



SOFTWAREENTWICKLUNGSPRAKTIKUM 2015

N-PONG1

Software-Entwicklungspraktikum (SEP)
Sommersemester 2015

Angebot

Auftraggeber
Technische Universität Braunschweig
Institut für Softwaretechnik und Fahrzeuginformatik (ISF)
Prof. Dr. Ina Schaefer
Mühlenpfordtstr. 23
D-38106 Braunschweig

Betreuer: Sven Schuster, M.Sc.

Auftragnehmer:

Name	E-Mail-Adresse
Christian Bunzeck	c.bunzeck@tu-braunschweig.de
Niklas Czerner	n.czerner@tu-braunschweig.de
Felix Franzke	f.franzke@tu-braunschweig.de
Niklas Hinze	n.hinze@tu-braunschweig.de
Adrian Hoff	a.hoff@tu-braunschweig.de
Manfred Küppers	m.kueppers@tu-braunschweig.de

Braunschweig, 24. April 2015

Inhaltsverzeichnis

1	Einleitung	4
1.1	Ziel	4
1.2	Motivation	4
2	Formale Grundlagen	5
3	Projektablauf	6
3.1	Meilensteine	6
3.2	Geplanter Ablauf	7
4	Projektumfang	8
4.1	Lieferumfang	8
4.2	Kostenplan	8
4.3	Funktionaler Umfang	8
5	Entwicklungsrichtlinien	9
5.1	Konfigurationsmanagement	9
5.2	Design- und Programmierrichtlinien	9
5.3	Verwendete Software	9
6	Projektorganisation	10
6.1	Schnittstelle zum Auftraggeber	10
6.2	Schnittstelle zu anderen Projekten	10
6.3	Interne Kommunikation	11
7	Glossar	12

Abbildungsverzeichnis

3.1	Beispielhaftes Gantt-Diagramm	7
-----	---	---

1 Einleitung

Das Projekt n-Pong entsteht im Rahmen des Software-Entwicklungs-Praktikums 2015. Es handelt sich dabei um den Spieleklassiker Pong, der um eine Mehrspielerkomponente erweitert ist, wodurch sich ad-hoc beliebig viele Spieler ins laufende Spiel ein- und ausklinken können.

1.1 Ziel

Primäres Ziel ist das Programmieren von komplexerer Software und die dabei gewonnene Erfahrung ist hilfreich auf dem Gebiet des Software-Engineerings. Android ist das unterstützte Betriebssystem. Ein Gerät steht dafür im Mittelpunkt und stellt die Server-Komponente dar, auf der das eigentliche Spiel stattfindet. Hingegen sind die Steuerelemente auf die Android-Geräte der Spieler ausgelagert. Beide Komponenten kommunizieren per WLAN. Außerdem ist das Spiel mit Plugins erweiterbar. Dazu verwenden wir ein Plugin-Framework. Das angestrebte Resultat ist ein Spiel, welches dem Nutzer ein positives Spielerlebnis bietet.

1.2 Motivation

n-Pong verbindet Elemente des Anbeginns der Videospiel-Ära mit zeitgemäßem Multiplayer. Durch diesen wird aus dem alten Eins gegen Eins ein gemeinschaftliches Erlebnis.

2 Formale Grundlagen

Die folgenden Grundlagen der zu entwickelnden Software sind ein wesentlicher Bestandteil des Projektes. Sie bilden das Grundgerüst des gesamten Projektes.

Als Programmiersprache wird Java verwendet, welches mit der IDE „Android Studio“ für mobile Android Geräte geschrieben wird. Wir halten uns an die Richtlinien der „Google-Code-Convention“. Die Formalitäten bezüglich der verwendeten Software sind im Kapitel 5.3 auf Seite 9 weiter ausgeführt.

Die Programmier- und Anwendungssprache ist Englisch.

3 Projektablauf

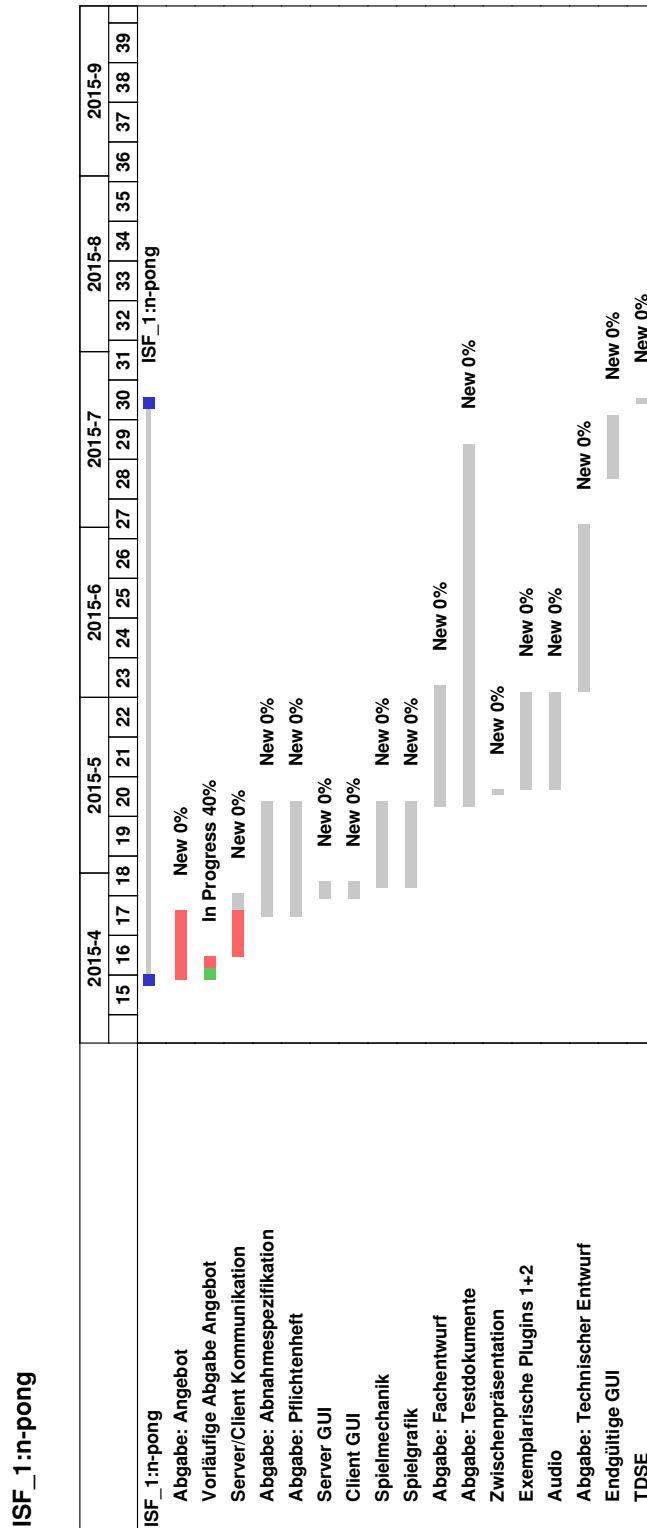
Das Projekt ist am 13.04.15 gestartet. Der Ablauf ist durch die in 3.1 aufgeführten Meilensteine beschrieben.

3.1 Meilensteine

Die Meilensteine des n-Pong Projektes befinden sich in chronologischer Reihenfolge in folgender Tabelle:

Nummer	Meilenstein	Dokumente	Abgabetermin
1	Projektstart	-	13.04.15
2	Interne Abgabe des Angebot-Dokumentes	Angebot	16.04.15
3	Angebot	Angebot	24.04.15
4	Server/Client-Kommunikation	-	27.04.15
5	Server-GUI	Quelltext	29.04.15
6	Client-GUI	Quelltext	29.04.15
7	Spielmechanik	Quelltext	13.05.14
8	Spielgrafik	Quelltext	13.05.15
9	Abnahmetestspezifikation	Pflichtenheft	13.05.15
10	Zwischenpräsentation	Prototyp, Präsentation	15.05.15
11	Abgabe Fachentwurf	Fachentwurf	03.06.15
12	Exemplarische Plugins 1+2	-	02.06.15
13	Audio	-	02.06.15
14	Abgabe Technischer Entwurf	Technischer Entwurf	01.07.15
15	Abgabe Testdokumentation	Testdokumentation	15.07.15
16	Endgültige GUI	-	20.07.15
17	TDSE	N-Pong	23.07.15

3.2 Geplanter Ablauf



04/24/2015

1/1

Abbildung 3.1: Beispielhaftes Gantt-Diagramm

4 Projektumfang

In diesem Kapitel wird der Projektumfang beschrieben.

4.1 Lieferumfang

Im fertigen Softwarepaket zu diesem Projekt enthalten sind:

1. Der Quellcode des Programms
2. Die ausführbare Anwendung
3. Die Spezifikation und der Entwurf der Software
4. Die Testprotokolle
5. Mini-Tutorial

4.2 Kostenplan

Für das Projekt wird von einem Mittelwert von 3,5 Stunden Arbeit am Wochentag pro Entwickler ausgegangen. Das bedeutet für den Arbeitszeitraum vom 13. April bis zum 23. Juli mit 71 Wochentagen 230 Stunden pro Entwickler und somit 1380 Stunden insgesamt. Das macht bei einem Stundenlohn von 100,00 Euro einen Entwicklungswert von rund 138.000,00 Euro.

4.3 Funktionaler Umfang

Das fertige Produkt ist eine Multiplayer Spiele-App mit Client- und Serverfunktion. Dabei fungiert eines der Geräte über die App als Server, der Platz für Spieler bietet.

Diese können einem existierenden Server mittels Clientfunktion beitreten und eine der Schläger-Plattformen steuern. Optional kann der Server mit einem Passwort versehen werden um das Beitreten von unbekannten Spielern in der Nähe zu vermeiden. Über Plugins kann das Produkt mit weiteren Features und Bonusmaterialien ausgestattet werden. Plugins sind zum Beispiel weitere Spielmodi.

5 Entwicklungsrichtlinien

In diesem Kapitel ist festgehalten, wie die Struktur des Codes, sowie das Codedesign aussieht.

5.1 Konfigurationsmanagement

Für die Verwaltung von Code und Dokumenten wird ein vom ISF bereitgestellter SVN-Repository genutzt. Außerdem sind etliche Arbeitsdokumente und Diagramme im Redmine hinterlegt. Die Ordnerstrukturen werden erst im Laufe der Entwicklung entstehen, jedoch wird voraussichtlich ein zentraler Ordner für LaTeX-Dokumente und einer für Code angelegt werden. Beim Hochladen von Code in das SVN-Repository wird ausreichende Kommentierung, sowie sprechende Methoden-, Klassen- und Variablennamen gewährleistet. In dem Fall eines nicht ausführbaren Codefragments wird eine Anmerkung sowohl in der entsprechenden Klasse als auch beim SVN-Commit beigefügt.

5.2 Design- und Programmierrichtlinien

Zur Dokumentation von Methoden und Klassen wird JavaDoc verwendet. Unübersichtliche Stellen innerhalb von Methoden sind zudem selbsterklärend zu kommentieren. Im Allgemeinen hält sich das Projekt, wie bereits erwähnt, an die Coding-Guidelines der Google-Code-Convention, um unterschiedliche Coding-Stile und daraus resultierende Unübersichtlichkeit zu vermeiden. Diese basieren auf den „Code Conventions for the Java Programming Language“.

5.3 Verwendete Software

Android Studio ist die verwendete Entwicklungsumgebung. Diese bietet, neben einer guten Dokumentation, auch einen Geräte-Emulator, der das Entwickeln entscheidend vereinfacht. Für das Erstellen von Diagrammen in der Projekt-Dokumentation wird außerdem der, im vom ISF bereitgestellten Redmine-Portal integrierte, Gantt-Creator verwendet. Anderweitige Diagramme entstehen mit Visio.

6 Projektorganisation

Die Kommunikation im Team läuft bisweilen sehr regelmäßig ab. Regelmäßige Treffen und viel Schriftverkehr ermöglichen schnelle Ergebnisse und gute Arbeitsteilung.

6.1 Schnittstelle zum Auftraggeber

Einmal in der Woche findet ein regelmäßiges Meeting mit dem Auftraggeber statt, in dem wir den aktuellen Leistungsstand diskutieren und bewerten. Das Treffen findet Donnerstags um 15:45 Uhr im Raum IZ 349 statt. Zudem ist jederzeit die Kommunikation per E-Mail oder persönlichen Erscheinens im Büro möglich, um kurzfristig in Kontakt zu treten. Den Auftraggeber erreichen wir unter folgender Adresse:

Sven Schuster, M.Sc.
Institut für Softwaretechnik und Fahrzeuginformatik (ISF)
Technische Universität Braunschweig
Mühlenpfordtstr. 23
D-38106 Braunschweig
Germany

Room: IZ 360b
Tel: +49-531-391-2294
email: s.schuster@tu-bs.de

6.2 Schnittstelle zu anderen Projekten

Schnittstellen zu anderen Projekten sind in keiner erwähnenswerten Ausprägung vorhanden.

6.3 Interne Kommunikation

Zur Kommunikation untereinander wird einen E-Mail-Verteiler sowie die Smartphone App "WhatsApp" verwendet, die auch kurzfristige Meetings und Absprachen von Unterwegs ermöglicht. Dadurch entstehen regelmäßige, fast tägliche Treffen, in denen offene Punkte diskutiert und bearbeitet werden. Jedes Teammitglied macht sich dabei Notizen zu den Punkten, die es später genau ausarbeiten wird. Wie bereits erwähnt wird für den Austausch von Daten jeglicher Art das Subversionierungssystem SVN genutzt.

7 Glossar

Hier werden Fachbegriffe erklärt.