# Arbeitsplanung:

Die Arbeiten der jeweiligen Milestones werden agil an die bestgeeigneten Person vergeben. Die bestgeeignete Person zeichnet sich dadurch aus, dass sie über das jeweilige Thema Kenntnisse hat und nicht mit anderen Tätigkeiten beschäftigt ist. Zeitbedingungen innerhalb der Milestones haben wir nicht.

## M0 Prototyp:

Für den Prototypen brauchen wir ein Klassendiagramm, welches in Umlet realisiert wird. Einen Dokumentationsprototypen muss auch schon erstellt werden. In diesem werden alle Arbeitsschritte genau dokumentiert.

Für die Programmierung brauchen wir jemanden der das GUI macht und jemanden der die ganze Logik im Hintergrund realisiert. Die Schnittstelle zwischen GUI und Logik wird im Klassendiagramm bereits definiert.

## M1 Basisversion

Die Basisversion wird auf dem Prototypen aufgebaut. An dem GUI wird nun nichts mehr geändert. Die Dokumentation des Prototypen muss vervollständigt und während des Ausbaus detailliert erweitert werden. Jemand muss das komplette Regelwerk in der Logik einbauen. Auch müssen Tests geschrieben werden, um die Basisversion zu kontrollieren.

Der Basisversion wird eine künstliche Intelligenz hinzuprogrammiert.

Jemand muss den Spielmodus Computer vs. Mensch implementieren.

## M2 Ausbaustufe 1

Je nach verfügbaren Ressourcen wird ein Highscore implementiert. Dies soll jemand auf dem GUI anzeigen lassen. Nun muss ein GUI für das Spieleinstellungsmenü von jemandem erstellt werden, welches von der anderen Person die Logik dazu programmiert wird. Töne müssen nun für die Aktionen ausgesucht und implementiert werden.

Jemand programmiert die Computerintelligenz weiter, sodass es möglich wird im Spieleinstellungsmenü eine Schwierigkeitsstufe einzustellen, um so seinen Level anzupassen.

Jemand muss den Spielmodus Computer vs. Computer implementieren.

Alle neuen Funktionalitäten müssen von der jeweiligen Person detailliert in der bestehenden Dokumentation festgehalten werden.

## M3 Ausbaustufe 2

Je nach verfügbaren Ressourcen wird das spielen über das LAN ermöglicht. Für dies muss man das Klassendiagramm zuerst erweitern.

Jemand implementiert die Funktion „Undo“, welche dazu dient eine Aktion rückgängig zu machen. Die Aktion „Undo“ soll mit der Tastenkombination „Ctrl + Z“ realisiert werden.

Jemand muss nun die Funktion „Spielaufzeichnung speichern“ hinzufügen. Gespeicherte Spielstände sollten wieder abgespielt werden können. Dementsprechend muss eine History von jemandem implementiert werden.

Schlussendlich muss noch von jemandem die Funktion „Spielstand speichern“ implementiert werden.

Alle neuen Funktionalitäten müssen von der jeweiligen Person detailliert in der bestehenden Dokumentation festgehalten werden.

## Zeitplan mit Meilensteinen:

