

Planung der Implementation (V1.7)



Dokumenteninformation

Erstellt von: Leon Daskov

Bearbeitet von: Leon Daskov

Dateiname: Implementierungsplanung_V1.7.pdf

Versionierung

| Version | Datum | Bearbeiter | Änderung | Stand |
|---------|------------|-------------|--|--------------|
| 1.0 | 22.04.2021 | Leon Daskov | Initiale Erstellung | Überarbeitet |
| 1.1 | 25.04.2021 | Leon Daskov | Änderung unter der Erstellung der Datenbanken #12 | Überarbeitet |
| 1.2 | 25.04.2021 | Leon Daskov | Hinzufügen von groben Zielen für Sprint 2-4 | Überarbeitet |
| 1.3 | 26.04.2021 | Leon Daskov | Hinzufügen von Testzielen | Abgenommen |
| 1.4 | 02.05.2021 | Leon Daskov | Hinzufügen von Stories ab Sprint 2 und Update der Sprints auf Basis des Fortschritts | Abgenommen |
| 1.5 | 09.05.2021 | Leon Daskov | Ausarbeitung des Sprint 3 | Abgenommen |
| 1.6 | 16.05.2021 | Leon Daskov | Ausarbeitung des Sprint 4 | Abgenommen |
| 1.7 | 19.05.2021 | Leon Daskov | Verlängerung des Sprint 4 | Abgenommen |

Inhalt

| | |
|--|---|
| 1. Das System zur Implementierung | 1 |
| 1.1 Sprintplanung | 1 |
| 1.2 Implementierung und Versionskontrolle | 1 |
| 2. Die 4 Sprints | 2 |
| 2.1 Sprint 1 [19.04.2021 – 26.04.2021] – UI, Authentifizierung und Mailing-Service | 2 |
| 2.2 Sprint 2 [26.04.2021-02.05.2021] | 4 |
| 2.3 Sprint 3 [03.05.2021-09.05.2021] | 5 |
| 2.4 Sprint 4 [10.05.2021-16.05.2021] | 6 |

1. Das System zur Implementierung

1.1 Sprintplanung

Zum Anfang jeder Woche, trifft sich die Gruppe, um ein Sprintplanning durchzuführen. In diesem Planning werden aus groben Implementationsschritten, Issues, oder auch Stories, definiert.

Diese Stories sollen in dem Zeitabschnitt einer Woche erstellt werden können und als Feature testbar sein. Unittests werden direkt in Kombination mit der Implementierung der Storys erstellt.

Die erstellten Storys werden in unserem GitHub-Project-Board gehalten. In unserem Projektmanagement innerhalb der GitHub-Seite, werden den verschiedenen Stories Prioritäten und IDs zugewiesen. Die IDs werden in diesem Dokument mit der „#Zahl“ Form angegeben.

Die Implementierung passiert in Paaren, um dem Ansatz des Pair-Programming gerecht zu werden. Jede Woche werden neue Programmierpaare zugewiesen, so dass wenn möglich, immer eine Person aus dem Paar, Wissen in den Themenbereich der zu bearbeitenden Stories besitzt.

1.2 Implementierung und Versionskontrolle

Die Implementierung geschieht nach den Codingrichtlinien von Kotlin. Abweichungen von diesen sollen durch Refactorings behoben werden.

Die genutzte Versionskontrolle ist Git, mithilfe der Plattform Github.

Zu jedem Issue oder Pairprogramming-Paar, soll ein Branch erstellt werden, damit unser Master-Branch möglich funktionsfähig bleibt.

Auf diesem Branch wird die Implementierung der Issues von dem jeweiligen Paar geschehen und nach eigenem Ermessen des Fortschritts, iterativ gepusht.

Wenn ein Paar fertig mit ihren Stories, einer Story, welches andere Gruppen benötigen oder eines komplexen und großen Story, soll ein erster Pull-Request erstellt werden. Hierbei wird nach Kontext entweder direkt in den Master-Branch requested oder auf einen Branch einer Gruppe, die den aktuellen Fortschritt der Gruppe benötigt.

Bei einer Pull-Request werden, nach unserer Testdokumentation: Testkonzept_v1.1.pdf, alle Tests ausgeführt und das Ergebnis der Tests wird angezeigt.

Um jedoch einen Pull-Request zu akzeptieren, muss ein weiterer dritter Programmierer, welcher nicht an der Story beteiligt war, den Code reviewen und die Pull-Request akzeptieren.

Insbesondere bei Pulls in den Master-Branch, sollte der Code mit hoher Sorgfalt reviewed werden.



2. Die 4 Sprints

2.1 Sprint 1 [19.04.2021 – 26.04.2021] – UI, Authentifizierung und Mailing-Service

Initialplanung:

Dieser Eröffnungssprint fokussiert sich auf die Erstellung der ersten UI-Komponenten, der User-Authentifizierung und des Mailing-Services.

Testzielsetzung:

Die Tests sollen laut unserer Testkonzept-Dokument umgesetzt werden und spätestens im nächsten Sprint hinzugefügt werden.

In diesem Sprint werden nur Unit Tests implementiert.

Authentifizierung #9 – Leon Daskov & Marius Armbruster

Diese Story umfasst die Backend-Implementation des Authentifizierungssystems.

Die Arbeitspunkte sind aufgeteilt in:

- Abfrage und Ablegen der User-Daten in der Datenbank
- Erstellung des Registrierungsprozesses
- Erstellung des Loginprozesses
- Verschlüsselung des Passworts, um dieses auf der Datenbank sicher zu speichern
- Optional: Erstellung eines Logins basierend auf einem Session-Cookie

Dieses Arbeitspaket wird Zusammenarbeit mit dem Datenbankpaar aus Story #12 erfordern, da datenbankseitige Arbeit notwendig ist.

Mailing-Service #11 – Leon Daskov & Marius Armbruster

Diese Story umfasst die Backend-Implementation des Mailing-Services.

Dem User sollen durch dieses Feature E-Mails gesendet werden, wenn dieser:

- Sich registriert, damit er sich verifizieren kann
- Sein Passwort vergessen hat und auf die „Reset Password“-Funktion zugreift.

Dieses Arbeitspaket erfordert Code aus der Story #9, weshalb bei beiden Stories das Programmierpaar Marius Armbruster und Leon Daskov zugeteilt worden sind.



Erstellung der Spiele- und Konfigurationsdatenbanken #12 – Jakub Cielecki und Johannes Hüttinger

Hier wird das Datenmanagement der Spiele und Konfigurationen behandelt.

Die Datenbanken für die Spieleerstellung, der Speicherdaten der verschiedenen Spieler und die Konfigurationsdateien sollen in Datenbanken gespeichert werden

Hinzugefügt am 25.04.2021:

Die Erstellung der Konfigurationsdatenbank wird in einem späteren Sprint erfolgen, da es noch nicht endgültig geklärt worden ist, wie die Konfigurationen gespeichert werden sollen.

Hinzugefügt am 26.04.2021:

Die Erstellung der Konfigurationsdatenbank wird in Sprint 2 erfolgen, da es endgültig geklärt worden ist, dass die Konfigurationen in einer relationalen Datenbank gespeichert wird.

Erstellung der Angular UI #7 – Linus Kadner & Daniel Scherer

In dieser Story wird die Erstellung der Angular UI in Form einer Nachbildung der UI-Mocks angestrebt.

Folgende UI Seiten werden in diesem Sprint mit hoher Priorität bearbeitet:

- Die Spiele-UI, um das MUD spielen zu können
 - o Hierbei wird MUD-Master- und normale Spieler-UI getrennt erstellt
- Die Basis-UI für den Konfigurator
- Die Register und Login UI Komponente
- Das Dashboard, über welches man, die verschiedenen Komponenten erreichen kann

Bereitstellung von Dummydaten für die UI #8 – Jakub Cielecki und Johannes Hüttinger

Es sollen Dummydaten bereitgestellt werden, damit die erstellten UI Komponenten abgenommen werden können und bereits einen Eindruck für das spätere Produkt bieten können.

Aufstellen der Kommunikation zwischen Authentifizierungslogik und UI #10 – Linus Kadner & Daniel Scherer

Die alleinige Erstellung von UI und Backendlogik der Authentifizierung führen zu keiner funktionierenden Software. Damit diese Software funktioniert, werden in dieser Story die Ergebnisse aus #11 und #7 zusammengeführt damit der Registrierungs- und Login-Prozess funktionieren.



2.2 Sprint 2 [26.04.2021-02.05.2021]

Grobe Zielsetzung:

In diesem Sprint sollen die ersten groben Gameplay und Konfigurator-Features implementiert werden, diese beinhalten:

- Erstellung einer simplen und abgespeckten MUD-Konfiguration
- Erstellung eines MUD-Spiels auf Basis einer Konfiguration
- Die Erstellung von einem Avatar in Koalition mit den Speicherständen
- Die ersten Moderierungsmöglichkeiten eines MUD-Masters in einem MUD
 - o Spawnen von Nicht-Spieler-Charakteren
 - o Teleportieren von Spielern in andere Räume
- Das Bespielen eines MUDs in seinen fundamentalen Features (Laufen, interagieren, umsehen), somit soll eine ähnliche Feature Varianz, wie in dem Prototypen gegeben werden

Bearbeitung vom 02.05.2021: Die Planung des Sprints wurde am 26.04.2021 überarbeitet. Die initiale Grobe Zielsetzung der Sprints wurde verworfen, da auf die Implementierungsstrategie der User-Stories gewechselt wurde.

Einführung der User-Stories

In jedem folgenden Sprint sollen die User-Stories definiert werden. Diese sollen Nutzungs-Abläufe darstellen, die implementiert werden.

User-Story – Authentifizierung:

Ich möchte als User in der Lage sein mich zu registrieren und anzumelden. Hierbei sollen Verifikations-E-Mails an mich gesendet werden und ich soll in der Lage sein mein Passwort zu ändern.

Verantwortliche:

Daniel Scherer, Marius Armbruster, Jakub Cielecki

Issues im Projekt: #22, #10

In diesem Sprint soll das Authentifizierungssystem komplettiert werden. Hierbei soll es dem User möglich sein sich zu registrieren, anzumelden und sein Passwort zu ändern oder zu resettet. Bei der Registrierung wird dem Nutzer eine Verifikationsmails gesendet damit sich dieser verifizieren kann. Bei der Anmeldung wird dem User eine Session gestellt, über welche er mit „seinen“ Inhalten später interagieren kann.

Weiteres Arbeitspaket – Command-Parsing

Dies ist ein Arbeitspaket, welches unabhängig von der User-Story des Sprints bearbeitet wird.

Abhängig von einem Inputstring, sollen auf Basis eines „Command-Aliases“ und Parametern ein Command erstellt werden der später einen Einfluss auf das Spiel nimmt.

Verantwortliche:



Leon Daskov, Linus Kadner

Hier soll der Commandparser implementiert werden. Dieser akzeptiert UserInputs in Form von Strings und bearbeitet diese. Hier werden Commands generiert die, dann mit dem vom User übergebenen Parametern, ausgeführt werden können.

Weiteres Arbeitspaket – Game-Runner

Dies ist ein Arbeitspaket, welches unabhängig von der User-Story des Sprints bearbeitet wird.

Verantwortliche: Johannes Hüttinger, Jakub Cielecki

In diesem Arbeitspaket wird der Game-Runner implementiert. Dieser handelt die Warteschlange für das Ausführen der Commands und kann den aktuellen Zustand es Spieles zurückgeben.

Ebenfalls können die aktuellen Spieler, welche Online sind, übergeben.

Connections zu einem Spiel werden ebenfalls durch den Game-Runner gehandelt.

Weiteres Arbeitspaket – Datenbanken

Dies ist ein Arbeitspaket, welches unabhängig von der User-Story des Sprints bearbeitet wird.

Verantwortliche: Linus Kadner, Leon Daskov

In diesem Arbeitspaket werden die Datenbank-Zugriffe erstellt, welche später zum Persistieren der Gamestates benötigt werden.

2.3 Sprint 3 [03.05.2021-09.05.2021]

Grobe Zielsetzung:

In diesem Sprint sollen die Gameplay-Features ausgeweitet werden und dem Spiel einen ersten „Spielwert“ verleihen. Auch wird in diesem Sprint dem User und dem MUD-Master endlich möglich sein miteinander zu chatten. Die einzubauenden Features sind unten gelistet:

- Erweiterung des Spieles um Features wie ein sehr simples Kampfsystem
 - o Aktionen wie greife an und nehme Schaden sind möglich
- Erweiterung des Spielers um Stats
 - o Erweiterung des Spielers um Ausrüstung
- Einfügen von Aliases für bestehende „Kommandos/Aktionen“ der Spieler und MUD-Master
- Erweiterung der MUD-Master-Basis-Kommandos
- Erweiterung der Möglichkeiten mit Nicht-Spieler-Charakteren und Items zu interagieren
- Zuweisen von Effekten auf Gegenstände soll möglich sein

Änderung vom 09.05.2021: Die initiale grobe Zielsetzung wurde mit der Einführung der User-Stories verworfen.

User Story – Erstellung der Konfiguration

Ich möchte als User in der Lage sein, nachdem ich mich eingeloggt habe, eine Konfiguration zu erstellen. Hierzu geh ich in den Konfigurator und erstelle meine Konfiguration. Wenn ich fertig bin mit der Erstellung der Konfiguration, soll diese auf die Datenbank gelegt werden.



Ich möchte nun die Konfiguration im Dashboard sehen, damit ich später mit dieser spielen kann.

Verantwortliche: gesamtes Team

Issues im Projekt: #33, #19, #34

In diesem Sprint soll der komplette Ablauf der Erstellung einer Konfiguration fertiggestellt werden. Diese Konfigurationen sollen dann auf dem Dashboard angezeigt werden, damit diese dann später mit der Implementierung des eigentlichen Spiels, bespielt werden können.

Weiteres Arbeitspaket – Commands und Actions

Dies ist ein Arbeitspaket, welches unabhängig von der User-Story des Sprints bearbeitet wird.

Verantwortliche: Leon Daskov, Johannes Hüttinger

Dieses Arbeitspaket wird im 3. und 4. Sprint bearbeitet, da dies die Vorbereitung auf die User-Story des eigentlichen Spieles ist.

In diesem Arbeitspaket werden die Actions und ToAction Methoden implementiert, damit die Commands, die geparsed werden, später in die Queue genommen werden, um sie auszuführen.

Weiteres Arbeitspaket – UI für die Password-Änderung und UI für die Masteransicht auf das Spiel

Dies ist ein Arbeitspaket, welches unabhängig von der User-Story des Sprints bearbeitet wird.

Verantwortliche: Leon Daskov, Daniel Scherer

In diesem Arbeitspaket soll die UI erstellt werden, die dem User erlaubt sein Passwort zu ändern. Ebenso soll in diesem Arbeitspaket auf den letzten Sprint und die letzte User Story vorbereitet werden. Dies geschieht durch die Erstellung der Master UI.

2.4 Sprint 4 [10.05.2021-16.05.2021]

Grobe Zielsetzung:

In diesem Sprint soll die Spielqualität gesichert werden und Features, welche nicht unbedingt notwendig sind, um ein simples Spielerlebnis zu bieten, implementiert.

- Erstellung von konfigurierbaren Kommandos (Es werden die vordefinierten Kommandos aneinandergereiht, um komplexere Kommandos möglich zu machen)
 - o Diese werden im Konfigurator konfiguriert und werden dann im Spiel nutzbar sein
- Finale Systemtests, um zu sichern, dass das System als Ganzes funktioniert
- Implementierung von Luxusfeatures

Änderung vom 09.05.2021: Die initiale grobe Zielsetzung wurde mit der Einführung der User-Stories verworfen.



User Story – Spielen des eigentlichen Spiels

Ich möchte als User in der Lage sein, nachdem ich mich eingeloggt habe, an einem Spiel teilzunehmen. Ich möchte in der Lage sein, Commands einzugeben, was dazu führt, dass Änderungen im Spiel vorgenommen werden. Auch möchte ich in der Lage sein, in diesem Spiel mit anderen Spielern zu chatten.

Wenn ich selber einen MUD konfiguriert habe, möchte ich in der Lage sein, als Master diesem Spiel beizutreten und das Spielgeschehen zu beeinflussen.

Egal ob ich normaler Spieler oder MUD-Master bin, möchte ich, dass ich das Spiel zu einem späteren Zeitpunkt weiterspielen kann.

Verantwortliche: gesamtes Team

Issues im Projekt: #37, #47, #36, #45, #19, #20, #21, #16

In diesem Sprint soll es dem normalen Spieler, als auch dem Master, möglich sein an einem Spiel auf Basis einer MUD-Konfiguration teilzunehmen. In diesem Spiel können über Commands, Aktionen ausgeführt werden, die den Zustand des Spiels verändern. Auch soll bereits die Chatfunktion implementiert sein, dass es möglich ist, mit anderen Spielern oder dem Master zu kommunizieren.

Weiteres Arbeitspaket – Erstellung der End-To-End-Tests für die Konfigurationen

Dies ist ein Arbeitspaket, welches unabhängig von der User-Story des Sprints bearbeitet wird.

Verantwortliche: Marius Armbruster

In diesem Arbeitspaket sollen die ersten End-To-End-Tests erstellt werden, welche den Ablauf der Erstellung der Konfiguration abdecken.

Weiteres Arbeitspaket – Erstellung eines FAQ

Dies ist ein Arbeitspaket, welches unabhängig von der User-Story des Sprints bearbeitet wird.

Verantwortliche: Jakub Cielecki, Linus Kadner

In diesem Arbeitspaket soll ein FAQ erstellt werden, welches offene Fragen der Spieler beantworten soll, um diesen das Spielen zu vereinfachen.



2.5 Sprint 4.1 [17.05.2021-16.05.2021] – Verlängerung des Sprint 4

User Story – Spielbarkeit des eigentlichen Spieles verbessern

Ich möchte als User das Spiel nun in einer verbesserten Version spielen. Bestimmte Aktionen sind nun über graphische UI-Elemente ansprechbar und sind dadurch leichter zu verwenden. Viele Commands wurden dem Spieler nun freigeschaltet und sind nun benutzbar.

Das Spiel ist zu vollem Umfang spielbar und nun, um einiges einfacher zu benutzen.

Diese Veränderung zur Verbesserung des Spielerlebnis, wird sowohl auf Master und Spieler-Perspektive vollbracht.

Verantwortliche: gesamtes Team

Issues im Projekt: #60, #58, #51, #57, #45

Diese Story ist eine Erweiterung der Story des Sprints 4 und soll zu einer besseren Spielbarkeit des eigentlichen Spieles führen.

Weiteres Arbeitspaket – Fixen von Bugs

Dies ist ein Arbeitspaket, welches unabhängig von der User-Story des Sprints bearbeitet wird.

Verantwortliche: gesamtes Team

In diesem Arbeitspaket wurden signifikante Bugs gefixed.

Beispiel: Der Master verliert nun nicht mehr die Verbindung, wenn er einen nicht implementierten Command ausführt.

Weiteres Arbeitspaket – Auskommentierung des Codes

Dies ist ein Arbeitspaket, welches unabhängig von der User-Story des Sprints bearbeitet wird.

Verantwortliche: Jakub Cielecki, Leon Daskov

Im Source-Code des Projektes war die Dokumentation teilweise nur sehr begrenzt vorhanden. Dies wurde in diesem Arbeitspaket behoben und die gesamte Logik des Servers, wurde mit KDoc-Kommentaren auskommentiert, wodurch eine Dokka-Dokumentation erstellt werden kann.

