Road to Reversi

**23.05.2019**

Jonas Pfingst, Joshua Purwin, Sebastian Greger

Die Spannende Reise von 3 jungen Programmieren

Inhalt

[Ausgangssituation 2](#_Toc9376094)

[Projektziel 2](#_Toc9376095)

[Projektumfeld 3](#_Toc9376096)

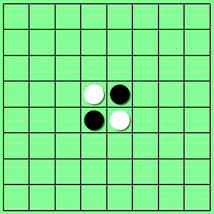
[Ressourcen und Ablaufplanung 3](#_Toc9376097)

[Personalplanung 3](#_Toc9376098)

# Ausgangssituation

## Projektziel

Zum Ende des ersten Lehrjahres haben wir uns zum Abschluss im Lernfeld 6 nochmal einem Projekt gewidmet. Das Ziel dieses Projektes war es ein funktionierendes Reversi Spiel in C zu Programmieren. Bei Reversi handelt es sich es sich um ein zwei-Spieler Strategiespiel welches auf einem 8 mal 8 großen Spielfeld gespielt wird. Zu Beginn des Spieles liegen vier Steine in vorgegebener Anordnung auf dem Spielfeld.



https://learnplaywin.net/reversi-rules/

Nun platzieren beide Spieler abwechselnd einen Stein ihrer Farbe auf dem Spielfeld und versuchen gegnerische Steine mit den eigenen Steinen einzuschließen. Befindet sich ein Stein zwischen gegnerischen Steinen wird dieser umgedreht und gehört nun zu den gegnerischen Steinen. Gewinner des Spiels ist derjenige, der am Ende die meisten Steine sein Eigen nennen kann. Wichtig zusagen ist noch, dass Steine nicht einfach willkürlich platzier werden dürfen, sondern immer horizontal, vertikal oder diagonal an ein bereits belegtes Feld angrenzen müssen. .

Unsere Mission war es nun, dieses Spiel zu Programmieren. Dabei sollte jeder Spieler abwechselnd an der Reihe sein, einen Stein zu platzieren. Der aktuelle Punktestand sowie welcher Spieler aktuell am Zug ist soll dem Benutzer jederzeit angezeigt werden.

Außerdem soll es einen Timer geben, der die Laufzeit des aktuellen Spiels anzeigt. Zudem soll es jederzeit möglich sein das Spiel zu pausieren und den Timer anzuhalten. Für den Benutzer soll es unmöglich sein Falscheingaben zu machen, außerdem soll der Spieler eine Runde passen können, wenn für ihn kein Zug möglich ist. Das Spie soll automatisch beendet werden wenn kein weiterer Zug mehr möglich ist.

## Projektumfeld

Wie bereits oben erwähnt ist die vorgegebene Programmiersprache für unser Projekt C. Als Entwicklungsumgebung ist CodeBlocks vorgegeben, als Compiler der MinGW-Compiler.

Des Weiteren soll das Programm sinnvoll gegliedert sein und der Quellcode sollte in einzelne Funktionen und Dateien ausgelagert werden, sodass das Programm hinterher auch durch den Quellcode nachvollziehbar ist und Änderungen einfacher zu implementieren sind. Hierzu soll auch jede einzelne Funktion einen Kommentar haben, welcher beschreibt was die Funktion genau macht, welche Übergabeparameter sie besitzt, sowie was für einen Wert/Werte sie gegebenfalls zurückgibt.

Die Benutzung von zusätzlichen Bibliotheken ist erlaubt. Die letzte Vorgabe ist noch, dass der Compiler beim Kompilieren keinerlei Warnungen ausspuckt.

# Ressourcen und Ablaufplanung

## Personalplanung

Um möglichst möglichst effizient voranzukommen haben wir uns in der ersten Stunde nach Projektbeginn erstmal zusammengesetzt und gemeinsam überlegt was überhaupt zu tun ist. Hier haben wir uns erstmal eine grundsätzliche Struktur erstellt nach welcher wir vorgehen. Wir haben uns entschieden das Programm in die einzelnen Dateien board, gamelogic, input und output aufzuteilen. Jonas wurde damit betraut sich ausschließlich um die gamelogic, also die eigentliche Spiellogik, zu kümmern. Er sollte sich hierzu genau mit den Regeln des Spiels vertraut machen und Algorithmen für die einzelnen Spielvorgaben entwickeln. Joshua und Sebastian sollten sich um jegliches „Drumherum“ kümmern. Dies beinhaltet mehrere Aspekte:

* Den Aufbau des Spielfeldes
* Der Ausgabe inklusive folgender Aspekte:
  + Das Spielfeld selber
  + Einem Timer mit aktueller Spielstand
  + Der aktuelle Punktestand
  + Welcher Spieler ist aktuell am Zug
* Die Eingabemöglichkeiten für den User:
  + Dem Spieler die Möglichkeit geben einen Stein auf dem Spielfeld zu platzieren
  + In der Lage sein das Spiel zu pausieren
  + Das Spiel jederzeit beenden zu können

## Terminplanung

Bei der Terminplanung hingegen haben wir versucht relativ flexibel zu bleiben. Wir haben beschlossen uns am Anfang jeder Unterrichtsstunde kurz zusammenzusetzen um den aktuellen Zwischenstand sowie die noch zu erledigenden Aufgaben zu besprechen. Daraufhin hat jeder an seinem aktuellen Thema gearbeitet und dieses gegebenfalls mit nach Hause genommen um dort noch daran weiterzuarbeiten. Zudem haben wir eine gemeinsame Whatsapp-Gruppe erstellt um uns jederzeit austauschen zu können oder den anderen Gruppenmitgliedern bei offenen Fragen jederzeit zur Hilfe kommen zu können. So standen wir permanent im gemeinsamen Austausch und konnten auch zu jeder Zeit neue Ideen oder Anregungen diskutieren.

## Kostenplanung

|  |  |  |
| --- | --- | --- |
| **Datum** | **Aufgewendete Zeit in Stunden** | |
| 29.04.2019 | 4,5 | |
| 07.05.2019 | 9 | |
| 12.05.2019 | 1 | |
| 13.05.2019 | 4,5 | |
| 21.05.2019 | 6 | |
| 22.05.2019 | 5 | |
| 23.05.2019 | 4 | |
| **Kosten pro Stunde** | | 60€ | |
| **Gesamtkosten** | | 2040€ | |

# Durchführung und Auftragsbearbeitung