Literature Study**: To build a user-friendly application that supports these functions I looked into principles to keep in mind that will ensure the ease of use and the nonfunctional requirements, found in (Jacobs, 2024, Design document).
- **Design pattern research**: I have looked into several patterns for the architecture of the application to improve the quality and structure but also the scalability of the application, more on this is found in (Jacobs, 2024, Technical Design).
- **Prototyping**: I made several wireframes and a page overview diagram for the application (Jacobs, 2024, Design document).

It is important to look into how the actual application will look. Applying these methods, found that there are a lot of sources and ways to give the user a pleasant experience for both visual as well as the expected behaviors from them such as fast responses, and proper error handling.

Sub-Question 5: What strategies and technologies can be implemented to make just that the law, regulations, and industry standards are guaranteed, keeping in mind data privacy and security?

- **Problem analysis**: By looking at what the requirements are regarding FestivalConnect protecting privacy and keeping in line with the law, this includes all the actions and nonfunctional that are involved in FestivalConnect, this can be found in (Jacobs, 2024, Data Storage and GDPR Compliance Strategy: Practical Guide, Understanding Data Requirements)
- **Literature Study**: studying how GDPR impacts my application and what needs to be done to avoid fines. This can also be found in (Jacobs, 2024, Data Storage and GDPR Compliance Strategy: Practical Guide, Privacy-Sensitive Data and GDPR Compliance)
- **Available product analysis**: By evaluating data storage using the CAP theorem I came to various conclusions as to what software to use to have consistency, availability, and partition tolerance for the application data storage. This can be found in (Jacobs, 2024, Data Storage and GDPR Compliance Strategy: Practical Guide, Evaluating Candidate Data Stores Using CAP Theorem).
- **Best good and bad practices:** Researching what is best for the application, I made several requirements and implementations that are made in FestivalConnect, listed in (Jacobs, 2024, Data Storage and GDPR Compliance Strategy: Practical Guide, Designing Architecture with Distributed Data Stores).

To conclude, FestivalConnect analyzed data storage and GDPR compliance for the application and the outcomes will successfully be incorporated, by implementing several data storages and other approaches to manage the persistent data. The architectural choices with the specific data requirements that FestivalConnect has set, ensure scalability, performance, and data integrity and it is compliant with GDPR. Furthermore, with storages of distributed data using MySQL, Redis, and RabbitMQ, FestivalConnect covers the key aspects of CAP theorem and with the architectural decisions of multitier and event-driven, will ensure scalability and fault tolerance. This design will FestivalConnect have an application that has data management and privacy protection that aligns with expected industry standards. For more in-depth information, see (Jacobs, 2024, Data Storage and GDPR Compliance Strategy: Practical Guide).

Sub-Question 6: What are the most effective methods for testing and validating FestivalConnect to make sure that it will meet the specified quality standards and requirements?

- **Literature Study:** research conducted on what test methods are important to ensure the availability and how to make sure the application is working as intended.
- **Best good and bad practices:** Looked into how FestivalConnect should approach the testing strategy.
- **System test:** These show scenarios based on the user stories that are made from the functional requirements, which are found in (Jacobs, 2024, Acceptance Testplan) where the user stories are located (Jacobs, 2024, User Stories).
- **Unit Test:** Testing single units of code that have business logic or particular workflows to see if they work as intended.
- **Component:** tested the flow of the application.
- **Benchmark testing:** Tested the deployed application on performance and if complies with the metrics.
- **E2E:** Testing the flow of the application as a user.

Testing is an important part of ensuring the quality and validating the usage of the application. It is essential that the tests are of good quality and that this can be integrated into an automated process in

order to constantly check on whether the quality of the code is meeting the expectations. Further information about testing can be found in (Jacobs, 2024, Acceptance Testplan).

Sub-Question 7: How can cloud-native principles and technologies be used to improve the scalability of FestivalConnect, with a particular look at handling peak festival periods that lead to increasing user demand?

- **Literature study**: Research has been done on what services can benefit FestivalConnect in regard to the non-functional requirements, see (Jacobs, 2024, Cloud Analysis).
- **Available product analysis**: Looked into several options that could be useful for FestivalConnect, to then choose what benefits the most, see (Jacobs, 2024, Cloud Analysis, Relevant Cloud Services).
- **Problem analysis**: Looked into what vulnerabilities FestivalConnect comes across such as storing securely the passwords, or single points of failure.

Using a cloud provider is beneficial to host several services to let FestivalConnect benefit from several things such as pay-as-you-go, resource usage and even hosted services available all over the world and much more. With the cloud providing Kubernetes services, which will provide a service to host my containerized application, it will let FestivalConnect scale up and down depending on the user demand.

Sub-Question 8: What measures should be taken into account to minimize the security risks and that FestivalConnect is designed with proper security, considering the trad-offs between performance?

- **Code review**: Have checks if the implemented security measurements are secure. This also included the dependency checks and whether FestivalConnec is using trustworthy third-party applications or services.
- **Unit tests**: The logic that is created for the security has been checked for correctness.
- **Literature study**: Research what the most common vulnerabilities are in web applications and how FestivalConnect can prevent this. This also included requirements that have been made to have a more secure application, which can be found in (Jacobs, 2024, Security Design)
- **Task analysis**: I analyzed the structure of the application and did boundary crossing analysis to find vulnerabilities in the application, see (Jacobs, 2024, Security Design, Analysis).
- **Available product analysis:** Researched the things that are best to use to secure the application which is stated in (Jacobs, 2024, Security Design, Requirements).

The importance of having a secure application is critical to prevent fines and even more hardening punishments. This question can be answered by the way I described it in the requirements inside (Jacobs, 2024, Security Design). This also for example includes automated security checks in the SDLC while also having the applied security implementations while keeping in mind the performance.

Sub-Question 9: How can FestivalConnect be designed and implemented to support continuous software development and deployment?

- **Literature Study**: Researched what DevOps and DevSecOps can contribute to the continuous software development and deployment in FestivalConnect, see (Jacobs, 2024, DevOps Report).
- **Available product analysis**: I looked into what stages and products are available to use to analyze and deploy my application to make sure the CI/CD. The results are in see (Jacobs, 2024, DevOps Report).
- **Pitch**: showed the teachers what application I used and validated what beneficial value it has.
- **Community research**: Gained knowledge of what people recommend for continuous development and deployment.

By applying various research methods to find the best software to apply to my FestivalConnect to ensure the quality of my application, it came down to having a structured way of working with a fully automated pipeline that has the stages: build, test, analyze, and deploy. With the importance of DevOps and how FestivalConnect integrates this, which can be seen in (Jacobs, 2024, DevOps Report), FestivalConnect will ensure that it will be designed and implemented that support a CI/CD environment.

Main Question Conclusion

To optimize and keep the privacy, performance, security, and scalability for FestivalConnect, we had to look at multiple aspects to approach this question.

Privacy is ensured with the implementation and appliance to GDPR practices, encryption for data storage of sensitive data, and user-controlled data preferences, gained from methods such as literature studies and community research, documented mainly in (Jacobs, 2024, Data Storage and GDPR Compliance Strategy: Practical Guide).

Performance is addressed by doing benchmark testing, continuous monitoring, and having a scalable architecture. The cloud-native services like Kubernetes services, will allow scaling during peak festival periods (Jacobs, 2024, Cloud Analysis).

Security measures include regular code reviews, automated testing and dependencies checks, and the security best practices for the SDLC. Furthermore, also looks into common vulnerabilities and the requirements that are set that are implemented and need to be checked in during the SDLC. This is all mainly in (Jacobs, 2024, Security Design) described.

Continuous improvement is ensured by looking into the community for feedback, to make sure that the application meets the user's needs. Moreover, the maintenance of the application will be done by prototyping and testing.

All and all developing and maintaining FestivalConnect to optimize the privacy, performance, security, and scalability will require a lot of facts and needs a great approach to integrate important topics such as GDPR compliance, scalable architecture, security measurements, and continous feedback and improvement. With the research I have done to find the best practices and technologies, FestivalConnec can provide a reliable, secure, and efficient platform for festival-goers and organizers, which will make sure the user will have an enjoyable experience user experience. Nevertheless, it is a continuous process of keeping up the standard of these non-functional requirements, and expanding your knowledge is essential in this constantly evolving industry because things you do not know are things the application can not handle.

References

The DOT Framework. (n.d.). Retrieved from ictresearchmethods: https://ictresearchmethods.nl/dot-framework/

Jacobs, L. (2024). Architecture Research (Unpublished manuscript), FontysICT.
Jacobs, L. (2024). Cloud Analysis (Unpublished manuscript), FontysICT.
Jacobs, L. (2024). Data Storage and GDPR Compliance Strategy: Practical Guide (Unpublished manuscript), FontysICT.
Jacobs, L. (2024). Design Document (Unpublished manuscript), FontysICT.
Jacobs, L. (2024). DevOps Report (Unpublished manuscript), FontysICT.
Jacobs, L. (2024). Project Plan (Unpublished manuscript), FontysICT.
Jacobs, L. (2024). Security Design (Unpublished manuscript), FontysICT.
Jacobs, L. (2024). Software Requirements Specification (Unpublished manuscript), FontysICT.
Jacobs, L. (2024). Software Platforms Research (Unpublished manuscript), FontysICT.
Jacobs, L. (2024). Technical Design (Unpublished manuscript), FontysICT.
Jacobs, L. (2024). User Stories (Unpublished manuscript), FontysICT.