Dokumentation

von Koch Liam Gideon

Inhalt

[Anforderungen 3](#_Toc159573571)

[1) Informieren 3](#_Toc159573572)

[Planung 3](#_Toc159573573)

[Entscheidung 5](#_Toc159573574)

[Realisieren 5](#_Toc159573575)

[Testen 5](#_Toc159573576)

[Testfälle 5](#_Toc159573577)

[Testbericht 8](#_Toc159573578)

# Anforderungen

### 1) Informieren

Ich habe mich für den Zeitrechner entschieden.

Gruppe: Liam Koch

|  |  |  |  |
| --- | --- | --- | --- |
| Anforderungsnumer | Verbindlichkeit | Typ | Beschreibung |
| 1 | muss | funktional | Man kann das Frontend starten und eine Zeit übergeben. |
| 2 | muss | funktional | Das Backend wird aufrufen und die eingegebene Zeit wird übergeben. |
| 3 | muss | funktionale | Das Backen nimmt die übergebene Zeit an. |
| 4 | muss | funktionale | Das Backend holt sich die aktuelle Zeit und speichert die aktuelle Zeit. |
| 5 | muss | funktional | Das Backend rechnet die Zeit zwischen den beiden Zeiten aus. |
| 6 | muss | funktional | Das Backend schickt die Zeit an das Frontend. |
| 7 | muss | funktional | Das Frontend zeigt die Zeit in der Konsole an. |
| 8 | kann | funktional | Das Backend ist über Docker ansprechbar. |

d) Legen Sie fest, wo und wie Sie in der Gruppe Arbeitsergebnisse speichern und austauschen möchten.

Gitup Respository

# Planung

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Anforderung/Aufgabe | Nr. Arbeitspaket | Frist | Zuständigkeit | Beschreibung | Geplante Zeit |
| 1 | 1.A | 19.1.24 | Liam | Man erstellt ein Programm das nach einer Zeit fragt und die speichert. | 60 |
| 2 | 2.A | 19.1.24 | Liam | Man erstellt ein Backend, dass durch das Programm aufgerufen werden kann und eine Zeit übergeben kann. | 120 |
| 3 | 3.A, 3.B, 3.C, 3.D | 19.1.24 | Liam | Die übergebene Zeit wird im Backend gespeichert. Es wird die aktuelle Zeit geholt und man rechnet die Differenz der beiden Zeiten aus. Dazu gibt Sie die Zeit als Antwort zurück. | 300 |
| 4 | 4.A | 2.2.24 | Liam | Das Frontend speichert die Zeit und gibt sie aus. | 120 |
| Testen |  | 23.2.24 | Liam | Alle Testfälle erstellen. | 180 |
| Testen |  | 23.2.24 | Liam | Das Testprotokoll erstellen | 30 |
| Testen |  | 23.2.24 | Liam | Das Programm testen, Testprotokoll ausfühlen und Testbericht verfassen. | 180 |
| Portfolio |  | 23.2.24 | Liam | Portfolio schreiben und abgeben | 225 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# Entscheidung

Erstellen Sie das Programm und die notwendigen Dokumente. **Erstellen Sie das Programm selbstständig und verwenden Sie nicht einfach das Resultat eines Lernvideos oder fertigen Code.**

# Realisieren

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nr. Arbeitspaket** | **Datum** | **Beschreibung** | **Zeit (geplant)** | **Zeit (effektiv)** |
| 1.A | 19.1.24 | Man erstellt ein Programm, das nach einer Zeit fragt und die speichert. | 400 |  |
| 2.A | 19.1.24 | Man erstellt ein Backend, dass durch das Programm aufgerufen werden kann und eine Zeit übergeben kann. | 50 |  |
| 3.A | 19.1.24 | Die übergebene Zeit wird im Backend gespeichert. | 320 |  |
| 3.B | 26.1.24 | Das Backend holt sich die aktuelle Zeit und speichert sie auch. | 50 |  |
| 3.C | 26.1.24 | Das Backen rechnet die Differenz Zeit der beiden Zeiten aus. | 225 |  |
| 3.D | 2.2.24 | Das Backend schickt die Zeitdifferenz ans Frontend. | 100 |  |
| 4.A | 2.2.24 | Das Frontend speichert die Zeit und gibt sie aus. | 100 |  |
| Testen | 23.2.24 | Alle Testfälle erstellen. | 180 |  |
| Testen | 23.2.24 | Das Testprotokoll erstellen | 30 |  |
| Testen | 23.2.24 | Das Programm testen, Testprotokoll ausfühlen und Testbericht verfassen. | 180 |  |
| Portfolio | 23.2.24 | Portfolio schreiben und abgeben | 225 |  |

# Testen

## Testfälle

**Testfallnummer:** 1.1

**Anforderungsnummer:** Anforderung 1

**Voraussetzungen: Das Programm ist geöffnet und startet**

**Eingabe: 1) Gehen Sie bei Swagger auf die Post Request.**

**2)Klicken auf den Bouten «Try out».**

**3)Geben Sie dieses Datum ins Input Feld «2024-01-19 15:30:00».**

**4)Drücken Sie den Bouten «Execute»**

**Ausgabe: 5)**

**Ein Bild, das Text, Screenshot, Software, Zahl enthält.

Automatisch generierte Beschreibung**

**Testfallnummer:** 4.1

**Anforderungsnummer:** Anforderung 3

**Voraussetzungen: Testfall 1.1**

**Eingabe: -**

**Ausgabe:** 5) **Ein Bild, das Text, Screenshot, Software, Webseite enthält.

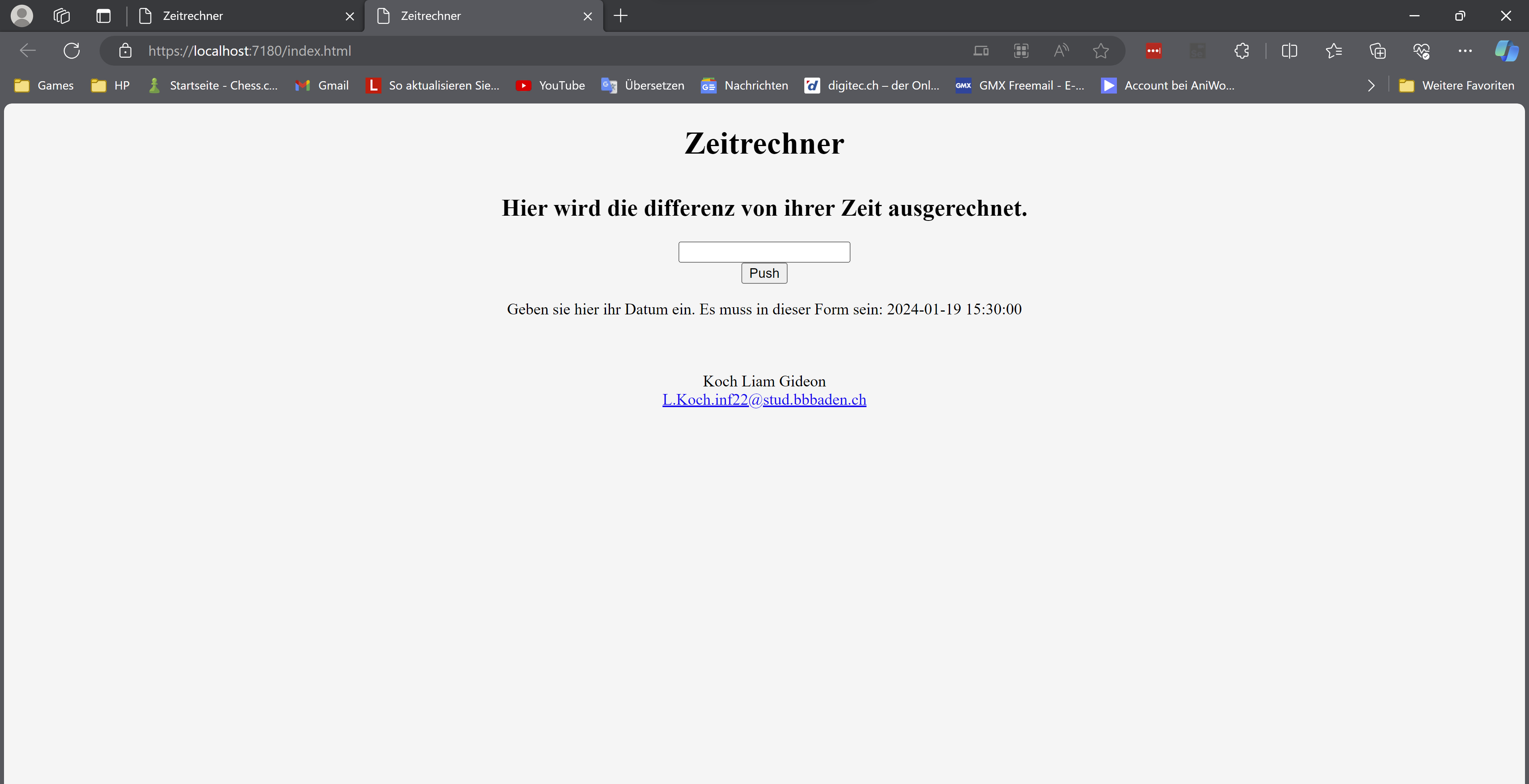
Automatisch generierte Beschreibung**

**Testfallnummer:** 4.1

**Anforderungsnummer:** Anforderung 4

**Voraussetzungen: Testfall 2.1**

**Eingabe: 1) Öffnen Sie die Webseite über** <https://localhost:7180/index.html>.

**Ausgabe:** 2)****

**Testfallnummer:** 3.1

**Anforderungsnummer:** Anforderung 2

**Voraussetzungen: Testfall 1.1**

**Eingabe: 1) Geben Sie in das Input Feld diese Zeit ein «2024-01-19 15:30:00»**

**2) Drücken Sie Push**

**Ausgabe:** 2) Es wird eine Zeit unter dem Text «Geben sie hier ihr Datum ein. Es muss in dieser Form sein: 2024-01-19 15:30:00» angezeigt.

Testprotokoll

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test-Nr. | Testfall-Nr. | Datum | Tester | Resultat | Bemerkung | Unterschrift |
| 1 | 1.1 | 15.12.23 | Liam Koch | OK |  |  |
| 2 | 2.1 | 15.12.23 | Liam | OK |  |  |
| 3 | 3.1 | 15.12.23 | Liam Koch | OK |  |  |
| 4 | 4.1 | 15.12.23 | Liam Koch | NOK | Die Zeit wird nicht vom Backend entgegen genommen, da es kein DateTime ist. |  |
|  |  |  |  |  |  |  |

## 

## Testbericht

Das Backend und das Frontend funktionieren, nur die Kommunikation nicht. Das Backend nimmt nicht den Input vom Frontend an und gibt somit auch keinen Wert zurück. Man müsste das Backend so überarbeiten, dass es andere Werte annimmt.