

zum Laufen bekommen

- JavaFX anschauen und raussuchen, was wir brauchen
- Template einrichten und Code verstehen
- Physics Engine verstehen
- Team in QIS und GitHub, Repository gecloned ✓
- Template mit Maven bauen, kompilieren, ausführen
- Sanity Checks (diese kurz dokumentieren)
- Prototyp
- Anforderungen dokumentiert → Story Cards, Anforderungsdokument ✓
- Projektplan → Arbeitspaket, Zeitplanung, Meilensteine ✓
- Verantwortlichkeitsverteilung, schriftl. festhalten ✓

↳ UML - Diagramm

• Model-View-Controller (Tutorial)

- Code: → Startseite (mit Levels?)
- Editor (Objekte platzieren)
- Kettenreaktion läuft

Meeting 30.04.

- Norxi: Settings, CSS, Story Cards
- Finn: Object Wrapper bauen
- Zeldu: Story Cards, Anwendungsfalldiagramm, Zeitplanung (Anforderungsdokument)

→ Anleitung anfangen, designen

Meilensteine:

- Physics Engine funktioniert
- JavaFX MVC implementiert
- Physics Objekte implementiert
- funktionierender Start- und Settings-View
- funktionierender Editor
- Spiel kann durchgeführt werden

Next Meeting:

- Settings
- Physics
- Doku

• CSS: Norxi
↳ Wie mit JavaFX?

• JBox2D: Finn

• Dokumentation: Zeldu
(Jira)

→ Tutorial anschauen

Verantwortlichkeiten

- Finn: Backend, Architektur
- Norxi: Frontend, Tests
- Zeldu: Projektplanung, Dokumentation
- Naur: ?

Anforderungsdokumente

- Story Cards
- Anwendungsfalldiagramm
- Anforderungsdokument

Projektplan

- Zeitplanung: Netzplan (Monday Dev)
- Meilensteinplanung: Zeitstrahl
- Arbeitspakete?

→ Architektur: Model-View-Controller mit Observer-Pattern

→ Sanity Checks + Doku