# Vorgehensmodell

Auf Grund des geringen Entwicklungsaufwands unserer Softwareentwicklung entschieden wir uns, zur Durchführung unseres Projektes das Wasserfallmodell zu nutzen.

Das Wasserfallmodell zeichnet sich durch einen gradlinigen Ablauf aus.

Abbildung 1 - Wasserfallmodell (Eigene Darstellung)

Wie im Wasserfallmodell beschrieben, begannen wir unsere Projektarbeit mit der Analyse der Anforderungen. Hierzu besprachen wir zuerst die Anforderungen, die aus der Aufgabenstellung resultierten. Nach dem wir diese zusammengetragen hatten, ergänzten wir diese Anforderungen um persönliche Wünsche. Diese umfassten unter anderem, die Möglichkeit die Startparameter zu variieren und eine statistische Auswertung während dem Programmablauf. Dies sollte die Analyse der Programmaufgabe vereinfachen.

Daraus entstand unser Lastenheft.

Im Anschluss beschäftigten wir uns mit der Systemlandschaft. Hierbei lag das Hauptaugenmerk auf der Wahl der Programmiersprache. Der wichtigsten Faktoren hierbei waren die vorhandenen Kenntnisse innerhalb des Projektteams und die einfache Umsetzbarkeit der Anforderungen.

Weil die meisten von uns mit der Webentwicklung (HTML, PHP, JavaScript & CSS) bereits Erfahrungen gesammelt hatten entschieden wir uns für dafür unser Projekt in Form einer Webanwendung umzusetzen.

Ein weiterer Vorteil der Webentwicklung sind die vielen vorgefertigten Module. Diese ermöglichen vor allem im Frontend-Bereich eine schnelle und unkomplizierte Implementierung.

Nach Abschluss der Analyse der Systemanforderungen begannen wir mit der Entwicklung. Um eine Kollaboration und Versionierung der Entwicklung zu ermöglichen, entschieden wir uns dazu Gitlab einzusetzen.

Nach Abschluss der Entwicklung, wurde das Programm nach festgelegten Kriterien getestet.

Hierbei wurden vor allem auf die Korrektheit der Programmausgabe und die richtige Funktionsweise der Funktionen geachtet. Des Weiteren wurde das Programm auf unterschiedlichen Systemen getestet.

Nach Abschluss der Entwicklung nach dem Wasserfallmodell, führten wir einen weiteren Durchlauf nach dem Wasserfallmodell durch. Bei diesem Durchlauf führten wir einen Soll – Ist Vergleich durch, um sicherzustellen, dass unser Programm korrekt umgesetzt ist. Im Anschluss an den Soll – Ist Vergleich führten wir die notwendigen Änderungen durch. Zuletzt folgten erneut Tests um die Korrektheit unseres Programmes sicherzustellen.

# Problembeschreibung

Die Aufgabe bestand darin, ein Programm zu entwickeln das einen Roboter simuliert. Dieser Roboter sollte zwei verschiedene Aktionen besitzen, die in Abhängigkeit von der Position ausgeführt werden.

Als „Spielbrett“ dient eine dynamische Fläche von Feldern. Diese können entweder die Farbe Schwarz oder Weiß annehmen. Abhängig von der Farbe des Feldes, auf dem sich der Roboter befindet, soll eine Aktion ausgeführt werden.

Ist das Feld Schwarz so soll der Roboter sich um 90 Grad nach links drehen, die Feldfarbe invertieren und in Blickrichtung auf das nächste Feld vorrücken.

Ist das Feld Weiß, so soll der Roboter sich um 90 Grad nach rechts drehen, die Feldfarbe invertieren und in Blickrichtung auf das nächste Feld vorrücken.

Eine weitere Anforderung an das Programm, war die Möglichkeit, das Verhalten des Algorithmus untersuchen zu können. Dazu soll eine statistische Auswertung implementiert werden.

Anforderungsprofil

* Ausgabe der Anzahl der Iterationen
* Eingabe der Anzahl der Iterationen
* Ausgabe der Schritte
* Timeout zwischen den Einzelnen Schritten
* Start Stop
* Variable Startbedingungen