Reno Muijsenberg

r.muijsenberg@student.fontys.nl

Abstract

S3 software portfolio

Festival planner



Contents

[Leeruitkomsten uitgelegd 2](#_Toc116292452)

[1. Web application 2](#_Toc116292453)

[2. Software quality 2](#_Toc116292454)

[3. Agile method 2](#_Toc116292455)

[4. CI/CD 2](#_Toc116292456)

[5. Cultural differences and ethics 2](#_Toc116292457)

[6. Requirements and design 2](#_Toc116292458)

[7. Business processes 3](#_Toc116292459)

[8. Professional 3](#_Toc116292460)

[Aantonen leerdoelen 4](#_Toc116292461)

[1. 4](#_Toc116292462)

[2. 5](#_Toc116292463)

[3. 6](#_Toc116292464)

[4. 6](#_Toc116292465)

[5. 6](#_Toc116292466)

[6. 7](#_Toc116292467)

[7. 8](#_Toc116292468)

[8. 9](#_Toc116292469)

# Leeruitkomsten uitgelegd

## 1. Web application

**User friendly:** You apply best practices when creating user interfaces and basic user experience testing and development techniques.

**Full stack:** You design and build a full stack application using a commonly accepted front end JavaScript framework and back-end application implementing relevant communication protocols, persistence of data by usage of ORM and addressing asynchronous communication issues.

## 2. Software quality

**Tooling and methodology:** Carry out, monitor, and report on unit integration, regression, and system tests, with attention for security and performance aspects, as well as applying static code analysis and code reviews.

## 3. Agile method

**Choose:** You are aware of the most popular agile methods and their underlying agile principles. Your choice of a method is motivated and based on well-defined selection criteria and context analyses.

## 4. CI/CD

**Design and implement:** You design a release process and implement a continuous integration and deployment solution (using e.g., Gitlab CI and Docker).

## 5. Cultural differences and ethics

**Recognize:** Recognition is based on theoretically substantiated awareness of cultural differences and ethical aspects in software engineering.

**Consider:** Adapt your communication, working, and behaviour styles to reflect project stakeholders from different cultures.

## 6. Requirements and design

**Multiple types of test techniques:** You apply user acceptance testing and stakeholder feedback to validate the quality of the requirements. You evaluate the quality of the design (e.g., by testing or prototyping) considering the formulated quality properties like security and performance.

## 7. Business processes

**Simple:** Involving stakeholders, predominantly sequential processes with one or two alternative paths.

**Related:** Business processes during which the software that you are developing will be used (business processes that the software must support by fully or partially automating them).

## 8. Professional

**Professional manner:**You develop software as a team effort according to a prescribed software methodology and following team agreements. You can track your work progress and communicate your progress with the team.

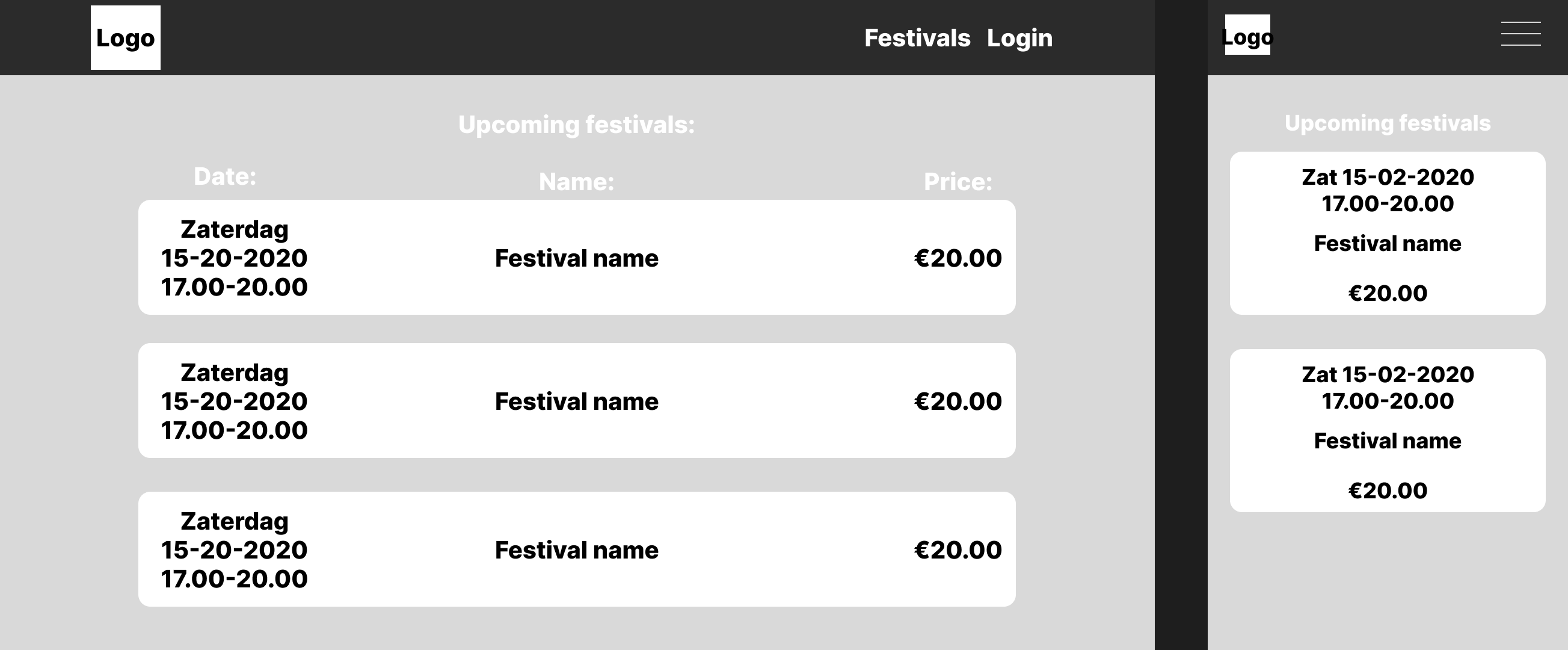
You actively ask and apply feedback from stakeholders and advise them on the most optimal technical and design (architectural) solutions.  
You choose and substantiate solutions for a given problem.

# Aantonen leerdoelen

## [1.](#_1._Web_application)

Ik heb gekozen om het front-end framework Angular te gebruiken ([uitgewerkte front-end keuze](https://github.com/RenoMuijsenberg/S3-Festival-Planner-Angular/wiki/Front-end-choice#front-end-language-choice)) en als backend heb ik ervoor gekozen om ASP.net te gaan gebruiken.

Na het kiezen van mijn programeer talen ben ik begonnen met het designen van de pagina die ik voor de eerste userstory nodig heb. Ik heb een snel efficiënt overzicht gemaakt van de festivals waardoor de gebruiker makkelijk kan zien wat de opkomende festivals zijn.



Wireframe front page

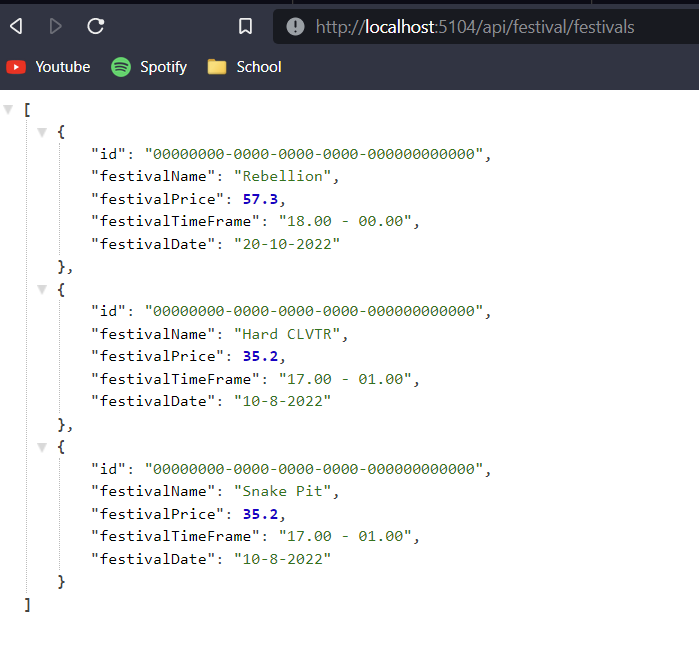
Vervolgens ben ik begonnen met het opzetten van het front-end project en ben ik een [video tutorial](https://www.youtube.com/watch?v=k5E2AVpwsko&t=3408s) gaan kijken over hoe Angular werkt. En heb met de kennis van deze tutorial de pagina van de wireframe nagemaakt in HTML, CSS (Screenshot bevat hardcoded data).

Graphical user interface, website

Description automatically generated

Front page in HTML, CSS

Nadat ik de hoofdpagina had gemaakt ben ik begonnen met het opzetten van de festival service. Ik ben hier mee begonnen zodat ik dynamische content op de hoofdpagina kan tonen. Ik he been project gestart in ASP.net en de endpoint gemaakt voor het ophalen van alle festivals door middle van ORM, ik heb dit voor nu met hardcoded data in de backend gedaan zodat ik kon testen of ik het kon ophalen via een HTTP request. Toen ik zag dat de endpoint werkte heb ik een database opgezet en deze geconnect met de app zodat de data ook echt uit de database komt.



API-data

## [2.](#_2._Software_quality)

Ik heb om ervoor te zorgen dat er zo weinig mogelijk bugs in zitten een static code analysis toegevoegd in mijn CI/CD hiervoor heb ik SonarCloud gebruikt.

A picture containing graphical user interface

Description automatically generated

SonarCloud scanner

## [3.](#_3._Agile_method)

## Scrum

Afbeelding met tekst, buiten, schermafbeelding

Automatisch gegenereerde beschrijvingVoor de proftaak werken wij de scrum methode. We werken aan ons project in sprints van 3 weken. Voor de user story’s gebruiken wij een Github bord dat we hebben ingedeeld met de categorieën: “User Story’s”, “Backlog”, “To Do This Sprint”, “In Progress,” “In Review” en “Done”

User story's met story points

### Sprint planning

Elke sprint beginnen we met een meeting met de product owner en gaan we samen kijken welke user story’s komende sprint gaan worden opgepakt en welke de hoogste prioriteit hebben. Dit wordt in overleg gedaan zodat we kunnen afstemmen met de product owner wat haalbaar is voor komende sprint. Dit besluit wordt genomen gebaseerd op het aantal story points wat wij sprints eerder behaald hebben. Deze story points zijn relatieve inschattingen van hoe lastig dat wij denken dat een user story gaat worden. Deze punten geven wij aan de user story’s voordat we in gesprek gaan met de product owner.

### Daily stand-up / stand-down

Elke dag dat we gaan werken aan dit project starten we met een stand-up. In deze stand-up gaan we samen kijken wat iedereen deze dag gaat doen. We pakken ons bord erbij en verdelen de issues over de mensen die aanwezig zijn. Dit gesprek wordt geleid door de scrum master. De scrum master in onze groep verwisselt elke sprint zodat iedereen een keer aan de beurt is geweest. De scrum master let dan ook op of het doel wat we hebben vastgesteld te halen is en of er nog extra werk moet gedaan worden. Op het einde van de dag wordt er rond gevraagd en op het github bord gekeken wat iedereen af heeft en waarom iemand iets niet af heeft terwijl hij/zij dit wel als taak had voor deze dag. Als blijkt dat de taak te moeilijk was voor de persoon dan spreken we af om de volgende dag erna te kijken met iemand die hier iets meer verstand van heeft.

### Sprint review

In de sprint review gaan we opleveren en presenteren wat we afgelopen sprint gehaald hebben. We doen dit aan de hand van een presentatie en een kleine demo van onze app. In de presentatie leggen we uit wat onze doelen waren deze sprint, welke van deze doelen we wel of niet gehaald hebben en dan ook waarom we deze niet gehaald hebben. Ook leggen we bepaalde keuzes uit die we gemaakt hebben en onderbouwen we waarom we voor dit hebben gekozen. De product owner geeft ons dan feedback/ complimenten over wat we afgelopen sprint gedaan hebben en ze stellen ons nog eventuele vragen als iets niet duidelijk was.

### Sprint retrospective

Aan het einden van onze sprint review dan houden we met ons groepje een retrospective, in deze retrospective bedenken we eerst allemaal individueel minimaal drie dingen die goed gingen deze sprint en drie dingen die minder goed gingen deze sprint. Vervolgens vullen we deze punten in op Github bord genaamd “Retrospective sprint (1, 2 etc.)”. De scrum master leest deze punten op en vraagt aan de persoon of hij/zij hier toelichting bij kan geven. De persoon legt dit dan uit en dan bedenken we met z’n allen wat we hier komende sprint aan kunnen veranderen of het juist zo laten. Dit wordt dan ook op het Github bord ingevuld onder het kopje “Action points”.

## [4.](#_4._CI/CD)

Voor de CI/CD ben ik begonnen met het kijken van een [tutorial](https://www.youtube.com/watch?v=R5ppadIsGbA). Ik heb hier met mee gevolgd en had uiteindelijk een CI die het project deed builden en testen. Aangezien deze tests nu nog leeg zijn liet hij dit slagen. Nadat ik dit werkend heb ben ik begonnen met het toevoegen van SonarCloud (een static code analyse tool).

Graphical user interface

Description automatically generated

CI/CD (GitHub actions)

## [5.](#_5._Cultural_differences)

## [6.](#_6._Requirements_and)

Ik heb een aantal mensen die geïnteresseerd waren in mijn product een foto gestuurd van hoe de homepagina eruitziet en hun hebben hier hun mening over gegeven, op deze manier pas ik de ‘user acceptance test’ toe in mijn product. Ik heb Thom en Kay gevraagd voor hun mening.

A screenshot of a video game

Description automatically generated

Reactie Thom

Graphical user interface, text, application, chat or text message

Description automatically generated

Reactie Kay

## [7.](#_7._Business_processes)

Ik heb een start gemaakt aan het maken van het C4 model. Ik heb als voorbereiding een [video](https://www.youtube.com/watch?v=x2-rSnhpw0g) gekeken over een conferentie waar dit uitgelegd werd. En vervolgens heb ik C1, C2 en C3 gemaakt.

Diagram

Description automatically generated

C4 model V1

In de eerste sprint had ik feedback gevraagd op mijn C4 model. Tijdens de feedback kwam ik erachter dat ik C2 en C3 door elkaar had gehaald dus heb ik van deze twee modellen er een gemaakt. Verder heb ik wat kleurtjes gegeven aan de componenten zodat dit nog duidelijker is.

Diagram

Description automatically generated

C4 model V2

Nadat ik ben begonnen met het maken van een Python web scrapper was er ook een van mijn service niet meer nodig. Eerst wou ik het zo maken dat de data door een admin gebruiker moest worden ingevuld waardoor ik dus een account service nodig had omdat de admin moest kunnen inloggen. Nu ik de scrapper gebruik is deze service dus niet nodi gen heb ik hierop het C4 model aangepast. Verder heb ik nog in plaats van eigen gemaakt vormen de standaard gebruikte C4 model vormen gebruikt.

Diagram

Description automatically generated

C4 model V3

## [8.](#_8._Professional)

Wanneer ik vragen heb communiceer ik op professionele wijze met mijn klasgenoten.