

Projekt: JBomberman  
  
Projektplan

Pascal Kistler

Silvan Adrian

Fabian Binna

# Änderungsgeschichte

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Version** | **Änderung** | **Autor** |
| **<Datum>** | <Nr> | <Art der Änderung> | <Vorname Name> |
|  |  |  |  |
|  |  |  |  |

# Inhalt

[Änderungsgeschichte 2](#_Toc317331516)

[Inhalt 3](#_Toc317331517)

[1. Einführung 4](#_Toc317331518)

[1.1 Zweck 4](#_Toc317331519)

[1.2 Gültigkeitsbereich 4](#_Toc317331520)

[1.3 Referenzen 4](#_Toc317331521)

[2. Projekt Übersicht 5](#_Toc317331522)

[2.1 Zweck und Ziel 5](#_Toc317331523)

[2.2 Lieferumfang 5](#_Toc317331524)

[2.3 Annahmen und Einschränkungen 5](#_Toc317331525)

[3. Projektorganisation 6](#_Toc317331526)

[3.1 Organisationsstruktur 6](#_Toc317331527)

[3.2 Externe Schnittstellen 6](#_Toc317331528)

[4. Management Abläufe 7](#_Toc317331529)

[4.1 Kostenvoranschlag 7](#_Toc317331530)

[4.2 Zeitliche Planung 7](#_Toc317331531)

[4.2.1 Phasen / Iterationen 7](#_Toc317331532)

[4.2.2 Meilensteine 7](#_Toc317331533)

[4.3 Besprechungen 7](#_Toc317331534)

[4.4 Abgabe 7](#_Toc317331535)

[5. Risikomanagement 8](#_Toc317331536)

[5.1 Risiken 8](#_Toc317331537)

[5.2 Umgang mit Risiken 8](#_Toc317331538)

[6. Arbeitspakete 9](#_Toc317331539)

[7. Infrastruktur 10](#_Toc317331540)

[8. Qualitätsmassnahmen 11](#_Toc317331541)

[8.1 Dokumentation 11](#_Toc317331542)

[8.2 Projektmanagement 11](#_Toc317331543)

[8.3 Entwicklung 11](#_Toc317331544)

[8.3.1 Vorgehen 11](#_Toc317331545)

[8.3.2 Unit Testing 11](#_Toc317331546)

[8.3.3 Code Reviews 11](#_Toc317331547)

[8.3.4 Code Style Guidelines 11](#_Toc317331548)

[8.4 Testen 11](#_Toc317331549)

[8.4.1 <Bezeichnung des Tests (z.B. Integrationstest oder Systemtest)> 11](#_Toc317331550)

# Einführung

## Zweck

Dieses Dokument beschreibt die Planung des SE2 Projekts.

## Gültigkeitsbereich

< Gültigkeitsbereich des Dokumentes>

## Referenzen

<Liste aller verwendeten und referenzierten Dokumente, Bücher, Links, usw.>

<Referenz auf ein Glossar Dokument, wo alle Abkürzungen und unklaren Begriffe erklärt werden>

<Die Quellen / Referenzen sollten mit dem Word Tool automatisch erstellt werden>

# Projekt Übersicht

JBomberman ist ein Geschicklichkeitsspiel, das die gleichen Regeln wie das klassische Bomberman befolgt. Es gibt nur einen Multiplayer Modus, in dem bis zu vier Spieler gegeneinander antreten können. Den Spielablauf übernimmt ein dedizierter Server.

## Zweck und Ziel

<Sinn und Zweck des Projektes, sowie Zielsetzung und auch persönliche Ziele>

## Lieferumfang

* Source Code
* Dokumentation

## Annahmen und Einschränkungen

<Annahmen die für diesen Projektplan getroffen werden und Einschränkungen denen er unterliegt>

# Projektorganisation

<Kurze Beschreibung der Projektstruktur>

## Organisationsstruktur

<Projektmitglieder nennen und deren Aufgaben und Verantwortungen aufzählen>

## Externe Schnittstellen

<Ansprechpartner, verantwortliche Personen, Betreuer, usw. aufzählen>

# Management Abläufe

## Kostenvoranschlag

<Wie viel Zeit steht zur Verfügung? In welcher Zeitspanne läuft das Projekt? Wird das Projekt früher beendet, dafür wöchentlich mehr gearbeitet?>

## Zeitliche Planung

<Kurze Beschreibung der zeitlichen Planung und mit einer Grafik einen Überblick über die Phasen, Iterationen und Meilensteine geben. Das Datum des Eintreffens der Meilensteine sollte in der Phasenübersicht ersichtlich sein.>

### Phasen / Iterationen

#### <Bezeichnung der einzelnen Phasen>

<Kurze Beschreibung und Dauer der Phase angeben>

##### <Bezeichnung der einzelnen Iterationen>

<Kurze Beschreibung und Dauer der Iteration angeben>

<Vorgehen bei Iterationsplanung und Iterationsassessment>

### Meilensteine

#### <Bezeichnung der einzelnen Meilensteine>

<Setzen Sie in Ihrem Projekt 6-8 Meilensteine. Kurze Beschreibung der Meilensteine mit genauem Datum. In der Regel auf Ende jeder Iteration einen Meilenstein setzen (diese Faustregel gilt nur für die SE-2Projekte, in realen Projekten haben Sie oft deutlich mehr Iterationen als Meilensteine, weil Meilensteine dort die nach aussen kommunizierten Ereignisse sind). Schreiben Sie zu jedem Meilenstein auf, welche Arbeitsprodukte (work products) Sie dann abliefern werden . Spezifizieren Sie wenn nötig auch den Fertigstellungsgrad der Arbeitsprodukte, z.B. „Zentrale Use Cases ‚fully dressed‘, restliche UCs im ‚brief‘ Format“, oder „Architekturskizze inkl. Definition der Interfaces zwischen Sub-Systemen und Deployment Diagramm“>

## Besprechungen

<regelmässige Besprechungen nennen (wann, wo, wer, Ziel und/oder Grund)>

# Risikomanagement

## Risiken

<Verweis auf Dokument TechnischeRisiken.xlsx>

## Umgang mit Risiken

<Begründungen zur Tabelle. Weitere Beschreibungen zu Massnahmen und Vorbeugungen. Werden Reserven /Rückstellungen eingeplant? Wieso und wie viele? Wann werden Risiken qualitätssichernd überprüft (Vorgehen und Zeitpunkt(e) zur Neubeurteilung der Risiken)?>

# Arbeitspakete

<Definieren Sie in einem separaten Tool (Redmine oder XLS, o.a.) diejenigen Arbeitspakete, die Ihnen zu Beginn des Projektes schon mal einfallen. Zu Beginn können Pakete eher generisch ausfallen (z.B. ‚Domainmodell erstellen‘ oder ‚GUI Programmieren‘, oder ‚ Usability Testing‘).

Denken Sie daran, dass es in jedem Projekt auch eine ganze Reihe von ‚overhead‘-Tätigkeiten gibt: Projektleitung und -Sitzungen, Aufsetzen der Server und Werkzeuge, Qualitätsmassnahmen, Schlusspräsentation erstellen und anderes mehr – alles Tätigkeiten, die nicht direkt zum Software-Produkt führen. Planen Sie genug Zeit auch für diese Tätigkeiten ein.

Dann sollte es in jeder Projektplanung einige projektspezifische Arbeitspakete geben, wie sie so nicht in anderen Projekten vorkommen (z.B. ‚ Level-Editor entwerfen‘, oder ‚Verifikation der Zahlungsangaben programmieren‘). Wenn es keine solchen Arbeitspakete gibt, ist der Projektplan zu generisch. Das heisst auch, dass man sich zu wenig Gedanken über die anfallenden Arbeiten gemacht hat.

Später im Projekt werden Sie die Arbeitspakete noch verfeinern, verschieben, jemandem zuordnen, neu schätzen, etc. Deswegen empfiehlt sich der Einsatz eines Werkzeuges wie Redmine.

Dokumentieren Sie URL und Logins auf das Projektmanagement Tool, in welchem die Meilensteine und die dazugehörigen Arbeitspakete erfasst sind.>

# Infrastruktur

<Benötigte Infrastruktur aufzählen. Spezielle Geräte, Laptop , Tools usw. und nötigenfalls aufzeigen für welche Bereiche diese verwendet werden. Eventuell auch Verfahren beschreiben (auf Tools bezogen).>

# Qualitätsmassnahmen

<Was wird unternommen damit das Produkt des Projektes, sowie dessen gesamter Verlauf eine hohe Qualität erreicht? Übersicht in einer Tabelle geben mit Massnahmen, Zeitraum und Ziel der Massnahme>

## Dokumentation

<Wo befinden sich die Dokumente (SVN oder Git Server) und wie wird deren Qualität sichergestellt?>

## Projektmanagement

<Welches Tool wird für Projektmanagement eingesetzt (z.B. Redmine oder Trac) und wie erfolgt dieser Einsatz? Dazugehörige Links und Logins (Gastbenutzer).>

## Entwicklung

<Wo befindet sich der Source Code (z.B. SVN oder Git) und wie wird dessen Qualität sichergestellt?>

### Vorgehen

<Vorgehen in der Entwicklung>

### Unit Testing

<Wo werden welche Unit Tests geschrieben um die Qualität sicherzustellen? Wie wird die Testabdeckung sichergestellt (z.B. durch EclEmma)?>

### Code Reviews

<Werden Code Reviews gemacht und wie werden diese gemacht?>

### Code Style Guidelines

<Welche Code Style Guidelines werden angewendet? Sie brauchen keine eigenen Guidelines zu erfinden. Am besten referenzieren Sie existierende Guidelines, mit denen Sie einverstanden sind. Evtl. noch Abweichungen dazu dokumentieren>

## Testen

### <Bezeichnung des Tests (z.B. Integrationstest oder Systemtest)>

<Beschreibung der Durchführung, Umsetzung und Umfang der Tests. Wenn möglich mit Mengenangaben (wieviele Tests)>