

PSIT3 – Resultate

Dieses Dokument beschreibt abschliessend die Resultate, die an den geforderten Meilensteinen abgegeben werden müssen. Laden Sie die Artefakte vor der Präsentation auf OLAT in den entsprechenden Team-Ordner hoch. Bringen Sie ein ausgedrucktes Exemplar mit in die Präsentation (Dozent).

Es bleibt Ihnen überlassen, wie Sie die Detailplanung vornehmen.

1 Projektskizze

1.1 Resultate

Projektskizze gemäss Vorgaben im Dokument „Projektskizze“

2 Analyse-Resultate

2.1 Übersicht

- Projektmanagement
- Anwendungsfälle
- Eine erste Architektur
- Zusätzliche Spezifikationen
- Anwendungsfalldiagramm
- System-Sequenzdiagramm
- Domänenmodell
- Glossar

2.2 Projektmanagement

Die bisherigen Aktivitäten und der zugehörige Aufwand sind erfasst und dargestellt. Der tatsächliche Aufwand und die erreichten Resultate werden mit dem geplanten Aufwand und den gesteckten Zielen verglichen und allenfalls Massnahmen getroffen. Die detaillierte Planung für die nächsten Iterationen wird dementsprechend aktualisiert. Die Risikoliste wird ebenfalls aufdatiert.

2.3 Anwendungsfälle

Liste aller aktuell bekannten Anwendungsfälle mit Priorisierung. Der wichtigste Anwendungsfall muss ausführlich („fully dressed“) ausformuliert werden. Dabei soll bevorzugt der Anwendungsfall verwendet werden, der in der ersten Iteration im Praktikum realisiert wird.

Weitere kritische Anwendungsfälle werden normal („casual“) ausformuliert, während der Rest der Anwendungsfälle noch kurz („brief“) beschrieben wird.

2.4 Anwendungsfalldiagramm

Die vorher erwähnten Anwendungsfälle werden in einem Anwendungsfalldiagramm mit den dazugehörigen Akteuren dargestellt.

2.5 Domänenmodell

Es ist ein Domänenmodell zu entwickeln, das die Konzepte der wichtigsten Anwendungsfälle darstellt. Anhand des Domänenmodells ist die Problemdomäne detailliert zu beschreiben.

2.6 Eine erste Architektur

Die erste Architektur soll eine erste Idee des Produkts vermitteln (Standalone-Applikation, Mobile App mit/ohne Server, C/S, P2P, 2D oder 3D-GUI) mit dem Sie die wichtigsten Anwendungsfälle abdecken und die wichtigen zusätzlichen Anforderungen erfüllen können.

2.7 Zusätzliche Spezifikationen

Hier werden weitere funktionale und vor allem die nichtfunktionalen Anforderungen gemäss dem FURPS+-Modell angegeben. Zusätzlich werden wichtige Regeln des Problemgebietes (z.B. Geschäfts- oder Spielregeln) und sonstige wichtige Informationen (z.B. weitere Vorgaben und Randbedingungen für die Entwicklung der Applikation) aufgelistet.

2.8 System-Sequenzdiagramm

Der Anwendungsfall, der vollständig ausformuliert wurde, wird als System-Sequenzdiagramm dargestellt.

2.9 Glossar

Die erste Version des Glossars, wo die wichtigste Begriffe des Problemgebiets erklärt und definiert werden, ist zu erstellen.

3 Design Resultate

3.1 Übersicht

- Projektmanagement
- Architektur
- Design-Klassendiagramm
- Klassenverantwortlichkeiten
- Interaktionsdiagramme
- Glossar

3.2 Projektmanagement

Die bisherigen Aktivitäten und der zugehörige Aufwand sind erfasst und dargestellt. Der tatsächliche Aufwand und die erreichten Resultate werden mit dem geplanten Aufwand und den

gesteckten Zielen verglichen und allenfalls Massnahmen getroffen. Die Planung für die nächsten Iterationen wird dementsprechend aktualisiert. Die Risikoliste wird ebenfalls aufdatiert.

3.3 Architektur

Beschreibung der gewählten Architektur mit Begründung des Entscheides.

3.4 Design-Klassendiagramm

Das Design-Klassendiagramm, das die wichtigsten Klassen des Lösungskonzepts umfasst. Das Diagramm kann logisch auch auf mehrere Seiten aufgeteilt werden. GUI-Klassen gehören **nicht** in dieses Diagramm.

3.5 Klassenverantwortlichkeiten

In einer Tabelle werden alle Klassen mit ihrer Verantwortlichkeit aufgeführt. Es ist zuerst die Gesamtverantwortlichkeit in **einem** Satz zu beschreiben. Daraus sind die Knowing- und Doing-Verantwortlichkeiten abzuleiten.

3.6 Interaktionsdiagramme

Für die wichtigsten Systemoperationen, die realisiert werden, muss je ein Interaktionsdiagramm erstellt werden. Dabei müssen beide Diagrammarten berücksichtigt werden.

3.7 Glossar

Das Glossar ist weiter auszubauen und zu pflegen.

3.8 GUI-Design

Falls ein anspruchsvolles GUI implementiert werden muss, soll zusätzlich ein GUI-Konzept erstellt werden (Skizzen der vorgesehenen GUI-Screens und ev. Beschreibung des Dialogablaufs)

4 Implementation/Resultat

4.1 Implementation

Folgende Aspekte werden bei der Implementation bewertet:

- Ist Code lauffähig
- Code-Qualität (Clean-Code, Namensgebung, Einhalten von Codierrichtlinien)
- Code-Dokumentation
- Test-Qualität (Unit-Test, Integrationstests, Systemtests)

4.2 Resultat

- Erreichte Ziele (im Vergleich zur Projektskizze)
- Umfang der implementierten Funktionalität
- Schwierigkeitsgrad
- Investierter Aufwand

5 Schlussbericht

Es ist ein Projektbericht (ca. 30 Seiten) zu erstellen mit den wichtigsten Resultaten des Projekts. Der Bericht soll sprachlich korrekt und logisch verfasst sein und zweckdienliche Bilder mit korrekter Nummerierung, Legenden und Referenzierung enthalten. Die im Bericht verwendeten Artefakte aus Projektskizze, Analyse und Design sollen angemessen gemäss Feedback der Betreuer aufdatiert werden. Diagramme sind ausreichend zu erläutern.

5.1 Inhaltsstruktur

- Projektidee
- Analyse
- Design
- Implementation
- Resultat
- Anhang

5.2 Projektidee

Die wichtigsten Punkte (alle ausser 2.2.6-2.2.9) der aufdatierten Projektskizze sollen hier nochmals zusammenfasst werden.

5.3 Analyse

Die wichtigsten Resultate der Problemanalyse sind hier zu dokumentieren:

- Domänenmodell
- Anwendungsfälle
 - Anwendungsfalldiagramm zur Übersicht
 - Wichtigste Use-Cases fully dressed, Rest casual oder brief

5.4 Design

Die wichtigsten Aspekte der gewählten Lösung sind zu dokumentieren:

- Architektur
- Design-Klassendiagramm
- Ausgewählte Interaktionsdiagramme
- Dokumentation wichtiger Designentscheide (z.B. Protokolle, State-Machines, Commands)

5.5 Implementation

Die wichtigsten Informationen zur Implementation der Lösung sowie die durchgeführten Tests sind hier zu dokumentieren:

- Testbericht: Zusammenfassung des gemachten Tests (Integrations- und Systemtests) und Testresultate
- Installationsanleitung
- Bedienungsanleitung (im Anhang)
- Code-Dokumentation
 - Vollständiger Code **nur** in elektronischer Form (primär OLAT, ansonsten CD, DVD)

- Der vollständige Quellcode (inkl. Unit-Tests) mitsamt einer kompilierten und lauffähigen Version wird in elektronischer Form abgegeben. Jede Klasse und alle öffentlichen Methoden und Attribute müssen (kurz) mittels JavaDoc dokumentiert werden. Der fertige JavaDoc-Kommentar muss ebenfalls (elektronisch) abgegeben werden. Der Sourcecode muss sich an eine einheitliche Richtlinie und die üblichen Standards halten (z.B. Java-Codierrichtlinie von Sun/Oracle).

5.6 Resultat

Die Projektergebnisse sind kurz zusammenzufassen und in Bezug auf die ursprüngliche Projektidee zu stellen:

- Zusammenfassung der erreichten Ziele
- Offene Punkte
- Ausblick auf mögliche Weiterentwicklungen

5.7 Anhang

- Projektmanagement
Beschreibung des Vorgehens/Methodik. Zusammenfassung der tatsächlichen Aufwände mit den geplanten Aufwänden pro Phase und insgesamt. Begründung von signifikanten Abweichungen.
- Verzeichnisse
 - Literatur-, Abbildungs-, Tabellenverzeichnis
 - Glossar
 - Bedienungsanleitung
 - Projektmanagement