Dokumentation

Android Studio-Projekt: SRunnerCreator



Erstellt von

*Ole Teichmann, Sven Lütje, Daniel Ojea & Daniel Schönberger*

**Inhaltsverzeichnis**

[1 Projektbeschreibung 3](#_Toc481493925)

[2 ProjektManagement-Dokumente 4](#_Toc481493926)

[2.1 Zielformulierung 4](#_Toc481493927)

[2.2 UML-Klassendiagramme 4](#_Toc481493928)

[2.3 Arbeitspakete und Zuständigkeiten 5](#_Toc481493929)

[2.4 Meilensteine 6](#_Toc481493930)

[3 Reflexion 6](#_Toc481493931)

3[.1 Problemstellung 6](#_Toc481493927)

[3.2 Projektmanagement 7](#_Toc481493928)

[3.3 Arbeit im Team 7](#_Toc481493929)

# **1. Projektbeschreibung**

Die SRAPP ist eine App zur Charaktererstellung eines Pen and Paper Charakters. Hierbei wird der Charakter nach dem Regelwerk von Shadowrun der 5. Edition erstellt. Die App erleichtert das Erstellen eines Charakters und führt/leitet den Anwender durch die unterschiedlichen Punkte, bis hin zur Fertigstelllung. Zusätzlich speichert die App erstellt Charaktere, sodass diese gesichert und abrufbar sind.

Die App startet mit einem Startscreen. Wenn der Anwender auf den Startscreen „klickt“, so gelangt dieser zu der Hauptmenüführung. In diesem Hauptmenü werden alle bisher erstellen Charaktere aufgeführt. Die Charakterauswahl gibt die Möglichkeit, bisher gespielte/gespeicherte/erstellte Charaktere aufzurufen. Die Charakterbögen (Charactersheets) sind als PDF und XML exportier- und importierbar. Klickt der Anwender im Hauptmenü auf den grünen Button unten rechts, so kommt dieser in die Charaktererstellung.

Die Charaktererstellung spaltet sich auf die einzelnen Bereiche (Metatyp, Attribute, Fähigkeiten, etc.) auf, wo sich der Anwender Schritt für Schritt durchklickt mithilfe eines „WEITER“ Buttons.

Hierbei sind die Werte, die verteilt werden können, nach dem Regelwerk und den Entscheidungen des Anwenders bemessen. Die Punkte werden verteilt, indem man neben der Fähigkeit/ dem Attribut ein Zahlenfeld hat, welches mit einem „+“ und „-“ Buttons den Wert entweder um eins erhöht oder vermindert. Um zu sehen wie viele Punkte noch offen sind, wird dem Anwender stets ein Punktezähler angezeigt, welcher die unverteilten Punkte anzeigt.

Sind alle Werte eingetragen, alle Punkte verteilt und Fähigkeiten ausgewählt, so kommt man zu einer Oberfläche, wo man seinem Charakter einen Namen und eine Geschichte geben kann als letzten feinschliff. Hinzu kommt, dass der Anwender dem Charakter ein Profilbild hinzufügen kann, indem er eine Bilddatei hochlädt.   
  
Ist der Charakter fertiggestellt, so wird der dieser lokal gespeichert und ist von nun an im Hauptmenü aufrufbar.

# **2. Projektmanagement-Dokumente**

## 2.1 Zielformulierung

Die Charaktererstellung von Shadowrun ist mit dem Regelwerk durchzuführen. Das Regelwerk leitet den Spieler bei der Erstellung seines Charakters und erläutert hierbei die Möglichkeiten, wie man seinen Charakter anpassen kann. Neben dem Grundregelwerk gibt es eine Art Einsteigerregelwerk, welches mit Schnellstartregeln den Spieler nur die nötigsten Regeln vermittelt um einen Charakter regelkonform zu gestalten.

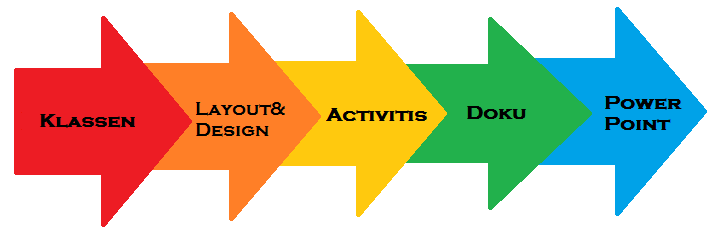
Die Schnellstartregeln haben uns den Anstoß gegeben eine App zu bauen, welches auf das Grundregelwerk aufbaut und den Spieler leicht durch die Charaktererstellung leitet, ähnlich wie die Schnellstartregeln.  
  
Als Zusatz ist geplant, die erstellten Charaktere im Hauptmenü aufzuführen und abrufbar zu machen.

## 2.2 UML-Klassendiagramme

|  |  |
| --- | --- |
| Gruppenmitglied | Zuständigkeit |
| Ole Teichmann | **Fertigkeiten:**  - Spezialisierung  - Werte Steigern  - Spezialisierungen hinzufügen „Funktion“  - Spezialisierung entfernen „Funktion“  - Paketfertigkeiten  - Wissensfertigkeiten  **Attribute:**  - Werte steigern  - Werte mindern  **Klassenstruktur**  **Scrum-Master**  **Connection:**  - Connection schreiben/hinzufügen  - Werte steigern  - Werte mindern  - Connection entfernen |
| Sven Lütje | **Grafisches Design:**  - Design Layout  **Styles**  **Stings vereinheitlicht** *( Maßnahmen gegen Hardcode)*  **UML-Klassendiagramm** |
| Daniel Ojea | **Prioritäten:**  -Einteilung  -Reihenfolgeschaltung „Funktion“  **Character-sheet**  **Vor- und Nachteile**  **Hauptmenü:**  -XML-file auf speichern und wiederverwenden  **Klassenstruktur** |
| Daniel Schönberger | **Klassenstruktur Metatypauswahl MetatypDetails Grafisches Design Vor- und Nachteile Datenpflege Dokumentation Powerpoint Profilbild einbinden vom internen Speicher „Funktion“** |

## 2.3 Arbeitspakete und Zuständigkeiten

## 2.4 Meilensteine



# **3. Reflexion**

## 3.1 Problemstellung

Das Regelwerk, welches sehr umfangreich ist, ließ uns durch Zeitliche Begrenzungen Kürzungen im Bereich Magiefertigkeiten und Auswirkungen vornehmen. Grund dafür waren viele Eigenklassen, keine verwendeten Standart-Views und aufwendiges Costumize.  
  
Was haben wir durch die Projektarbeit gelernt?  
- Grundkenntnisse im Umgang mit Android Studio   
- Grundkenntnisse in der Programmiersprache Java  
- Verwaltung eines Projekts über GitHub  
- Strukturierte Teamorganisation  
- Gute Arbeitsaufteilung  
- Projektablaufplanung

## 3.2 Projektmanagement

Unsere Projektplanung und dessen Durchführung bewerten wir als Gruppe mit einem sehr gut. Die Aufgabenverteilung wurde stets gleichermaßen belastbar aufgeteilt und auf Wünsche wurde eingegangen. Kenntnisse der Gruppe wurden effektiv eingesetzt. Trotz der Arbeitsaufteilung wurde im Team untereinander Unterstützt.  
  
Was haben wir aus der Gruppenarbeit gelernt?  
  
Durch unsere Art die Arbeit einzuteilen und unserer guter Zusammenarbeit, konnten wir unsere geplanten Zielsetzung zufriedenstellend erreichen und unser Projekt, trotz großen Themenumfang, Termingerecht abschließen.

## 3.3 Arbeit im Team

Abschließend kann man sagen, dass jeder in unseren Team Spaß an der Arbeit hatte, welches zumal auf die große Unterstützung untereinander, Privates Engagement für das Projekt und produktive Planungen, wie z.B. Scrum-Meetings, zurück zu führen ist. Generell war die Arbeit innerhalt des Teams, als sehr angenehm zu bewerten. Es herrschte ein gutes Arbeitsklima.