Lernatelier: Projektdokumentation

Herrmann/Hartmann/Dal Corso

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Version** | **Änderung** | **Autor** |
|  | 0.0.1 |  | [Dal Corso] |
| … | | | |
|  | 1.0.0 | Finale Version |  |

# Informieren

## Ihr Projekt

Wir haben uns entschieden ein RPG erstellen mit Windows Form.

## Quellen

[SOSCSRPG – Scott's Open-Source C# Role-Playing Game](https://soscsrpg.com/)

## Anforderungen

|  |  |  |  |
| --- | --- | --- | --- |
| **Nummer** | **Muss / Kann?** | **Funktional? Qualität? Rand?** | **Beschreibung** |
| 1 | Muss | **Funktional** | Player kann sich bewegen |
| 2 | Muss | **Funktional** | Monster erscheinen |
| 4 | Muss | **Funktional** | Monster greifen Player an |
| 5 | Muss | **Funktional** | Player greift Monster an |
| 6 | Muss | **Qualität** | Schaden Berechnung |
| 7 | Muss | **Qualität** | Quest annehmen |
| 8 | Muss | **Qualität** | Quest Benden |
| 9 | Muss | **Funktional** | Kann player erstellen mit einer Klasse |
| 10 | Muss | **Funktional** | Stirbt player gameover |
| 14 | Kann | **Qualität** | Von Monster exp, Gold und item bekommen |
| 15 | Kann | **Funktional** | Player Items verkaufen und kaufen |
| 16 | Kann | **Qualität** | Level aufsteigen |
| 17 | Muss | **Rand** | Mit Cshapre Programmiren |

## 1.4 Diagramme

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

Ein Bild, das Diagramm enthält.

Automatisch generierte Beschreibung

## 1.5 Testfälle

[Erstellen Sie zu jeder Muss-Anforderung mindestens einen Testfall.]

|  |  |  |  |
| --- | --- | --- | --- |
| **Nummer** | **Voraussetzung** | **Eingabe** | **Erwartete Ausgabe** |
| 1.1 | Programm lauf und Testfall 9 erledigt | Pfeil Taste nach unten | player geht nach unten im Spiele |
| 1.2 | Programm lauf und Testfall 9 erledigt | Pfeil Taste nach links | player geht nach links im Spiele |
| 1.3 | Programm lauf und Testfall 9 erledigt | Pfeil Taste nach rechts | player geht nach rechts im Spiele |
| 1.4 | Programm lauf und Testfall 9 erledigt | Pfeil Taste nach oben | player geht nach oben im Spiele |
| 2 | Programm lauf und player ist in einem dungeons | In dungeons gehen | Monster erscheint |
| 4 | Testfall drei | Player in Monster range | Monster setzt Attacke ein |
| 5 | Testfall drei Monster in Player range | Drückt attack button | Player greift Monster an. |
| 9 | Programm lauf Klassen Bildschirm erscheint | Drück auf Bogenschütze Klasse | Spiele wird gestartet |
| 10 | Programm lauf | HP points 0 | Game over Screen |
| 15.1 | Programm lauf und player hat items | Verkauft Item für 20 Gold  Gold 0 | 20 Gold bekommen |
| 15.2 | Programm lauf und player hat items | Kauft Schwert für 20 Gold  Gold 20 | 0 Gold  Schwert bekommen |

1. Planen

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nummer** | **Frist** | **Beschreibung** | **Zeit (geplant)** | **Wer** |
| 1 | 5.4 | Player kann sich bewegen | 2X45 | Dal Corso |
| 2 | 5.4 | Monster erscheinen | 2X45 |  |
| 4 | 3Woche | Monster greifen Player an | 2X45 |  |
| 5 | 3Woche | Player greift Monster an | 1X45 |  |
| 6 | 4Woche | Schaden Berechnung | 1X45 |  |
| 7 | 4Woche | Quest annehmen | 2X45 |  |
| 8 | 4Woche | Quest Benden | 2X45 |  |
| 9 | 5.4 | Kann player erstellen mit einer Klasse | 3X45 | Hartmann/Herrmann |
| 10 | 5Woche | Stirbt player gameover | 3X45 |  |
| 14 | 5Woche | Von Monster exp, Gold und item bekommen | 3X45 |  |
| 15 | 5Woche | Player Items verkaufen und kaufen | 3X45 |  |
| 16 | 5Woche | Level aufsteigen | 3X45 |  |

\* Die Nummer hat das Format N.m, wobei N die Nummer der Anforderung ist, zu der das Arbeitspaket gehört, und m von 1 an fortlaufend durchnummeriert wird.

\*\* Teilen Sie diesmal Ihre Anforderungen in 45-Minuten-Arbeitspakete ein

1. Entscheiden

GitHub für Zusammenarbeit <https://github.com/>

Mit Tutorial arbeiten ([SOSCSRPG – Scott's Open-Source C# Role-Playing Game](https://soscsrpg.com/))

Arbeitspakete auf Personen aufteilen

Luca ist Haupt-Dokuleiter

1. Realisieren

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nummer** | **Frist** | **Beschreibung** | **Zeit (geplant)** | **Zeit (effektiv)** |
| 1 | 5.4 | Player kann sich bewegen | 2X45 | 1X45 |
| 2 | 5.4 | Monster erscheinen | 2X45 |  |
| 4 | 3Woche | Monster greifen Player an | 2X45 |  |
| 5 | 3Woche | Player greift Monster an | 1X45 |  |
| 6 | 4Woche | Schaden Berechnung | 1X45 |  |
| 7 | 4Woche | Quest annehmen | 2X45 |  |
| 8 | 4Woche | Quest Benden | 2X45 |  |
| 9 | 5.4 | Kann player erstellen mit einer Klasse | 3X45 | 3X45 |
| 10 | 5Woche | Stirbt player gameover | 3X45 |  |
| 14 | 5Woche | Von Monster exp, Gold und item bekommen | 3X45 |  |
| 15 | 5Woche | Player Items verkaufen und kaufen | 3X45 |  |
| 16 | 5Woche | Level aufsteigen | 3X45 |  |

[Übernehmen Sie Ihre Planung aus 2., und tragen Sie nach, wie lang Sie effektiv zur Bearbeitung der jeweiligen Arbeitspakete benötigt haben.]

1. Kontrollieren

## Testprotokoll

|  |  |  |  |
| --- | --- | --- | --- |
| **Nummer** | **Datum** | **Resultat** | **Durchgeführt** |
|  |  |  | [Nachname] |

[Vergessen Sie das Fazit aus dem Testprotokoll nicht!]

1. Auswerten

[Listen Sie hier je mindestens einen Punkt, der gut gelaufen ist, und einen Punkt, der schlecht gelaufen ist – mit diesen starten Sie dann in Ihren Portfolio-Eintrag.]