

Teamprojekt SS2022

A Web-Platform for Turn-Based Games

Team – wir stellen uns vor

"Robin Bonkaß <robin.bonkass@student.uni-tuebingen.de>",
"Alexander Löwe <alexander.loewe@student.uni-tuebingen.de>",
"Nicole Bretz <nicole.bretz@student.uni-tuebingen.de>",
"Dennis Bonna <dennis.bonna@student.uni-tuebingen.de>",
"Mario Laux <mario.laux@uni-tuebingen.de>",
"Timon Höfer <timon.hoefer@uni-tuebingen.de>",
"Emma Urban <emma-rebecca.urban@uni-tuebingen.de>",
"Elias Altmann <elias.altmann@student.uni-tuebingen.de>",
"Thomas Vogel <thomas.vogel@studnet.uni-tuebingen.de>",
"Maleck Ramadan <mohammed-maleck.ramadan@student.uni-tuebingen.de>",
"Lutz Warnecke <lutz.warnecke@student.uni-tuebingen.de>",
"Janina Roth <janina.roth@student.uni-tuebingen.de>"

1. Phase - Teaching

Grundlagen zu Typescript

OOP

Trennung von Model View Container

Mockups

State machines

PubSub-Konzept

Testen mit Mocha

CSS Grundlagen

Socket.io

1. Phase – Implementierung von Nim

Gelerntes wurde umgesetzt

Ergänzende Aufgaben zwischen meetings

Festigen der gelernten Konzepte

2. Phase – Team übernimmt Planung

Scrum Meetings

Freie Spielwahl

Aufgabenverteilung

Probleme im Team lösen

Playful

Homepage

Erstes Spiel: 4 Gewinnt

Zweites Spiel: TicTacToe

Achievements

Homepage (Aufbau und Anforderung)

Login/Register

Statistik speichern

Human vs Human

Human vs AI

4 Gewinnt - UI

Styling (Hover mit eingeblendetem Stein)

4 Gewinnt - Logic

State machine

4 Gewinnt - AI

Zwei Schwierigkeitsstufen

Minmax Algorithm (depth first search, depth limited search)

Tic Tac Toe - UI

Tic Tac Toe - Logic

State Machine

Tic Tac Toe - AI

Zwei Schwierigkeitsstufen

Einfach: depth limited search tiefe von 1

Schwer: Minmax Algorithm

Datenbank

Anforderungen

Nutzen

Probleme

Styling

Verworfenne Ideen

Ausblick
