



Softwarepraktikum

SS 2009



Allgemeines

- Pflichtveranstaltung Bachelor Informatik
- 6 ECTS Punkte
- Gruppen mit 5-6 Teilnehmern
- Gruppennoten - keine Klausur / Prüfung
- Anmeldung beim Prüfungsamt nicht vergessen!



Organisation

- Leitung
 - Evren Ermis, Daniel Dietsch, Stephan Arlt
- Tutoren
 - Justus Bisser, Benjamin Bäurle, Florian Bäurle, Felix Ruzzoli, Philipp Vath



Termine

- Heute bis 23:59 Uhr Fragebogen ausfüllen
- Mittwochs 14-18 Uhr
 - Sprechstunde im Pool
 - keine Anwesenheitspflicht
- Wöchentliches Gruppentreffen mit Tutor
 - Termin nach Vereinbarung
 - Anwesenheitspflicht!

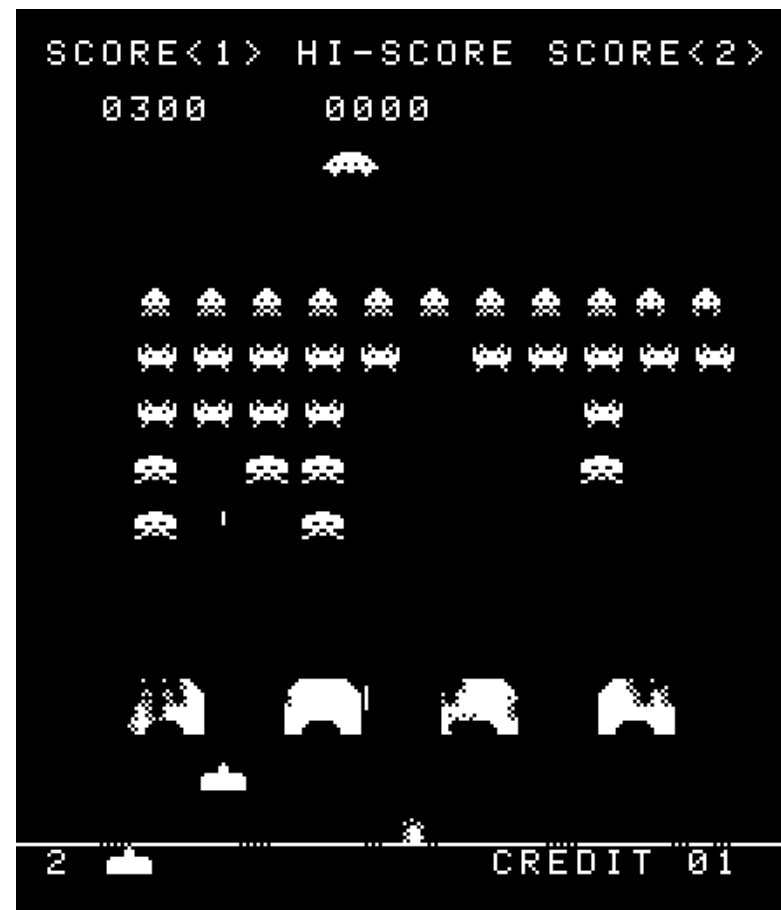


Richtlinien

- Teilnahme an wöchentlichen Gruppentreffen
 - Scrum-Meeting (später mehr)
 - Dauer
 - ca. 2 Stunden
 - Inhalt
 - Was wurde erreicht?
 - Was soll erreicht werden?
 - Welche Probleme gibt es?
- Reporting

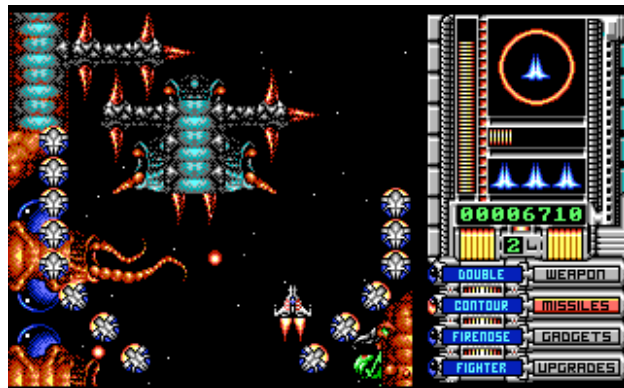
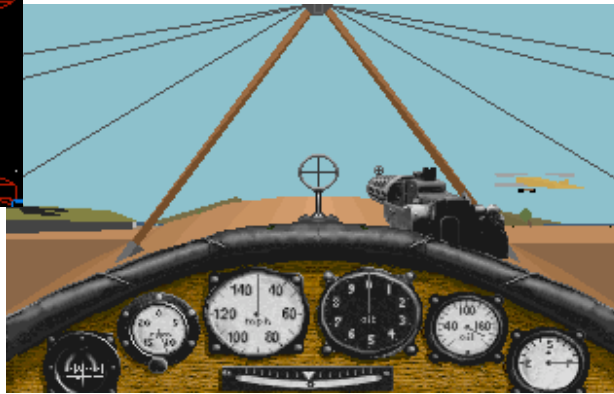
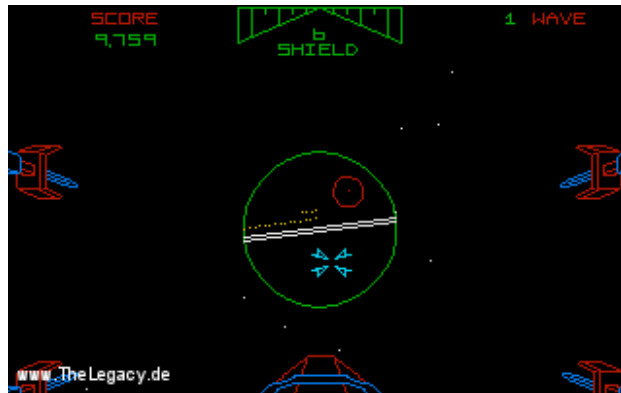


Genre “Fliegerspiel”





Klassiker





Neuere





Aufgabenstellung

- Entwickeln Sie
 - ein gutes Produkt, das Ihnen (und uns) gefällt!
 - in zufälligen Gruppen von 5-6 Teilnehmern
- Programmiersprache
 - C#
 - Visual Studio
- Plattform
 - Windows XP oder höher
 - XBOX 360 (optional)
- Framework
 - XNA 3.0 oder höher



Aufgabenstellung (II)

- Bei genügend Interessenten (Fragebogen!)
 - iPhone Developer University Program
 - Programmiersprache
 - Objective-C
 - Xcode
 - Plattform
 - Mac OS X
 - iPhone oder iPod touch (optional)
 - Framework
 - iPhone SDK



Bewertung

Gebrauchsqualität nach ISO 9126

Effektivität

- Spielbarkeit
- Plausibilität

Produktivität

- Relative Ressourcennutzung
- Einarbeitungszeit

Sicherheit

- Abstürze

Zufriedenheit

- Spielspass
- Dauerspass
- Erscheinung
- Spielidee



Ablauf

- Gesamtzeit 14 Wochen
- Woche 0
 - Organisation
- Woche 1-3
 - Analyse & Entwurf
- Woche 4-8
 - Implementation (beta)
- Woche 9-13
 - Implementation (final)



Ablauf (II)

- Allgemein
 - Abgaben jeweils Samstags 23:59 Uhr
 - Präsentationen jeweils Mittwochs zwischen 14-18 Uhr
- Roadmap
 - 09.05. Abgabe GDD (beta)
 - 16.05. Abgabe Klassendiagramm (beta)
 - 13.06. Abgabe GDD (final)
 - 17.06. Präsentation Programm (beta)
 - 20.06. Abgabe Programm (beta)
 - 27.06. Abgabe Klassendiagramm (final)
 - 22.07. Präsentation Programm (final)
 - 25.07. Abgabe Programm (final)



To Do

- Füllen Sie bis 23:59 Uhr den Fragebogen aus!
 - Wichtig: Ohne einen IIF-Account können Sie nicht teilnehmen
 - Sie müssen Ihren Account-Namen im Fragebogen angeben
- Melden Sie sich im Vorlesungsverzeichnis an!
- Erstellen Sie ein WWW-Passwort!
- Alle Informationen unter:
 - <http://sopra.informatik.uni-freiburg.de>



Softwarepraktikum 2009

Game Design Document



Worum geht es?

- Jeder mag Spiele
- Alle haben tolle Ideen
- Aber
 - Wir könnten doch...
 - Wir sollten unbedingt...
 - Und dann noch dies und das...
- Man benötigt durchdachtes Vorgehen



Durchdachtes Vorgehen

- Design Treatment
- Preliminary Design
- Final Design
 - Produkt Spezifikation
 - Graphic Bible
 - Interaktives Screenplay



Design Treatment

- Genre
- Key features
- Spielerlebnis
- Look-And-Feel
- Spielstruktur
- Charaktere/Einheiten/Spielobjekte
- Zielgruppe
- Zielplattform



Preliminary Design

- Technische Spezifikationen
- Hintergrundgeschichte
- Eigenschaften der Charaktere
- Umgebungen/Missionen
- Besondere Objekte
- Spezielle Aktionen
- Steuerung
- Statistiken



Final Design

Design Treatment + Preliminary Design

+

- Produkt Spezifikation
- Graphic Bible
- Interaktives Screenplay



Final Design

- Design Treatment und Preliminary Design
 - sind **keine** formalen Dokumente
 - werden nur Gruppen intern verwendet
- Final Design
 - formales Dokument
 - Abgabe als GDD
 - beinhalten Erkenntnisse aus DT und PD



Final Design

- So detailliert wie möglich
- Alles ist relevant
- Schreiben zwingt zum Nachdenken
- Liste der Eigenschaften:
 - Was muss unbedingt ins Spiel?
 - Was muss nicht, wäre aber gut?
 - Was kann warten bis man Zeit hat?
- Bilder/Zeichnungen/Design Art(Graphics Bible)
- Dialoge/Storyline/Events(Storyboard)



Softwarepraktikum 2009 Software Engineering



Softwarepraktikum 2009

Software Engineering

to engineer |,enjə'ni(ə)r|

design and build (a machine or structure)



Softwarepraktikum 2009

Software Engineering

to engineer |,enjə'ni(ə)r|

design and **build** (a machine or structure)

- Entwurf und Planung
- Realisierung und Auslieferung



Es war einmal...

- Softwarekrise Mitte der 60er
- viele Softwareprojekte scheitern wegen wachsender Komplexität
- 1968 wird auf einer Nato-Tagung Software Engineering “erfunden”



Wege aus der Softwarekrise

- Objektorientierte Programmierung
- Aspektorientierte Programmierung
- Modularisierung durch Bibliotheken
- Vorgehensmodelle aus anderen Ingenieursdisziplinen
- Software-“Philosophie”

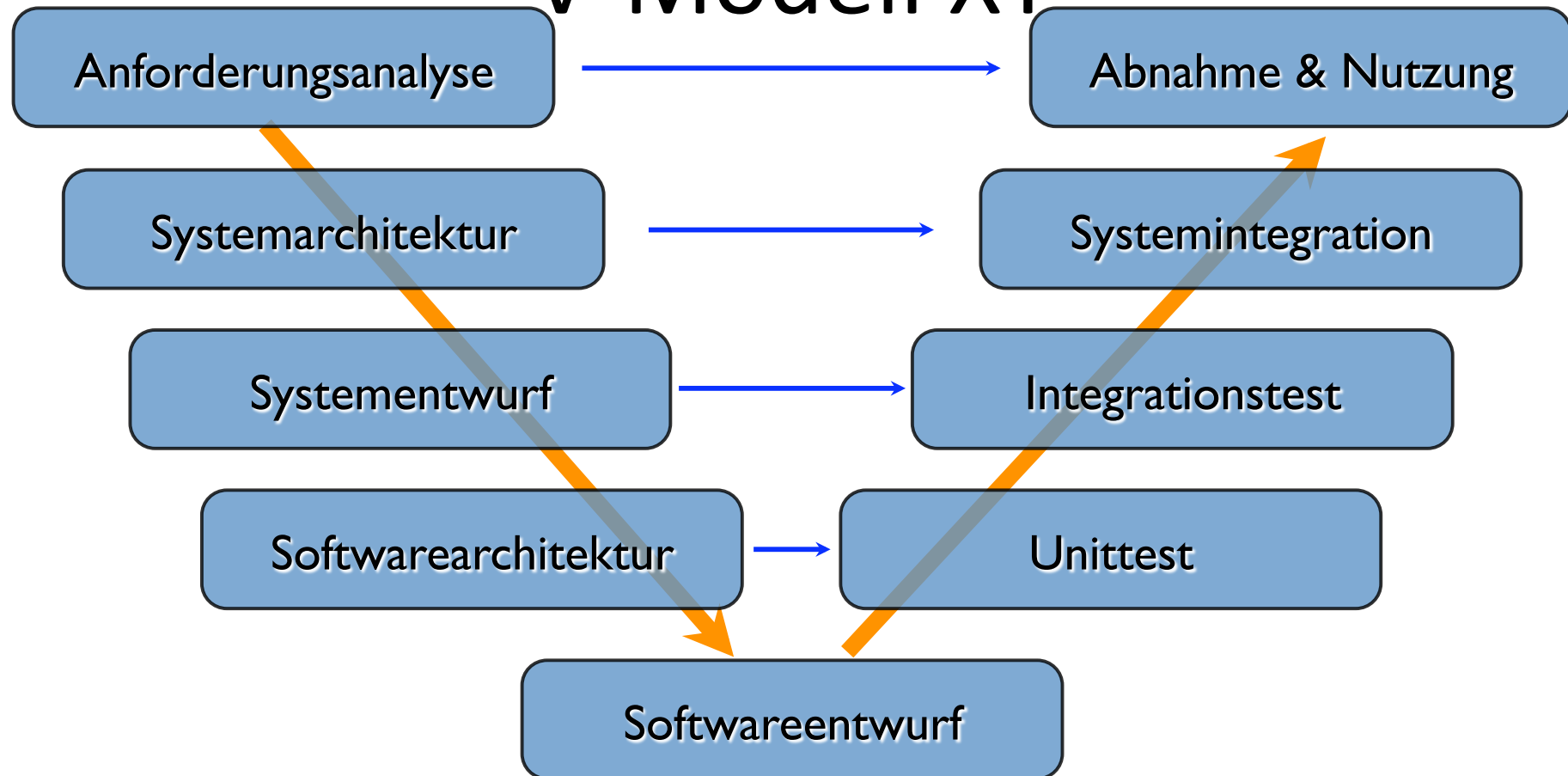


Software Entwicklungsprozesse

- Wasserfall-Modell
- V-Modell und V-Modell XT
- viele mehr (VL Softwaretechnik)



V-Modell XT



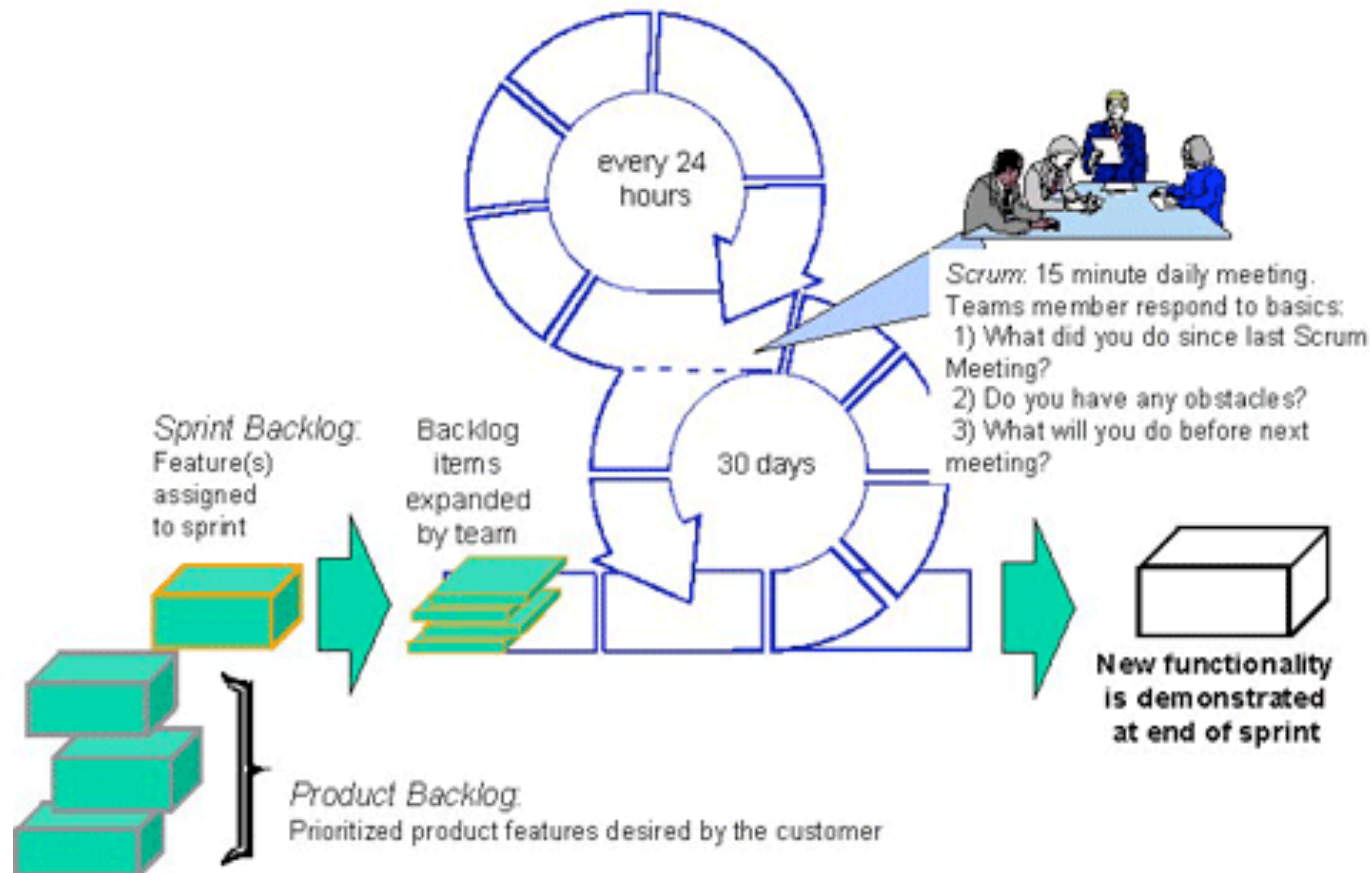


Software Philosophie

- Agile Methoden (z.B. Scrum)
- Rapid Prototyping
- Extreme Programming
- viele mehr (VL Softwaretechnik)



Scrum



Quelle: <http://agile-scrum.de/>



Vorgehensmodelle

pro

contra

V-Modell

- Kostenkontrolle
- Qualitätssicherung
- Dokumentation

- Bürokratie
- Zeitaufwand
- starre Vorgaben

Scrum

- kurze Reaktionszeiten
- wenig Overhead
- ständiger Kundenkontakt

- hohe Verantwortung des Einzelnen
- wenig Dokumentation



Vorgehensmodelle

pro

contra

V-Modell

- Kostenkontrolle
- Qualitätssicherung
- Dokumentation

- Bürokratie
- Zeitaufwand
- starre Vorgaben

Scrum

- kurze Reaktionszeiten
- wenig Overhead
- ständiger Kundenkontakt

- hohe Verantwortung des Einzelnen
- wenig Dokumentation



Scrum: Rollenverteilung

Product Owner (Sopra Leitung)

- legt die Richtung fest
- Entscheidet, ob der Weg gut ist oder nicht.
- bestimmt Prioritäten im Product Backlog

Scrum Master (Tutoren)

- sorgt dafür, dass das Team produktiv ist
- unparteiisch
- sorgt für Transparenz während der Entwicklung

Team (Studenten)

- bestimmt den Aufwand der Sprint Backlog Elemente
- implementiert das Programm



Scrum: Artefakte

Product Backlog (Sopra Leitung)

- Produktanforderungen
- Anforderungen für den aktuellen Sprint

Sprint Backlog (Tutor, Studenten)

- Auswahl von Product Backlog Elementen
- Umsetzungsvorschläge
- Arbeitsteilung

Implement List (Studenten)

- Hindernisse in der Implementierung
- Nur ein Hilfsmittel, das nicht eingesetzt werden muss.



Scrum: Artefakte

Product Backlog	Game Design Document
Sprint Backlog	Elemente aus GDD Zuordnung auf Personen erwartete Reife
Implement List	Element aus GDD Aufgetretenes Problem Priorität



Scrum: Zeitplan

Sprint	7 Tage statt 30 Tage
Scrum Meeting	1x pro Woche mit Tutor, so oft wie möglich im Team



Scrum Meetings

Chicken:

Let's start a restaurant!

Pig:

What would we call it?

Chicken:

Ham n' Eggs!

Pig:

No thanks. I'd be committed, but you'd only
be involved!

von Ken Schwaber



Scrum Meetings

- **Pigs** sind die Team-Mitglieder
Note und Schein hängen vom Erfolg des Projekts ab
- **Chickens** sind die Tutoren
Sollen bei der Durchführung helfen haben
aber kein Risiko



Scrum Meetings

- Besprechen Sie täglich den Projektstand (z.B. über IM, IRC oder E-Mail).
- Teilen Sie die Arbeit untereinander auf.
- Machen Sie verbindliche Vereinbarungen, was bis wann fertig sein muss.



Und jetzt?

- Füllen Sie den Fragebogen aus!
- Sie finden alle Informationen und den Zugang zum Fragebogen in unserem Wiki unter

<http://sopra.informatik.uni-freiburg.de>

- **Wenn Sie noch Fragen haben, fragen Sie!**