**Dokumentation JMAL-X3 – Alles hört auf Morsecode**

1. **Ziel des Projekts**

Das Ziel unseres Projekts JMAL-X3 ist, dass ein Anwender eine Eingabe morst und der Drucker die Eingabe intern umwandelt zu natürlicher Sprache. Danach soll die Eingabe in natürlicher Sprache auf ein DIN-A4-Blatt gedruckt werden. Dabei soll es noch für den Anwender möglich sein, einzelne Buchstaben oder Wörter zu löschen, falls er etwas Falsches eingibt, was aber ein gültiges Morsezeichen ist.

1. **Rollen- und Aufgabenverteilung in der Gruppe**
   1. **Allgemeine Aufteilung:**

Das Projekt lässt sich einfach in zwei große Unterbereiche aufteilen: Drucker und Morsen. Für beide Unterbereiche haben wir noch jeweils zwei Subbereiche uns überlegt: Software und Hardware. Die Aufgabenverteilung war wie folgt:

* 1. **Drucker:**

Für den Drucker und dessen Hardware war Matja zuständig. Seine Hauptaufgabe bestand darin einen Drucker mit beweglicher X-Achse, Y-Achse und Stifthalterung mit Bewegung auf der Z-Achse zu bauen. Zusammen mit Anne haben sich die beiden um die Software des Druckers gekümmert. Somit war ihre Aufgabe, sich zu überlegen, wie der Stift sich auf den Achsen bewegen lässt und die natürliche Sprache lesbar auf das Blatt zu drucken. Im weiteren Verlauf werden Anne und Matja auch das Drucker-Software-Team genannt.

* 1. **Morsen:**

Für den Bau des Morsetasters war Jonathan zuständig, seine Aufgabe bestand darin ein stabiles Gestell für den Berührungssensor mit einem Taster zum Drücken zu bauen. Des Weiteren hat er sich mit Lea die Aufgabe geteilt die Software für den Morsetaster zu schreiben. Dabei wurde die Aufgabe nochmal leicht aufgeteilt. Jonathan schrieb den Code, um aus dem Input die Berührungsdauer zu berechnen und diesen in Morsecode umzuwandeln. Das Übersetzen des Morsecodes in Buchstaben hat Lea übernommen, sowie die Funktion einzelne Buchstaben und Wörter zu löschen. Gemeinsam haben sie an der Speicherung in die Queue gearbeitet und den Code gedebuggt. Im weiteren Verlauf werden Jonathan und Lea auch das Morsen-Software-Team genannt.

* 1. **Sonstiges:**

Das Protokoll wurden in Teamarbeit mit Allen erstellt. Die 1. Präsentation wurde von Lea erstellt, der Zeitplan wurde von Anne beigesteuert, und gehalten. Die Dokumentation wurde in ihrem ersten Entwurf von Lea übernommen. TODO: ändern/updaten

1. **Projektplan und tatsächlicher Ablauf**
   1. **Projektplan:**Ein Bild, das Text, Screenshot, Diagramm, parallel enthält.

      Automatisch generierte Beschreibung
   2. **Tatsächlicher Ablauf:**

Tatsächlich haben wir für die meisten Teilaufgaben kürzer gebraucht als wir geplant hatten, dadurch sah unser tatsächlicher Ablauf so aus:

Innerhalb der 1. Woche stand das Konzept.

Matja hatte eine Anleitung für einen Drucker aus dem Internet, für die uns nur weniger Teile gefehlt haben. Für den Morsetaster hatten wir ein Bild als Bauanleitung. Nur das Einrichten von Git hat gedauert bis einschließlich Woche x (5?).

Ab der zweiten Woche konstruierte Matja den Drucker und wurde damit am Ende der 4. Woche fertig. Zeitgleich zur Konstruktion begann Anne mit dem Programmieren der Software für den Drucker. (Ende der Software war wann?) Ab der 5. Woche konnten die ersten Buchstaben erfolgreich gedruckt werden.

Die Konstruktion des Morsetasters hatte Jonathan innerhalb kürzester Zeit, am 13. Mai, fertiggestellt. In Woche 5 wurde der Taster an den Drucker angebaut. Zeitgleich mit dem Software-Team vom Drucken fing das Software-Team vom Morser, Jonathan und Lea, damit an die benötigte Software zu schreiben. Nach der 4. Woche war die Software so weit fertiggestellt.

Sobald die Software des Druckers fertig war, xx.xx., begannen die Testläufe mit beiden Komponenten.

Weiterschreiben sobald Meilenstein erreicht

1. **Probleme und Herausforderungen**
   1. **Probleme beim Drucker:**

Beim Drucker war klar, dass der Workaround für die fehlenden Teile für Probleme sorgen würde. Wie wurde das gelöst?

Ein weiteres Risiko war, dass das Blatt sich in der Konstruktion verhaken könnte, falls die Motoren der Räder sich nicht synchron bewegen. Wie wurde das gelöst?

Da bei dem Drucker keine glatte Oberfläche an der Stelle ist, an der der Stift schreiben soll, befürchtete man, dass der Stift durchdrücken könnte. Das Drucker-Software-Team löste das, indem es den Stift passend ausrichtete und nicht zu weit nach unten auf der Z-Achse bewegte. Wurde das so gelöst?

Um gut Drucken zu können war die Wahl des passenden Stiftes essenziell. Kugelschreiber hätte zu viel gedruckt gebraucht, also fiel die Wahl schnell auf einen Filzstift, der bei leichtem Kontakt mit dem Blatt schon schreibt. Doch nicht nur der Stift war eine Herausforderung, die Synchronisation der x- und y-Achse durfte nicht fehlschlagen und die Buchstaben nicht zittrig geschrieben werden. Wie wurde das gelöst? Durch stockende klare Bewegungen des Stiftes sind die Buchstaben nicht zittrig, sondern schön lesbar kantig.

* 1. **Probleme bei der Software:**

Für Jonathan bestand die Herausforderung bei dem Bau des Morsetasters darin, dass der Touchsensor Eingaben fehlerhaft messen könnte. Dadurch kann als Konsequenz folgen, dass die Zeiträume für Dit und Dah angepasst werden müssen. Gelöst wurde das Problem, indem Zeittoleranzen in den Code eingebaut wurden.

Eine weitere Herausforderung war, die Einigung zum Umgang bestimmter Sonderfälle. Unser Code verfügt über die Möglichkeit ungültige Morsezeichen zu erkennen und Eingaben zu löschen. So wurde sich gefragt, wenn der Anwender ein ungültiges Zeichen eingibt und dann löschen betätigt, wird dann das ungültige Zeichen gelöscht oder das vorherige Zeichen, weil das ungültige gleich aussortiert wird. Man einigte sich darauf, dass vorherige Zeichen zu löschen und alle ungültigen Zeichen zu ignorieren.

Beim Zusammenspiel von Drucker und Morsen sah man das Problem, dass es Schnittstellenprobleme geben könnte. Da die Queue mit den natürlichen Buchstaben reibungslos an den Drucker übergeben werden muss.

1. **Beschreibung nennenswerter Algorithmen, Konstruktionen etc.**
   1. **Druckerbeschreibung**
   2. **Druckerverhalten**

TODO: Drucker Aufbau und Algorithmus beschreiben

* 1. **Morsetasterbeschreibung**
  2. **Morsetasterverhalten**

Der Morsetaster solle über den Touchsensor bedient werden. Der Anwender soll durch Morsen dem Programm einen Text eingeben können.

TODO: Morsetaster und Algorithmus beschreiben

1. Live-Demo oder Video?
2. TODO: Aufnahmen machen
3. Todo: wir oder man?
4. TODO: Welche Schriftart, Vorgabe des Abstandes, etc.?