- Systemanalyse -

Lastenheft für „Geheime Kommunikation mit Quantenkryptographie - eine Lernsoftware zum Verstehen der Grundlagen“

Version: 1.0

|  |  |  |
| --- | --- | --- |
| Projektbezeichnung | QuaKrypto | |
| Projektleiter | Simon Schisslbauer | |
| Verantwortlich | Team | |
| Erstellt am | 30.03.2023 | |
| Zuletzt geändert | 08.04.2023 11:44 | |
| Bearbeitungszustand | X | in Bearbeitung |
|  | vorgelegt |
|  | fertig gestellt |
| Dokumentablage |  | |

Änderungsverzeichnis

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Änderung | | | Geänderte Kapitel | Beschreibung der Änderung | Autor | Zustand |
| Nr. | Datum | Version |
| 1 |  | 1.0 | Alle | Initiale Produkterstellung |  |  |

Prüfverzeichnis

Die folgende Tabelle zeigt einen Überblick über alle Prüfungen – sowohl Eigenprüfungen wie auch Prüfungen durch eigenständige Qualitätssicherung – des vorliegenden Dokumentes.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Datum | Geprüfte Version | Anmerkungen | Prüfer | Neuer Produktzustand |
|  |  |  |  |  |

Inhalt

[1 Einleitung 4](#_Toc34239361)

[2 Ausgangssituation und Zielsetzung 4](#_Toc34239362)

[3 Funktionale Anforderungen 5](#_Toc34239363)

[3.1 Use-Case Übersicht 5](#_Toc34239364)

[3.2 Use-Beschreibungen 5](#_Toc34239365)

[3.3 (Sonstige) Funktionalität 7](#_Toc34239366)

[3.4 Modell des Problembereichs (Konzeptionelles Datenmodell) 7](#_Toc34239367)

[4 Nicht-Funktionale Anforderungen 7](#_Toc34239368)

[4.1 Benutzbarkeit (Usability) 7](#_Toc34239369)

[4.2 Zuverlässigkeit (Reliability) 8](#_Toc34239370)

[4.3 Leistung (Performance) 8](#_Toc34239371)

[4.4 Unterstützbarkeit (Supportability) 9](#_Toc34239372)

[4.5 Sonstige Einschränkungen 9](#_Toc34239373)

[5 Risikoakzeptanz 10](#_Toc34239374)

[6 Skizze der Gesamtsystemarchitektur 10](#_Toc34239375)

[7 Lieferumfang 11](#_Toc34239376)

[8 Abnahmekriterien 11](#_Toc34239377)

[9 Glossar 11](#_Toc34239378)

[10 Abkürzungsverzeichnis 12](#_Toc34239379)

[11 Literaturverzeichnis 12](#_Toc34239380)

[12 Abbildungsverzeichnis 12](#_Toc34239381)

# Einleitung

Dieses Dokument enthält alle an das zu entwickelnde System gestellten Anforderungen. Die Gliederung orientiert sich am Aufbau des V-Modell-XT®[[1]](#footnote-1)-Produkts „Anforderungen (Lastenheft)“, ist jedoch zur Verwendung für die Veranstaltung **„Software-Projekte“** in Informatik-Curricula der **OTH-Amberg-Weiden** angepasst worden (und nicht konform zum V-Modell-XT): Teilnehmer dieser Veranstaltung erhalten von ihrem „Auftraggeber“ lediglich einen Überblick über das gewünschte System, was ungefähr dem Thema „Ausgangssituation und Zielsetzung“ in diesem Dokument entspricht; die Anforderungen müssen die Teilnehmer dann in enger Abstimmung mit ihrem „Auftraggeber“ selbst erarbeiten und in diesem Dokument niederlegen. Dadurch sollen sie Gelegenheit erhalten, auch Tätigkeiten der System-Analyse intensiver zu üben. Die „Auftraggeberseite“ liefert also nicht – wie im V-Modell-XT vorgesehen - das komplette Lastenheft, aus dem die „Auftragnehmerseite“ ein separates Pflichtenheft ableitet; stattdessen wird das hier vorliegende Dokument vom studentischen Entwicklerteam zur Dokumentation der Analyse-Ergebnisse erstellt und zugleich als Ersatz für die im V-Modell-XT vorgesehenen Dokumente Lasten- und Pflichtenheft verwendet.

Kern dieses Dokuments sind die funktionalen und nicht-funktionalen Anforderungen an das System, sowie eine Skizze des Gesamtsystementwurfs. Der Entwurf berücksichtigt die zukünftige Umgebung und Infrastruktur, in der das System später betrieben wird, und gibt Richtlinien für Technologieentscheidungen. Ebenfalls Teil der Anforderungen ist die Festlegung von Lieferbedingungen und Abnahmekriterien.

Die funktionalen und nicht-funktionalen Anforderungen dienen nicht nur als Vorgaben für die Entwicklung, sondern sind zusätzlich Grundlage der Anforderungsverfolgung und des Änderungsmanagements. Die Anforderungen sollten so aufbereitet sein, dass die Verfolgbarkeit (Traceability) sowie ein geeignetes Änderungsmanagement für den gesamten Lebenszyklus eines Systems möglich sind.

Im Allgemeinen sollten keine technischen Lösungen vorgegeben werden, um Architekten und Entwickler bei der Suche nach optimalen technischen Lösungen nicht einzuschränken.

# Ausgangssituation und Zielsetzung

…hier Ihren Text einfügen…

# Funktionale Anforderungen

## Use-Case Übersicht



## Use-Beschreibungen

|  |  |  |
| --- | --- | --- |
| **Definition eines Übungsszenarios** | | |
| **Kennung** | | UC-1 |
| **Priorität** | | Hoch |
| **Kurzbeschreibung:** | | |
| Eine Benutzergruppe wählt:   * das Protokoll, das sie durch das System erlernen will * ob das Übungsszenario lokal oder netzwerkbasiert durchgeführt werden soll * die gewünschte Variante * die gewünschte(n) Phase(n) * den gewünschten Schwierigkeitsgrad   im Übungsszenariomenü aus. | | |
| **Vorbedingung(en):** | | |
| * Der Benutzer befindet sich im Übungsszenariomenü | | |
| **Nachbedingung(en):** | | |
| * Eine Lobby wurde erstellt | | |
| **Normaler Ablauf:** | | |
|  | 1. Der Anwendungsfall beginnt, wenn die Benutzergruppe das Protokoll auswählt, welches sie durch das System erlernen will 2. Die Benutzergruppe wählt aus, ob das Übungsszenario lokal oder netzwerkbasiert bearbeitet werden soll 3. Die Benutzergruppe wählt zunächst die gewünschte Variante aus 4. Die Benutzergruppe wählt die gewünschte(n) Phase(n) aus 5. Die Benutzergruppe wählt den gewünschten Schwierigkeitsgrad aus 6. Die Benutzergruppe bestätigt ihre Eingaben 7. Das System prüft, ob die Eingaben der Benutzergruppe gültig sind 8. Das System übernimmt die Eingabedaten und erstellt eine Lobby   **Ende**. | |
| **Ablauf-Varianten:** | | |
| 7a | Die Eingabe(n) der Benutzergruppe ist ungültig | |
|  | 1. Das System weist die Benutzergruppe auf die ungültige(n) Eingabe(n) hin   **Ende.** | |
| 8a | Die Benutzergruppe hat die netzwerkbasierte Bearbeitung für das Übungsszenario gewählt | |
|  | 1. Das System legt eine netzwerkbasierte Lobby an.   **Ende.** | |
| 8b | Die Benutzergruppe hat die lokale Bearbeitung für das Übungsszenario gewählt | |
|  | 1. Das System legt eine lokale Lobby an.   **Ende.** | |
| **Spezielle Anforderungen:** | | |
|  | | |
| **Zu klärende Punkte:** | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **Vorhandenem Übungsszenario beitreten** | | |
| **Kennung** | | UC-2 |
| **Priorität** | | Hoch |
| **Kurzbeschreibung:** | | |
| Die Benutzergruppe tritt einem vorhandenen Übungsszenario bei.  Für ein lokales Übungsszenario legt sie   * ihre Rolle * ihr Alias * ihr Passwort   fest.  Für ein netzwerkbasiertes Übungsszenario legt sie   * die gewünschte Lobby * ihre Rolle * ihr Alias * ihr Passwort   fest.  Jede im Übungsszenario vorhandene Rolle (Alice, Bob, Eve) muss von genau einer Benutzergruppe repräsentiert werden. | | |
| **Vorbedingung(en):** | | |
| Für ein lokales Übungsszenario:   * Eine lokale Lobby wurde erstellt * Das System hat das Lobbymenü erstellt   Für ein netzwerkbasiertes Übungsszenario:   * Im selben Netzwerk wurde mindestens eine netzwerkbasierte Lobby erstellt * Die Benutzergruppe befindet sich im Lobbyauswahlmenü | | |
| **Nachbedingung(en):** | | |
| * Die Benutzergruppe ist der Lobby des Übungsszenarios beigetreten | | |
| **Normaler Ablauf:** | | |
|  | 1. Der Anwendungsfall beginnt, wenn die Benutzergruppe der Lobby eines Übungsszenarios beitritt 2. Das System prüft, ob die Eingaben gültig sind   **Ende**. | |
| **Ablauf-Varianten:** | | |
| 1a | Die Benutzergruppe tritt einer lokalen Lobby bei | |
|  | 1. Die Benutzergruppe wählt eine Rolle aus 2. Die Benutzergruppe legt ihr Alias fest 3. Die Benutzergruppe legt ihr Passwort fest   **Rückkehr nach**: 2 | |
| 1b | Die Benutzergruppe wählt im Lobbyauswahlmenü die gewünschte Lobby aus | |
|  | 1. Die Benutzergruppe wählt eine Rolle aus 2. Die Benutzergruppe legt ihr Alias fest 3. Die Benutzergruppe legt ihr Passwort fest   **Rückkehr nach**: 2 | |
| 1ba | Die gewünschte Lobby ist nicht mehr verfügbar | |
|  | 1. Das System weist die Benutzergruppe darauf hin, dass die angegebene Lobby nicht gefunden werden konnte   **Ende.** | |
| **Spezielle Anforderungen:** | | |
|  | | |
| **Zu klärende Punkte:** | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **Bearbeitung eines Übungsszenarios** | | |
| **Kennung** | | UC-3 |
| **Priorität** | | Hoch |
| **Kurzbeschreibung:** | | |
| Eine oder mehrere Benutzergruppen durchlaufen die ihnen zugeordneten Handlungsschritte. | | |
| **Vorbedingung(en):** | | |
| * Eine Lobby wurde erstellt * Jede in der Lobby vorhandene Rolle ist von genau einer Benutzergruppe belegt | | |
| **Nachbedingung(en):** | | |
| * Eine Aufzeichnung der Handlungsschritte der Bearbeitung des Übungsszenarios wurde erstellt | | |
| **Normaler Ablauf:** | | |
|  | 1. Der Anwendungsfall beginnt, wenn der Lobbyhost das Übungsszenario startet 2. Das System initialisiert das Übungsszenario (inkl. Aufzeichnung) 3. Der Rolle wird des Zuges entsprechende Handlungsschritte angezeigt 4. Die Rolle führt gewünschte Handlungsschritte aus 5. Das System speichert die durchgeführten Handlungsschritte in einer Aufzeichnung ab 6. Die Rolle bestätigt das Ende ihres Zuges durch entsprechenden Handlungsschritt 7. Das System gibt die Steuerung für die nächste Rolle frei 8. Dieser Vorgang (Schritte 3 bis 8) wird so lange wiederholt, bis die aktuelle Phase abgeschlossen wurde 9. Dieser Vorgang (Schritte 3 bis 9) wird so lange wiederholt, bis die letzte durch den Lobbyhost ausgewählte Phase abgeschlossen wurde   **Ende**. | |
| **Ablauf-Varianten:** | | |
| 6a | Es handelt sich um ein netzwerkbasiertes Übungsszenario | |
|  | 1. Die Rolle bestätigt das Ende ihres Zuges durch eine entsprechende Aktion   **Rückkehr nach**: 7 | |
| 6a1a | Mindestens zwei Rollen benutzen den gleichen Rechner | |
|  | 1. Die Rolle bestätigt das Ende ihres Zuges durch zusätzlich Sperren durch ihr jeweiliges Passwort   **Rückkehr nach**: 7 | |
| **Spezielle Anforderungen:** | | |
|  | | |
| **Zu klärende Punkte:** | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **Aufruf des Wikis** | | |
| **Kennung** | | UC-4 |
| **Priorität** | | Hoch |
| **Kurzbeschreibung:** | | |
| Um sein Wissen und seine Kenntnisse zu vertiefen, soll die Benutzergruppe ein Wiki mit Informationen über das Protokoll aufrufen können. | | |
| **Vorbedingung(en):** | | |
| * Es existiert ein Wiki | | |
| **Nachbedingung(en):** | | |
| * Das Wiki ist geöffnet | | |
| **Normaler Ablauf:** | | |
|  | 1. Der Anwendungsfall beginnt, wenn die Benutzergruppe die Aktion zum Öffnen des Wikis ausführt. 2. Das System zeigt der Benutzergruppe das Wiki mit allen Wiki-Einträgen an. | |
| **Ablauf-Varianten:** | | |
|  |  | |
| **Spezielle Anforderungen:** | | |
|  | | |
| **Zu klärende Punkte:** | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **Bearbeitung des Wikis** | | |
| **Kennung** | | UC-5 |
| **Priorität** | | Hoch |
| **Kurzbeschreibung:** | | |
| Die Benutzergruppe soll das vorhandene Wiki editieren können. | | |
| **Vorbedingung(en):** | | |
| * Das Wiki ist geöffnet | | |
| **Nachbedingung(en):** | | |
| * Die Wiki-Einträge wurden gemäß der Benutzereingaben verändert * Die Änderungen am Wiki wurden abgespeichert | | |
| **Normaler Ablauf:** | | |
|  | 1. Der Anwendungsfall beginnt, wenn die Benutzergruppe die Aktion zum Editieren eines Wiki-Eintrags ausführt. 2. Die Benutzergruppe wählt aus, ob sie einen Wiki-Eintrag bearbeitet, löschen oder hinzufügen will. 3. Das System speichert die von der Benutzergruppe getätigten Änderungen am Wiki.   **Ende**. | |
| **Ablauf-Varianten:** | | |
| 2a | Die Benutzergruppe will einen Wiki-Eintrag bearbeiten. | |
|  | 1. Die Benutzergruppe wählt den zu bearbeitenden Wiki-Eintrag aus. 2. Das System lädt den zu bearbeitenden Wiki-Eintrag. 3. Die Benutzergruppe tätigt ihre Änderungen an dem gewählten Wiki-Eintrag. 4. Die Benutzergruppe bestätigt die getätigten Änderungen am gewählten Wiki-Eintrag.   **Rückkehr nach**: 3 | |
| 2a4a | Die Benutzergruppe verwirft die Änderungen. | |
|  | **Rückkehr nach**: 2 | |
| 2b | Die Benutzergruppe will einen Wiki-Eintrag löschen. | |
|  | 1. Die Benutzergruppe wählt den zu löschenden Wiki-Eintrag aus. 2. Die Benutzergruppe bestätigt die Löschung des gewählten Wiki-Eintrags.   **Rückkehr nach**: 3 | |
| 2b3a | Die Benutzergruppe verwirft die Änderungen. | |
|  | **Rückkehr nach**: 2 | |
| 2c | Die Benutzergruppe will einen Wiki-Eintrag hinzufügen. | |
|  | 1. Das System lädt einen neuen Wiki-Eintrag. 2. Die Benutzergruppe fügt Informationen im neuen Wiki-Eintrag ein. 3. Die Benutzergruppe bestätigt die Erstellung des neuen Wiki-Eintrags.   **Rückkehr nach**: 3 | |
| 2c3a | Die Benutzergruppe verwirft die Änderungen. | |
|  | **Rückkehr nach**: 2 | |
| 2d | Die Benutzergruppe bricht die Bearbeitung des Wikis ab. | |
|  | 1. Die Benutzergruppe bestätigt den Abbruch des Vorgangs.   **Ende** | |
| **Spezielle Anforderungen:** | | |
|  | | |
| **Zu klärende Punkte:** | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **Anzeigen der Aufzeichnung** | | |
| **Kennung** | | UC-6 |
| **Priorität** | | Gering |
| **Kurzbeschreibung:** | | |
| Das System soll die Aufzeichnung des Übungsszenarios anzeigen. | | |
| **Vorbedingung(en):** | | |
| * Die Bearbeitung eines Übungsszenarios ist vollständig abgeschlossen (Use-Case 3 „Bearbeitung eines Übungsszenarios“) * Das System hat die Handlungsschritte der Bearbeitung des Übungsszenarios aufgezeichnet | | |
| **Nachbedingung(en):** | | |
| * Der Benutzergruppe wurden die aufgezeichneten Handlungsschritte der Bearbeitung des Übungsszenarios angezeigt. | | |
| **Normaler Ablauf:** | | |
|  | 1. Der Anwendungsfall beginnt, wenn das Übungsszenario vollständig abgeschlossen ist. 2. Das System zeigt dem Benutzer die Aufzeichnung an. 3. Die Benutzergruppe schließt die Aufzeichnung.   **Ende**. | |
| **Spezielle Anforderungen:** | | |
|  | | |
| **Zu klärende Punkte:** | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **Nachricht senden** | | |
| **Kennung** | | UC-7 |
| **Priorität** | | Hoch |
| **Kurzbeschreibung:** | | |
| Die Rollen sollen Nachrichten in Form von Photonen oder Bitfolgen senden können. | | |
| **Vorbedingung(en):** | | |
| * Es wird ein Übungsszenario bearbeitet * Die Rolle, welche eine Nachricht senden möchte, ist am Zug | | |
| **Nachbedingung(en):** | | |
| * Die Nachricht wurde gesendet und ist auf dem entsprechenden Übertragungskanal gespeichert und kann ausgelesen werden | | |
| **Normaler Ablauf:** | | |
|  | 1. Der Anwendungsfall beginnt, wenn die Rolle den Handlungsschritt zum Senden einer Nachricht ausführt. 2. Die Rolle wählt die zu sendende Information aus. 3. Das System sendet die gewählte Information in Form einer Nachricht.   **Ende**. | |
| **Ablauf-Varianten:** | | |
| 2a | Die zu sendenden Informationen sind Photonen. | |
|  | 1. Die Rolle wählt vorhandene Polarisationsschemata als Emitter für die Photonenfolge aus 2. Die Rolle wählt die Bitfolge, aus welcher die Photonen generiert werden sollen, aus 3. Das System berechnet anhand dieser beiden Eingaben die zu sendende Photonenfolge   **Rückkehr nach**: 3 | |
| 2b | Die zu sendende Information sind Polarisationsschemata. | |
|  | 1. Die Rolle wählt die Art der Generierung der Polarisationsschemata aus | |
| 2b1a | Die Rolle möchte die Polarisationsschemata selbst generieren. | |
|  | 1. Die Rolle gibt die zu sendenden Polarisationsschemata an   **Rückkehr nach**: 3 | |
| 2b1b | Die Rolle möchte, dass die Polarisationsschemata durch das System generiert werden. | |
|  | 1. Die Rolle gibt die Länge der zu generierenden Polarisationsschemata an 2. Das System generiert die zu sendenden Polarisationsschemata   **Rückkehr nach**: 3 | |
| 2c | Die zu sendende Information ist eine Bitfolge. | |
|  | 1. Die Rolle wählt die Art der Bitfolgengenerierung aus | |
| 2c1a | Die Rolle möchte die Bitfolge selbst generieren. | |
|  | 1. Die Rolle gibt die zu sendenden Bitfolge an 2. Das System verarbeitet die Bitfolge   **Rückkehr nach**: 3 | |
| 2c1b | Die Rolle möchte, dass die Bitfolge durch das System generiert werden. | |
|  | 1. Die Rolle gibt die Länge der zu generierenden Bitfolge an 2. Das System generiert die zu sendenden Bitfolge 3. Das System verarbeitet die Bitfolge   **Rückkehr nach**: 3 | |
| **Spezielle Anforderungen:** | | |
|  | | |
| **Zu klärende Punkte:** | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **Nachricht empfangen** | | |
| **Kennung** | | UC-6 |
| **Priorität** | | Hoch |
| **Kurzbeschreibung:** | | |
| Die Rollen sollen Nachrichten in Form von Photonen oder Bitfolgen empfangen können. | | |
| **Vorbedingung(en):** | | |
| * Es wird ein Übungsszenario bearbeitet * Die Rolle, welche eine Nachricht empfangen möchte, ist am Zug | | |
| **Nachbedingung(en):** | | |
| * Die Nachricht wurde empfangen und wird angezeigt * Die empfangene Nachricht wurde aus dem Übertragungskanal entfernt | | |
| **Normaler Ablauf:** | | |
|  | 1. Der Anwendungsfall beginnt, wenn die Rolle den Handlungsschritt zum Empfangen einer Nachricht ausführt. 2. Das System zeigt an, auf welchem Übertragungskanal Informationen verfügbar sind. 3. Die Rolle wählt aus, von welchem Übertragungskanal sie Informationen empfangen will. 4. Die Nachricht wird gemäß der Benutzereingaben ausgelesen   **Ende**. | |
| **Ablauf-Varianten:** | | |
| 3a | Die Rolle möchte Informationen von dem Photonenübertragungskanal empfangen | |
|  | 1. Das System zeigt an, wie viele Photonen empfangen werden können 2. Es werden Polarisationsschemata zum Auslesen der Photonen generiert.   **Rückkehr nach**: 4 | |
| 3a2a | Die Rolle möchte die Polarisationsschemata selbst generieren. | |
|  | 1. Die Rolle gibt die Polarisationsschemata zum Auslesen der Photonen an   **Rückkehr nach**: 4 | |
| 3a2b | Die Rolle möchte, dass die Polarisationsschemata durch das System generiert werden. | |
|  | 1. Die Rolle gibt die Länge der zu generierenden Polarisationsschemata an 2. Das System generiert die Polarisationsschemata zum Auslesen der Photonen   **Rückkehr nach**: 4 | |
| 3b | Die Rolle möchte eine Bitfolge vom Bitübertragungskanal empfangen | |
|  | 1. Das System zeigt an, welche Bitfolge empfangen werden kann und wie lang diese ist 2. Die Rolle wählt aus, wie die Bitfolge interpretiert werden 3. Das System interpretiert die Bitfolge und zeigt die interpretierte Bitfolge der Rolle an 4. Die angezeigte Information wird abgespeichert   **Rückkehr nach**: 4 | |
| **Spezielle Anforderungen:** | | |
|  | | |
| **Zu klärende Punkte:** | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **Nachricht abhören** | | |
| **Kennung** | | UC-6 |
| **Priorität** | | Hoch |
| **Kurzbeschreibung:** | | |
| Um dem Benutzer nach Vollendung eines Übungsszenarios Feedback über sein Protokollverständnis geben zu können, soll das System die Durchführung des Übungsszenarios auswerten. | | |
| **Vorbedingung(en):** | | |
| * Der Benutzer hat die Bearbeitung eines Übungsszenarios vollständig abgeschlossen (Use-Case 3 „Bearbeitung eines Übungsszenarios“) * Das System hat das auszuwertende Übungsszenario protokolliert | | |
| **Nachbedingung(en):** | | |
| * Der Benutzer hat Feedback zu seinen während der Bearbeitung des Übungsszenarios getätigten Aktionen erhalten. | | |
| **Normaler Ablauf:** | | |
|  | 1. Der Anwendungsfall beginnt, wenn der Benutzer die Aktion zur Anzeige der Auswertung des Übungsszenarios ausführt. 2. Das System zeigt dem Benutzer die Auswertung des Übungsszenarios an. 3. Der Benutzer schließt die Auswertung des Übungsszenarios.   **Ende**. | |
| **Ablauf-Varianten:** | | |
|  |  | |
|  | 1. … 2. …   **Rückkehr nach**: | |
| **Spezielle Anforderungen:** | | |
|  | | |
| **Zu klärende Punkte:** | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| **One-Time-Pad vereinbaren** | | |
| **Kennung** | | UC-6 |
| **Priorität** | | Hoch |
| **Kurzbeschreibung:** | | |
| Um dem Benutzer nach Vollendung eines Übungsszenarios Feedback über sein Protokollverständnis geben zu können, soll das System die Durchführung des Übungsszenarios auswerten. | | |
| **Vorbedingung(en):** | | |
| * Der Benutzer hat die Bearbeitung eines Übungsszenarios vollständig abgeschlossen (Use-Case 3 „Bearbeitung eines Übungsszenarios“) * Das System hat das auszuwertende Übungsszenario protokolliert | | |
| **Nachbedingung(en):** | | |
| * Der Benutzer hat Feedback zu seinen während der Bearbeitung des Übungsszenarios getätigten Aktionen erhalten. | | |
| **Normaler Ablauf:** | | |
|  | 1. Der Anwendungsfall beginnt, wenn der Benutzer die Aktion zur Anzeige der Auswertung des Übungsszenarios ausführt. 2. Das System zeigt dem Benutzer die Auswertung des Übungsszenarios an. 3. Der Benutzer schließt die Auswertung des Übungsszenarios.   **Ende**. | |
| **Ablauf-Varianten:** | | |
|  |  | |
|  | 1. … 2. …   **Rückkehr nach**: | |
| **Spezielle Anforderungen:** | | |
|  | | |
| **Zu klärende Punkte:** | | |
|  | | |

## (Sonstige) Funktionalität

Hier werden funktionale Anforderungen erfasst, die sich nicht vernünftig durch einen einzelnen Use-Case beschreiben lassen, weil sie z.B. „Querschnitt“-Funktionalität betreffen, die sich über viele Use-Cases erstreckt. Beispiele könnten sein:

|  |  |  |
| --- | --- | --- |
| **ID** | **Beschreibung** | **Querverweise** |
| FR-001 | Das System soll jede Fehlersituation dauerhaft protokollieren |  |
| FR-002 | Nutzung des Systems ist generell nur nach vorheriger Authentifizierung möglich |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Modell des Problembereichs (Konzeptionelles Datenmodell)

Hier wird ein Modell des Problembereichs („konzeptuelles Datenmodell“) in Form eines oder mehrerer UML-Klassendiagramme eingefügt. Das konzeptuelle Datenmodell wird oft nicht streng getrennt von den Anforderungen erarbeitet, weil ein genaues Verständnis z.B. der Use-Cases nicht ohne Verständnis der „Konzepte“ aus dem Problembereich möglich ist. Aus diesem Grund wird das konzeptionelle Modell an dieser Stelle zusammen mit den Use-Cases dokumentiert.

# Nicht-Funktionale Anforderungen

Nicht-funktionale Anforderungen beschreiben Anforderungen an das System, die nicht-fachlicher Natur sind, jedoch entscheidend zur Anwendbarkeit des Systems beitragen. Sie definieren beispielsweise Qualitätsanforderungen, Sicherheitsanforderungen oder Performanceanforderungen.

Nicht-funktionale Anforderungen definieren grundlegende Eigenschaften eines Systems, die im Architekturentwurf berücksichtigt werden müssen. Sie können zur Abschätzung der Entwicklungskosten herangezogen werden und sollten, soweit möglich, messbar beschrieben sein.

Zur einfachen Strukturierung der Anforderungen werden diejenigen Anforderungen, die nicht eindeutig zu den funktionalen Anforderungen gehören, den nicht-funktionalen Anforderungen zugeordnet.

Die hier verwendete Einteilung unterscheidet verschiedene Arten von Anforderungen nach dem „FURPS“-Schema (**F**unctionality, **U**sability, **R**eliability, **P**reformance, **S**upportability), das auf Hewlett-Packard zurückgeht. Das FURPS-Schema ist hier noch um die Kategorie „Sonstige Einschränkungen“ erweitert. Anforderungen zur Funktionalität sind bereits im Kapitel 3 dokumentiert, in diesem Kapitel folgen lediglich alle restlichen Anforderungen.

Bei Bedarf kann dieses Schema zur Einteilung der Anforderungen auch durch ein anderes Schema (z.B. nach DIN ISO) ersetzt werden – wichtig ist nur, dass bei der Erfassung der Anforderungen überhaupt ein erprobtes Schema verwendet wird. Dies soll eine bessere Übersicht bieten und dazu beitragen, dass keine Anforderungen vergessen werden.

## Benutzbarkeit (Usability)

Hier werden Anforderungen erfasst, die die Benutzbarkeit („usability“ = Benutzbarkeit / Benutzerfreundlichkeit / Gebrauchstauglichkeit) des Systems betreffen. Hierzu zählen insbesondere Anforderungen zur (Software/Hardware)-Ergonomie („human factors“).

|  |  |  |
| --- | --- | --- |
| **ID** | **Beschreibung** | **Querverweise** |
| UR-001 | Das System soll… |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Zuverlässigkeit (Reliability)

Hier werden Anforderungen erfasst, die die Zuverlässigkeit („reliability“ = Zuverlässigkeit) des Systems betreffen. Hierunter fallen insbesondere Anforderungen an die Wiederherstellbarkeit („recoverability“) und die Verfügbarkeit („availabilty“) des Systems. Die Wiederherstellbarkeit betrifft die Fähigkeit, bei Ausfall oder Störung das Leistungsniveau wieder zu erreichen und betroffene Daten wieder zu gewinnen. Eine Kenngröße in diesem Zusammenhang ist MTTR („**m**ean **t**ime **t**o **r**epair“), die die mittlere Zeit bis zur Wiederinbetriebnahme des Systems bei einem Ausfall angibt. Die Verfügbarkeit kann als Verhältnis zwischen der Zeit, in der das System funktionsfähig ist, und der Gesamtzeit angegeben werden:

V=MTBF / (MTBF+MTTR)

MTTR: mean time to repair (s.o.)

MTBF: **m**ean **t**ime **b**etween **f**ailures (also die mittlere Zeit zwischen zwei Ausfällen)

|  |  |  |
| --- | --- | --- |
| **ID** | **Beschreibung** | **Querverweise** |
| RR-001 | Das System soll… |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Leistung (Performance)

Hierunter fallen Anforderungen an die Leistung („performance“) des Systems. Die Anforderungen beziehen sich insbesondere auf das Zeitverhalten (Ausführungsgeschwindigkeit, Antwortzeiten, Durchsatz) sowie auf das Verbrauchsverhalten (Anzahl der belegten Betriebsmittel und Dauer der Betriebsmittelbelegung).

|  |  |  |
| --- | --- | --- |
| **ID** | **Beschreibung** | **Querverweise** |
| PR-001 | Das System soll… |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Unterstützbarkeit (Supportability)

Hierunter fallen Anforderungen, die Bereiche wie Anpassbarkeit („adaptability“), Testbarkeit („testability“), Wartbarkeit („maintainability“), Erweiterbarkeit („extensibility“), Lokalisierbarkeit („localizability“ = Anpassbarkeit an verschiedene Sprach- und Kulturräume) betreffen.

|  |  |  |
| --- | --- | --- |
| **ID** | **Beschreibung** | **Querverweise** |
| SR-001 | Das System soll… |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Sonstige Einschränkungen

In manchen Fällen können von vorneherein Einschränkungen („constraints“) für Entwurf, Implementierung, Schnittstellen und Hardware des geplanten Systems bestehen, die ebenfalls als Anforderungen zu berücksichtigen sind und das bisherige „FURPS“-Schema zu „FURPS+“ (vgl. [Lar], S. 88) erweitern.

### Schnittstellen

### Implementierung

### Entwurf

# Risikoakzeptanz

Für sicherheitskritische Systeme werden in diesem Thema Vorgaben für die Behandlung der Systemsicherheit festgelegt. Es wird aufgezeigt, welche Risiken im Rahmen des Systembetriebs bestehen, welche Schäden, oder auch welche Klassen von Schäden, mit welcher Wahrscheinlichkeit auftreten können und inwieweit das Eintreten eines Schadensfalls toleriert wird bzw. nicht mehr akzeptabel ist.

Die Risikoakzeptanz für die identifizierten möglichen Schadensfälle wird beispielsweise in Form einer Risikoakzeptanzmatrix dokumentiert. Die Matrix ist eine Vorgabe des Auftraggebers, in der er festlegt, bei welcher Schadensklasse und welcher Eintrittswahrscheinlichkeit er welche Risikoklasse akzeptiert.

Bei vielen Aufgabenstellungen in der Veranstaltung „Software-Projekte“ kann dieses Thema weggelassen werden.

…hier Ihren Text einfügen…

# Skizze der Gesamtsystemarchitektur

Das reine Aufstellen von Anwenderanforderungen ohne Überlegungen zu möglichen Lösungsräumen birgt die große Gefahr, unrealistische Anwenderanforderungen zu definieren. Für die Einordnung, Systematisierung, Kategorisierung und auch Priorisierung von Anwenderanforderungen ist ein Koordinierungsrahmen hilfreich, um die Visualisierung der Anwenderanforderungen zu erleichtern.

Diese Aufgabe kann eine Gesamtsystemarchitektur leisten, die die Sichtweise des Anwenders repräsentiert und nicht die technische Sichtweise des Systemanalytikers beziehungsweise des Systemarchitekten. Das heißt, es ist eine funktionale Systemarchitektur mit Einbettung in die funktionalen Abläufe von Nachbarsystemen zu erstellen. Eine technische Systemarchitektur ist in dieser frühen Phase kaum möglich.

Des Weiteren sind die Besonderheiten der Einsatzumgebung des neuen Systems zu beschreiben, um vor allem die Anforderungen an die Systemsicherheit berücksichtigen zu können.

…hier Ihren Text einfügen…

# Lieferumfang

Die folgende Tabelle enthält alle Arbeitsergebnisse, die in der Veranstaltung „Software-Projekte“ zu dem vom Team zu liefernden „End-Produkt“ gehören – für die individuell von jedem Projektteilnehmer zu liefernden Ergebnisse lesen Sie bitte im Projektleitfaden bzw. im Projektkalender nach. Die Benotung erfolgt nicht nur auf Grundlage des lauffähigen Programms, sondern bezieht die Qualität der Analyse, des Entwurfs und des Systemtests mit ein.

|  |  |  |  |
| --- | --- | --- | --- |
| **Lfd. Nr.** | **Was?** | **Art des Dokuments** | **Bemerkungen** |
| **Ergebnis der System-Analyse** | | | |
| 1 | Das Dokument „Systemanalyse(XYZ)“ (also dieses Dokument) mit funktionalen, nicht-funktionalen Anforderungen und konzeptionellem Datenmodell. | * Siehe Vorlage. * Wird bei Projektbeginn mit einer Überblick gebenden Systembeschreibung an das Team ausgegeben. Das Dokument ist vom Team weiterzuführen und wieder abzugeben. | * Bitte auf Abgabetermin während des Semesters achten (s. Projektkalender). * Rechtzeitig vor Abgabe auf Qualitätssicherung achten (Review) |
| **Dokumentation des Systementwurfs** | | | |
| 2 | Das Dokument „Systementwurf(XYZ)“. | * Siehe Vorlage. | * Bitte auf Abgabetermin während des Semesters achten (s. Projektkalender) * Rechtzeitig vor Abgabe auf Qualitätssicherung achten (Review) |
| **Implementierung** | | | |
| 3 | Lauffähiger und getesteter Quellcode |  | Abgabe am Semesterende |
| **Test** | | | |
| 4 | Testspezifikation Systemtest | * Siehe Vorlage | * Endgültige Abgabe am Semesterende; zur Vorbereitung des Abnahmetests ist die Aufstellung der in den Abnahmetest einbezogenen Testfälle früher vorzulegen (Termin im Projektkalender) |

# Abnahmekriterien

In der Veranstaltung „Software-Projekte“ werden vom „Auftraggeber“ (in Absprache mit den Teilnehmern) rechtzeitig vor Semesterende Systemtestfälle ausgewählt, die das System dann am Tag der Abnahme ohne Beanstandung „überstehen“ muss.

# Glossar

|  |  |
| --- | --- |
| Begriff | Erklärung |
|  |  |

# Abkürzungsverzeichnis

|  |  |
| --- | --- |
| Abkürzung | Erklärung |
|  |  |

# Literaturverzeichnis

**[Lar]** Larman Craig, *Applying UML And Patterns. An Introduction to Object-Oriented Analysis And Design,* Prentice Hall, 2nd ed., 2002

# Abbildungsverzeichnis

1. V-Modell® ist eine geschützte Marke der Bundesrepublik Deutschland. [↑](#footnote-ref-1)