**HTL Saalfelden**

**Systemplanung und Projektentwicklung**

****

**Projektdokumentation**

**2024 / 2025**

|  |  |
| --- | --- |
| **Projektbezeichnung** | Mbot Lore |
| **Projektteam** | Johannes Ellmer, Christos Magos, Timo Winkler |
| **Erstellt am** | 01.10.2018 |
| **Letzte Änderung am** | 15.10.2018 |
| **Status** | [in Bearbeitung/fertiggestellt/pausiert/abgebrochen/Prüfung] |
| **Aktuelle Version** | 1.5 |

**Änderungsverlauf**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nr.** | **Datum** | **Version** | **Geänderte Kapitel** | **Art der Änderung** | **Autor** |
| <1> | <01.10.2018> | <1.0> | <Alle> | <Erstellung> | <Max Mustermann> |

**Inhalt**

[1. Allgemeines / Projektübersicht 4](#_Toc965522712)

[1.1 Projektbeschreibung 4](#_Toc1040632592)

[1.2 Projektteam und Schnittstellen 4](#_Toc284604490)

[2. Funktionale Anforderungen 4](#_Toc521691458)

[2.1 Use Cases 4](#_Toc1840059643)

[2.1.1 <Name Use Case 1> 4](#_Toc1384425644)

[2.1.2 <Name Use Case 2> 4](#_Toc2131939576)

[2.1.3 <Name Use Case n> 4](#_Toc1332338971)

[3. Nichtfunktionale Anforderungen 4](#_Toc542409565)

[4. Projektplanung 5](#_Toc1181161558)

[4.1 Variantenbildung 5](#_Toc191553135)

[4.2 Machbarkeitsstudie 5](#_Toc729802831)

[4.3 Allgemeine Planungsinformationen 5](#_Toc2024584587)

[4.4 Projektumfeldanalyse 5](#_Toc502666407)

[5. Softwarearchitektur 6](#_Toc2079537197)

[5.1 Aktivitätsdiagramme 6](#_Toc659892040)

[5.1.1 Aktivitätsdiagramm 1 Name 6](#_Toc1701078734)

[5.1.2 Aktivitätsdiagramm n Name 6](#_Toc1952219604)

[5.2 Sequenzdiagramme 6](#_Toc452400050)

[5.2.1 Sequenzdiagramm 1 Name 6](#_Toc539048854)

[5.2.2 Sequenzdiagramm n Name 6](#_Toc1656117885)

[5.3 Komponentendiagramme 6](#_Toc1549670347)

[5.4 Verteilungsdiagramme 6](#_Toc1782933099)

[5.5 Softwarekomponenten / Programme 7](#_Toc863584749)

[5.5.1 SW Programme 7](#_Toc2059131748)

[5.5.2 SW Komponenten 7](#_Toc2020768912)

[6. Projektdurchführung 8](#_Toc1459545787)

[6.1 Sprint 1 8](#_Toc965912477)

[6.1.1 Sprintplanung 8](#_Toc1238068464)

[6.1.2 Sprint Demo 8](#_Toc1608356264)

[6.1.3 Sprint Retrospektive 8](#_Toc1207657317)

[6.1.4 Sprint Zusammenfassung 8](#_Toc56107529)

[6.2 Sprint 2 9](#_Toc501505208)

[6.2.1 Sprintplanung 9](#_Toc1492261807)

[6.2.2 Sprint Demo 9](#_Toc577798987)

[6.2.3 Sprint Retrospektive 9](#_Toc194081203)

[6.2.4 Sprint Zusammenfassung 9](#_Toc729203803)

[6.3 Sprint n 9](#_Toc562254915)

[7. Installation / Software deployment 10](#_Toc1526420175)

[8. Projektabschluß 10](#_Toc1271613368)

[8.1 Projektzusammenfassung 10](#_Toc1743416473)

[8.2 Attachments 10](#_Toc1717973310)

# Allgemeines / Projektübersicht

## Projektbeschreibung

<Kurzbeschreibung um was es bei diesem Projekt genau geht>.

## Projektteam und Schnittstellen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Rolle(n)** | **Name** | **Telefon** | **E-Mail** | **Team** |
| Produktmanager | XY | 0123456 | xy@muserfirma.de | AB001 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Funktionale Anforderungen

## Ein Bild, das Diagramm, Reihe, Plan, Text enthält. KI-generierte Inhalte können fehlerhaft sein.Use Cases

### Connection:

Der User gibt IP-Adresse und beliebigen Namen des Mbots ein, um sich mit ihm zu verbinden. Hierbei kann der Benutzer die Daten einer vorherigen Session mittels Restore Session abrufen und anschließend eine Verbindung aufstellen. Anschließend wird die Verbindung aufgestellt und überprüft.

### Restore Session:

Bei der Connection-page kann sich der Benutzer sich vergangene Sessions anzeigen lassen bzw. wieder mit diesen Werten verbinden. Dazu muss der Benutzer nur den „Restore Session“-Button betätigen. Danach öffnet sich eine Liste an Sessions, die angeklickt werden können. Wenn das getan wurde, werden die Eingabefelder automatisch ausgefüllt. Dadurch muss der Benutzer sich nicht die IP-Adresse des Mbots, den er schon einmal verwendet hat, merken.

Hier UseCase Diagramm und Beschreibung eintragen.

Für jeden Use Case ein eigenes Kapitel erstellen.

2.1.1 Use Case 1

2.1.2 Use Case 2

2.1.n Use Case n

### Manuelle Steuerung:

Nach dem der Benutzer sich mit dem Mbot erfolgreich verbunden hatte, kann er „manuelle Steuerung“ auswählen. Dabei muss der User den „drive“-Button drücken. Dadurch wird einerseits das manuelle Steuern aktiviert und andererseits im Hintergrund das Aufzeichnen der Route gestartet. Jetzt kann der Benutzer je nach belieben die Farbe und die Geschwindigkeit ändern. Außerdem kann der Benutzer den MBOT steuern. Wenn der Benutzer fertig gefahren ist, klickt er auf stop Drive. Dann wird der Fahrmodus beendet und er bekommt die Anfrage zum Speichern der gefahrenen Route. Dieser Route kann so gespeichert werden und bei bedarf später z.B. beim automatischen Abfahren von Routen wiederverwendet werden.

### Automatisches Fahren

Der Benutzer kann im automatischen Modus, eine gefahrene oder gespeicherte Route laden. Die gespeicherten Routen können per Name über ein Dropdown ausgewählt werden. Nach dem erfolgreichen Speichern der Route kann diese ausgewählt werden und nach dem Betätigen des Start-Buttons beginnt der MBot die Route abzufahren.

### Route definieren

Um eine Route zu definieren kann der User entweder eine gefahrene Route speichern oder beim automatischen Fahren ein Fenster öffnen, in dem verschiedene Checkpoints als Liste für eine Route gespeichert werden. Dabei können Dauer, Geschwindigkeit, Farbe und Richtung innerhalb dieses Abschnittes gespeichert werden. Diese Route kann später ausgewählt und abgefahren werden.

### Farbe ändern:

Um die Farbe zu ändern, muss zuerst die Farbe eingeschalten werden. Dazu muss der Benutzer das „toggle“ der Farbe auf on stellen. Danach soll er auf das Quadrat daneben drücken. Darin wird die aktuell ausgewählte Farbe angezeigt. Bem Click darauf, öffnet sich der ColorPicker, in dem eine beliebige Farbe des RGB-Schemas ausgewählt werden kann.

### Katografieren:

# Nichtfunktionale Anforderungen

Gibt es spezielle „Nichtfunktionale Anforderungen“ so sind diese in diesem Kapitel anzugeben. z.B.: Verwendetes System, zu erwartender Speicherbedarf, zu erwartende Anzahl an Transaktionen, ....

# Projektplanung

In diesem Kapitel sollen grundlegende Fragen geklärt werden bevor mit der Projektdurchführung begonnen wird. Obwohl es klar ist, das dieses Projekt mit Hilfe von agilen PM Methoden durchgeführt wird, sollen vorab grundsätzliche Entscheidungen geklärt werden wie z.B. verwendete Programmiersprache, Variantenbildung, Betriebssystem, ...

Manche dieser Punkte können auch auf Grund der gegebenen Anforderungen als „gesetzt“ hingenommen werden.

## Variantenbildung

Auf Basis der Projektanforderungen, welche Varianten für die Umsetzung wurden ausgearbeitet und welche wurde schließlich gewählt und warum. Welche Varianten wurden verworfen und aus welchem Grund.

## Machbarkeitsstudie

Gab es einzelne Punkte, die vorher grob getestet wurden um zu Zeigen das die Umsetzung grundsätzlich möglich ist.

z.B. Kommunikation mit einem MQTT-Server mit Hilfe der Programmiersprache Python, ...

## Allgemeine Planungsinformationen

Andere Planungsinformationen welche nicht durch andere Kapitel abgedeckt sind werden hier eingetragen.

## Projektumfeldanalyse

Führen Sie eine Analyse des Projektumfeldes durch. Welche vergleichbaren Produkte gibt es bereits am Markt. Wie erfolgt die Abgrenzung zu diesen bereits bestehenden Produkten? Wer sind die relevanten Stakeholder des Projektes.

# Softwarearchitektur

In diesem Kapitel soll der Aufbau der Software/Hardware beschrieben werden.

aus welchen Komponenten besteht das SW-Produkt.

Wie interagieren die einzelnen Komponenten miteinander.

Auf welcher Hardware läuft das System bzw. handelt es sich vielleicht sogar um ein verteiltes System. Wie kommunizieren diese Komponenten miteinander.

## Aktivitätsdiagramme

### Aktivitätsdiagramm 1 Name

Beschreibung der entsprechenden Aktivität und einfügen des Aktivitätsdiagramm

### Aktivitätsdiagramm n Name

Beschreibung der entsprechenden Aktivität und einfügen des Aktivitätsdiagramm

## Sequenzdiagramme

### Sequenzdiagramm 1 Name

Beschreibung der entsprechenden Sequenz und Einfügen des Sequenzdiagramm

### Sequenzdiagramm n Name

Beschreibung der entsprechenden Sequenz und Einfügen des Sequenzdiagramm

## Komponentendiagramme

Aus welchen Komponenten besteht die Software und welche Schnittstellen bieten diese an.

## Verteilungsdiagramme

Ein Bild, das Screenshot, Text, Diagramm, Multimedia-Software enthält.

KI-generierte Inhalte können fehlerhaft sein.

Zeigt an wie die einzelnen Teile der Software auf die Hardwarekomponenten verteilt sind und wie die Hardwarekomponenten miteinander verbunden sind.

Auf welchem Rechner läuft welcher Software. Wie sind diese über ein Netzwerk miteinander verbunden.

## Softwarekomponenten / Programme

### SW Programme

Auflistung aller verwendeten SW Programme die bei der Umsetzung des Projektes verwendet worden sind. inkl. Angabe der Versionsnummer

z.B.: Visual Studio 2022,...

### 5.5.2 SW Komponenten

Auflistung aller verwendeten SW Komponenten welche für den Betrieb der SW benötigt werden. z.B.: Java Version, Apache Webserver, DotNet Framework, SW Library XY

inkl. Versiosnummer, Hersteller, Bezugsquelle (Downloadlink, ...) und SW-Lizenz (GPL, LGPL, Apache License, ...

# Projektdurchführung

## Sprint 1

### Sprintplanung

Dauer: 29.01.2025 – 18.02.2025

Ausgewählte User Stories:

* Mockup:

Erstellung und Planung des Designs sowie Umsetzungsplanung der Anforderungen.

* 14 Story Points
* Aufstellung der Verbindung zum Backend über API-Zugriff:

Das Frontend sollte über diese API mit dem Backend kommunizieren. Dabei muss das Frontend Daten zur Visualisierung aus der API holen können und aber auch Befehle hinschicken, wie der MBOT fahren sollte.

* 10 Story Points
* Erstellung der Connection Funktionalität und UI:  
  Der User sollte sich mit einem MBot anhand der IP-Adresse verbinden und ihm hier einen Namen geben können. Außerdem sollte der USER eine vorherige Sitzung wiederherstellen können.
* 20 Story Points
* Steuerung UI:   
  Der User soll mit dem System über eine GUI interagieren können. Dabei muss das Fenster für manuelle Steuerung und automatische Steuerung erstellt werden.
* 27 Story Points

Anzahl Story Points: 71

Ausgewählte Punkte aus der Impediment Liste:

* Missverständnisse und Streitigkeiten beim Design – Farben verbessern
* Kommunikation
* Timo Winkler muss mehr tun.

### Sprint Demo

In diesem Sprint wurden alle User Stories wie geplant umgesetzt. Außerdem haben wir alles gut umgesetzt. Alle Fehler wurden bereits entfernt korrigiert.

### Sprint Retrospektive

Besonders gut verlief die Kommunikation mit dem Backend, mit dem wir ständig im Austausch waren. Wir sind außerdem mit dem Aufbau der Connection und Control Pages sehr zufrieden.

Verbesserungsbedarf gibt es bei der Aufgabenverteilung und bessere Kommunikation beim Design.

Auflistung der Impediment Taskliste:

* Missverständnisse und Streitigkeiten beim Design – Farben verbessern
* Timo Winkler muss mehr tun.

### Sprint Zusammenfassung

1. Zuerst haben wir uns um das Design gekümmert.
2. Wir haben die Login Page mit Funktionalität eingebaut
3. Wir haben die API-Schnittstelle erstellt.
4. Aufbau und Design der Steuerung Pages

**Während des 1. Sprints hat es keine weiteren Änderungen am Product Backlog gegeben.**

**Burndownchart:**

**Ein Bild, das Text, Reihe, Diagramm, Zahl enthält.

KI-generierte Inhalte können fehlerhaft sein.**

**Sprint Velocity**

In diesem Sprint hatten wir eine Velocity von 71. Mit einer solchen Geschwindigkeit wäre man in 2-3 Sprints fertig, also ca. 1 Sprint vor dem geplanten Ende. Da es noch keine Vorherigen Sprints lassen sich aber noch keine konkreten Vorhersagungen tätigen.

Ein Bild, das Text, Screenshot, Diagramm, Zahl enthält.

KI-generierte Inhalte können fehlerhaft sein.

## Sprint 2

### Sprintplanung

Dauer: 26.02.2025 – 18.03.2025

Ausgewählte User Stories:

* Mockup Handy:

Erstellung und Planung des Designs sowie Umsetzungsplanung der Anforderungen für die mobile Ansicht.

* 15 Story Points
* Implementierung der manuellen Steuerung:

Das Frontend sollte über die API mit dem Backend kommunizieren und dabei Befehle zum Fahren, aber auch für Farbe und Geschwindigkeit mitgeben können. Dafür muss das Frontend die Daten richtig verarbeiten und an das Backend schicken.

* 20 Story Points
* Abgefahrene Route aufzeichnen und schicken:  
  Der User kann bei der manuellen Steuerung automatisch seine Route aufzeichnen lassen. Am Ende wird er gefragt, ob er diese Route speichern will. Diese wird dann an das Backend zum Speichern in der Datenbank geschickt. ob er eine hat sollte sich mit einem MBot anhand der IP-Adresse verbinden und ihm hier einen Namen geben können. Außerdem sollte der USER eine vorherige Sitzung wiederherstellen können.
* 35 Story Points
* Datenvisualisierung:   
  Kommunikation mit dem Backend um Detaildaten wie die Batterieladung zu holen. Geschwindigkeit etc. werden berechnet und visualisiert.
* 15 Story Points

Anzahl Story Points: 85

Ausgewählte Punkte aus der Impediment Liste:

* Verzögerung bei Datenvisualisierung
* Kommunikation mit Backend über API 🡪 Probleme bei Steuerung (Verzögerung)
* Timo Winkler muss mehr tun.

### Sprint Demo

In diesem Sprint gab es leider eine Verzögerung bei der Steuerung, da immer wieder mehrmals dieselben Befehle an das Backend geschickt wurden. Das kostete dem Team viel Zeit und Energie. Schlussendlich konnten aber diese Probleme behoben werden. Die Speicherung der abgefahrenen Route wurde ohne Probleme umgesetzt. Durch die Verzögerung und mangelnde Daten vom Backend für die Visualisierung, konnte die Datenvisualisierung leider nicht bis zum Ende des Sprints fertiggestellt werden und wurde wieder im Sprint Backlog aufgenommen. Insgesamt konnten wir unsere Velocity von ~ 70, trotz Verzögerung halten.

### Sprint Retrospektive

Besonders gut verlief die Kommunikation mit dem Backend, mit dem wir ständig im Austausch waren. In diesem Sprint haben wir das manuelle Fahren vollständig umgesetzt und begonnen eine Route zu definieren, abzufahren und zu speichern. Des Weiteren haben wir im Team gute Zusammenarbeit genossen, so haben wir oft gemeinsam Probleme und Fehler gelöst. Am Ende haben wir eine ähnliche Velocity wie im Sprint davor, also 70 erreicht. Leider haben wir die Datenvisualisierung nicht vollständig umsetzen, da einerseits Daten vom Backend fehlten und allgemein die Visualisierung nicht abgeschlossen wurde. Möglicherweise wurde sich bei der Planung auch überschätzt und es wurde deshalb nicht abgeschlossen.

Verbesserungsbedarf gibt es bei der Aufgabenverteilung und bei dem Vorausplanen.

Auflistung der Impediment Taskliste:

* Missverständnisse und Streitigkeiten – Endpoints, Doppelte Nachrichten etc.
* Timo Winkler muss mehr tun.

### Sprint Zusammenfassung

## Sprint n

# Installation / Software deployment

Anleitung welche Schritte notwendig sind um das fertige SW Produkt zu installieren und in Betrieb zu nehmen.

# Projektabschluss

## Projektzusammenfassung

Zusammenfassung der Projektdurchführung. Was lief gut/schlecht. Welche Erkenntnisse wurden während der Durchführung des Projektes gewonnen. Was würde man, nun anders machen bzw. wieder gleich machen?

## Attachments

Tabellarische Auflistung der Projektdateien.

z.B.: ZIP-File mit dem Quellcode, Projektpräsentationen, ...