**Projekt: Mosti**

System - Testdokumentation

[Dokumentstruktur basiert auf RUP „Dokument Test Evaluation Summary“]

# Dokumentinformationen

## Änderungsgeschichte

|  |  |  |  |
| --- | --- | --- | --- |
| Datum | Version | Änderung | Autor |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Inhalt

[1 Dokumentinformationen 2](#_Toc449603931)

[1.1 Änderungsgeschichte 2](#_Toc449603932)

[1.2 Inhalt 2](#_Toc449603933)

[2 Einführung (Introduction) 3](#_Toc449603934)

[2.1 Definitionen und Abkürzungen (Definitions, Acronyms, Abbreviations) 3](#_Toc449603935)

[2.2 Referenzen (References) 3](#_Toc449603936)

[2.3 Übersicht (Overview) 3](#_Toc449603937)

[3 Testvorgehen 3](#_Toc449603938)

[3.1 Funktionale Tests 3](#_Toc449603939)

[3.1.1 Grundtests (Smoke Tests) 3](#_Toc449603940)

[3.1.2 Modul- und Unittests 3](#_Toc449603941)

[3.1.3 Integrationstests 3](#_Toc449603942)

[3.1.4 System Acceptance Test 3](#_Toc449603943)

[3.2 Bedienbarkeit und Nutzerinterface (Usability) 3](#_Toc449603944)

[3.3 Datenschutz, Datensicherheit (Security) 3](#_Toc449603945)

[3.4 Leistungsanforderungen (Performance) 3](#_Toc449603946)

[3.5 Zuverlässigkeit 3](#_Toc449603947)

[3.6 Schnittstellen 3](#_Toc449603948)

[3.7 Wartung und Servicefunktionen 4](#_Toc449603949)

[3.8 Installation 4](#_Toc449603950)

[3.9 Internationalisierung / Lokalisierung 4](#_Toc449603951)

[3.10 Testautomatisierung 4](#_Toc449603952)

[3.11 Verfolgbarkeit (Traceability) 4](#_Toc449603953)

[4 Übersicht der Testpläne 4](#_Toc449603954)

[5 Freigabe von Testergebnissen 4](#_Toc449603955)

# Einführung (Introduction)

## Definitionen und Abkürzungen (Definitions, Acronyms, Abbreviations)

*Vgl. separates Glossary-Dokument im Repository, Dateiname: Glossary.docx*

## Referenzen (References)

*Buch: Craig Larman: UML 2 und Patterns angewendet (2005)*

## Übersicht (Overview)

In diesem Dokument werden verschieden Testverfahren erläutert die wir für unsere Software entwickelt haben. Wir gehen darauf ein wie unsere Tests durchgeführt und diese dokumentiert werden.

//Da mou nu a wengl was hi !!

# Testvorgehen

## Funktionale Tests

### Grundtests (Smoke Tests)

*<Beschreiben Sie Tests, mit denen offensichtliche Fehler schnell gefunden werden können, z.B. Tests, die vor jedem Commit in das Repository durchzuführen sind (ggf. Verweis auf externes Dokument).>*

### Modul- und Unittests

*<Wie (mit welchen Werkzeugen, in welcher Umgebung, in wessen Verantwortung) sollen Modul- und Unittests durchgeführt werden? Wie erfolgt die Dokumentation der Testergebnisse? Gibt es Vorgaben zur Testüberdeckung?*

*Hier werden nicht einzelne Tests beschrieben, sondern nur die Vorgehensweise!>*

### Integrationstests

*<Sind gesonderte Tests notwendig, während / nachdem verschiedene Module zusammengeführt werden?>*

### System Acceptance Test

*<Auflistung der Tests auf System- / Nutzerebene, die sich an den Use Cases orientieren. Ggf. Verweis auf separates Dokument. Beschreibung, wie / von wem diese Tests durchzuführen sind.>*

## Bedienbarkeit und Nutzerinterface (Usability)

## Datenschutz, Datensicherheit (Security)

## Leistungsanforderungen (Performance)

## Zuverlässigkeit

## Schnittstellen

## Wartung und Servicefunktionen

## Installation

## Internationalisierung / Lokalisierung

## Testautomatisierung

*<Welche Tests werden mit welchen Werkzeugen automatisiert? Wer erstellt / prüft die automatisierten Tests?>*

## Verfolgbarkeit (Traceability)

*<Wie wird sichergestellt, dass alle erfassten Anforderungen verifiziert werden?>*

# Übersicht der Testpläne

*<Hier werden alle Dokumente, in denen Testfälle beschrieben werden, identifizierbar erfasst.>*

# Freigabe von Testergebnissen

*<Wer entscheidet auf Grund welcher Kriterien darüber, ob Testergebnisse akzeptiert werden. (z.B.: Klassifikationsschema minor – moderate – major für Fehler. Für Alpha-Release werden moderate und minor Fehler, für Final Release nur minor Fehler akzeptiert.>*