

# **Archive of Austrian Video Games Development**

ASE Milestone Release 2

# Inhalt

- Projektbeschreibung
- Features
- Aktueller Entwicklungsstand
- to do
- the good stuff
- the not so good stuff
- Demo

# Projektbeschreibung

## **Archive for Austrian Video Game Development**

Präsentation von Spielen und Entwicklern

Archiv (Langzeit Entwicklung)

externer Auftraggeber (Austrian Game Jam)

Ruby on Rails als Framework vorgegeben

# Userstories

vordefiniert durch Auftraggeber

sortiert nach Prioritäten (Input für Sprints)

Den jeweiligen Meilensteinen zugeordnet

# Features

- Invite-only (von Administrator)
- Spiele und Entwickler Seiten (Erstellen, Bearbeiten)
- Spiele Genres (Benutzerdefiniert, Vorschlag von bestehenden)
- Querverweise (z.B.: Spiel -> Entwickler)
- externe Links
- Video embedding (z.B.: YouTube)
- Tags und Suche
- Versionierung von Einträgen
- ...

# Besondere Features

- Inhalt mit semantischer Bedeutung
  - Tags
  - Genres
  - Crossreferences
  - andere freie Relationen
- Suche (Volltext mit Priorisierung)
- Data Mining
- Versioning und History
- Umfangreiche Tests (Behaviour Driven)  
und Erweiterbarkeit

# Aktueller Entwicklungsstand

- ~ 50% der features implementiert
  - Invite-only (von Administrator)
  - Spiele und Entwickler Seiten (Erstellen, Bearbeiten)
  - Spiele Genres (Benutzerdefiniert, Vorschlag von bestehenden)
  - Querverweise (z.B.: Spiel -> Entwickler)
  - externe Links
  - Video embedding (z.B.: YouTube)
  - Inhalte mit semantischer Bedeutung
- BDD verwendet und implementiert

# to do

- Refactoring der Tests
- Killer Features wie
  - Suche
  - Statistiken
  - Revision Control



# the good stuff (1)

- Rails wird einfacher bis komfortabel
  - Code wird durch conventions einfach wartbar
- Git verursacht bei keinem Teammitglied mehr ein Problem
- Viel Arbeitsaufwand investiert (bis zu 100 Stunden pro Person ...)

# the good stuff (2)

- Bei Meetings meist alle dabei
- Kommunikation Auftraggeber
- Kommunikationskanäle
  - mind 1x/Woche Meeting (in person)
  - Skype
  - Google Docs
  - Google Groups (intern, extern)

# not so good stuff

- Hohe Learning Curve
  - Ruby/Rails -> Tutorials und viel lesen
  - BDD/Cucumber
  - Git
- Schlechte Google Help bei speziellen Problemstellungen
- Kommunkation da sprachliche Barrieren
- Vorwissen variiert stark
- Unterschätzen der User Stories (Aufwand)
- hohe Abnahmekriterien/Genauigkeit
- heterogene Systeme/native libraries (Mac, Linux, Windows)

# Demo