**Rock-Paper-Scissors-Game**

**Web-App**

1. **Einführung**

Ich habe im Rahmen vom Modul 150 ein Project meiner Wahl gestalten können.   
Ich habe eine Web-App entwickelt, mit Hilfe welcher mein Online das klassische «Schere, Stein, Papier»-Spiel online gegen andere Spieler oder gegen einen **BOT** spielen kann.

1. **Technologie-entscheid**

Da ich mich bei der Arbeit mehrheitlich mit C# arbeite, wollte ich eine Abwechslung zum Arbeits-alltag und habe mich somit für eine Web-App entschieden.  
Anfangs wollte ich meine App auf der Bootstrap-Basis gestalten, doch mein Mitschüler «David Gataric»-welcher lange Erfahrung mit Web-Apps hat, da er das bereits bei der Arbeit überwiegend Web-Apps entwickelt, hat mir geraten, das Projekt mit Angular umzusetzen.   
Ich habe ich mich dazu entschieden, den Ansatz «**Mobile-First»** zu verfolgen, weil ich mein Produkt vorwiegend für Handy-Benutzer gestalten will. In der Schule wurde uns im ÜK 335 beigebracht, dass es sich lohnt, Mobile-First zu entwickeln, da es einfacher sein wird, die Information auf einem grösseren Bildschirm darzustellen, als sie übersichtlich auf einer kleineren Fläche darzustellen.

1. **Management-Tools**

Für mein Project habe ich eine verschiedene Management-Tools verwendet. Diese sowie auch meine Entwicklungs-umgebung, beschreibe ich detaillierter in diesem Kapitel.

* 1. **GitHub**

Den Source-Code pflege ich mit Hilfe von Git-hub. Git-hub ist eine Entwicklungs-Code Management Tool

* 1. **Trello**

Im Git hub Reporsitory, verlinke ich ebenfalls auf mein **ProductBacklog** imTrello verlinke. Trello verwende ich aus dem Grund, dass mir dieses Tool in einem der Überbetrieblichen-Kurse vorgestellt worden ist. Da dieses Tool gratis zur Verfügung steht und ich bereits einmal damit gearbeitet habe, habe ich mich für dieses entschieden.

* 1. **Entwicklungs-umgebung**

Als Entwicklungs-umgebung, habe ich mich für Visualstudio-Code entschieden, da dies die geeignetste Umgebung fürs Arbeiten an Web-Projekten ist. Dort kann ich verschiedene Formatierungs-Extensions herunterladen und es enthält eine Benutzeroberfläche für das Verwalten von Git-Befehlen.

1. **Vorgehen**

Ich habe damit angefangen, dass Default-projekt der Angular CLI anzupassen. Ich habe alles Überflüssige (was für mein Projekt nicht notwendig ist) gelöscht. Ich habe bloss den Header, die Buttons und den footer sowie jeweils die styles übriggelassen. Ich habe anschliessend meine Bilder, welche ich im Internet ausgesucht habe, als Hintergrund der jeweiligen Buttons gesetzt.

* 1. Material Design

Ich habe als nächstes das Material Design für meine Applikation konfiguriert. Ich habe es per Terminal im Projekt mit folgendem command hinzugefügt:  
ng add @angular/material

* 1. Projektmanagement

Bei der Projektmanagement-Methode habe ich mich für Scrum entschieden.   
Scrum eignet sich vor allem für Projekte, welche unendlich-langen Entwicklungsprozess haben.  
Da mein Produkt später auf einem Server gehostet wird und ich in der Zukunft vor habe, das Spiel zu erweitern (mehr Elemente, nebst Schere, Stein und Papier), eignet sich diese Methode bestens.  
Ich habe dabei mich selber als Product Owner, Scrum Master und Entwickler.   
Da wir Wöchentlich unseren Fortschritt der Klasse präsentieren müssen, habe ich die Sprint-Dauer auf «Wöchentlich» gesetzt. Das bedeutet das ich wöchentlich einen Stand haben muss, wo ein Teilstück der Arbeit funktionell vollständig abgeschlossen ist.

Copyright:

Rock: <https://www.flickr.com/photos/32279598@N02/33507875820>

Paper: <https://pxhere.com/en/photo/869916>

Scissors : <https://www.wannapik.com/vectors/3081>

**Arbeits-Journal**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Datum | Was habe ich gemacht? | Was habe ich gelernt | Wo hatte ich Schwierigkeiten? | Was ist pendent? |
| 12.11.2020 | Ich habe das PB erstellt | Ich muss das PB viel einheitlicher Definieren (ganze Funktionen als PBI, nicht nur Teile davon) | Ich habe das PB zu wenig genau definiert.  Die Funktionalität vom Produkt ist noch zu mager | Ich muss die PBI anpassen |
| 12.11.2020 | Ich habe einen Technologie-Entscheid getroffen (Bootstrap) | Ich hatte durch fehlendes Wissen, nicht von Anfang an die richtige Technologie gewählt. | Ich habe mich nach langem überlegen doch gegen Bootstrap entschieden, da ich die App Mobiele-User freundlich gestalten wollte. Nach einer Diskussion mit Arbeits-Kollegen, habe ich mich für eine Angular-pwa entschieden. | Ich muss die neue Technologie anlegen, da Bootstrap für mich nicht mehr aktuell ist. |
| 12.11.2020 | Ich habe ein leeres Git Repo erstellt | Ich habe dies schon mehrere Male gemacht | - | Ich muss das Repo stehts mit Commits pflegen |
| 19.11.2020 | Ich habe das Angular Start-Projekt heruntergeladen und pwa extension hinzugefügt | - | - |  |
| 19.11.2020 | Ich habe das .yml File erstellet, welches für mich automatisch ein build anstosst | Wie man eine Workflow Datei im Github hinzufügt | Die Syntax war für mich bis anhin unbekannt und ich musste mich erst mit dieser auseinandersetzen. | - |
| 19.11.2020 | Ich habe einen init Commit ins Repo gepusht | - | - | - |
| 26.11.2020 | Ich habe passende Bilder für meine Buttons gesucht | - | - | Ich muss diese Bilder als Hintergrund für die Buttons verwenden |
| 26.11.2020 | Ich habe die Dokumentation weiter gepflegt | Ich sollte die Dokumentation immer wieder anpassen. So habe ich weniger Pendenzen. |  | Immer weiter pflegen. |
| 03.12.2020 | Ich habe das default-Projekt von Angular soweit angepasst, dass ich nur das nötigste für mich übrig gelassen habe |  | Ich habe anfangs zu viel gelöscht, was die ganze Formatierung beeinflusste | - |
| 03.12.2020 | Ich habe die passenden Bilder für das GUI herausgesucht und als Hintergrund bei den Buttons gesetzt. |  | Ich habe es nicht geschafft, die grösser der Bilder passend darzustellen. | Ich werde bei Gelegenheit mich mit der Platzierung der Bilder befassen. |
| 10.12.2020 | Ich habe die Syles der Site angepasst (Schrift, Titel, Schriftgrösse, Schriftfarbe etc.) angepasst |  | Ich musste erst wieder die HTML Syntax nachschauen, da ich diese wegen mangelhaften nutzen vergessen habe. |  |
| 10.12.2020 | Ich habe das mat-toggle-slide Modul hinzugefügt. | Wie man ein Modul ins Angular integriert | Ich konnte die Labels beim «toggle» nur jeweils auf eine Seite platzieren (davor oder danach) |  |
| 17.12.2020 | Ich habe das Score-board für mein Spiel designed | Wie man Daten aus dem Backend im Frontend darstelle. | Mit dem CSS (design): Das darstellen vom Scoreboard, wie ich es mir vorgestellt habe, erforderte viel CSS. | Ich muss die Logik hinter dem Scoreboard implementieren. |
| 17.12.2020 | Ich habe Spiel-dialoge sowie Resultat-Dialoge mit Sweetalert erstellt. | Wie ich schöne Dialoge im Web darstellen kann (nicht mit «alert ()» Befehl) | Ich habe lange gebraucht, um zu verstehen, wie ich das Modul richtig nutzen kann | - |
| 17.12.2020 | Ich habe die Logik für das Scoreboard und somit für die Logik im Spiel implementiert. | Ich kann eine einfache Logik im typescript umsetzen. | Das Scoreboard hat bei mir anfangs nicht aktualisiert. Es ignorierte teilweise eine Vielzahl an Runden. | - |
| 20.12.2020 | Ich habe weiter an der Dokumentation gearbeitet. |  |  |  |