

Softwareentwicklungsplan von Klimaschrank

Gruppe04

Version: 01

Seite 1 / 6

INHALTSVERZEICHNIS

1	Identifikation	2
1.1	<i>Dokumentübersicht</i>	2
1.2	<i>Abkürzungen</i>	2
1.3	<i>Verweise</i>	2
1.3.1	Projektreferenzen	2
1.3.2	Standard and regulatory References	2
1.4	<i>Conventions</i>	2
2	Softwareentwicklungsaktivitäten	2
2.1	<i>Softwareentwicklungsprozess</i>	2
2.1.1	Übersicht der Prozessphasen	3
2.1.2	Technische Dokumentation	3
2.1.3	Lieferung	3
2.2	<i>Softwareentwicklungstool</i>	4
2.2.1	Arbeitsstationen	4
2.2.2	Anforderungsmanagement und Dokumentation	4
2.2.3	Software Design	4
2.2.4	Codierung und automatisierte Tests	4
2.2.5	Konfigurationsmanagement	4
2.3	<i>Regeln und Standards für die Softwareentwicklung</i>	4
3	Verantwortungsbereiche	5
3.1	<i>Aktivitäten und Verantwortlichkeiten</i>	5
4	Risikoabschätzung	5
4.1	<i>Risikoanalyse</i>	5
4.2	<i>Risikoplanung</i>	6

1 Identifizierung

1.1 Dokumentübersicht

Dieses Dokument enthält den Softwareentwicklungsplan für Software "Klimaschrank".

Dieses Dokument enthält den Softwareentwicklungsplan von KSS - Klimaschrank Software. Dieser Plan ist der Plan einer Desktop-Anwendung, die die Steuerung vom Computer aus ermöglicht. Dazu gehört die Software spezieller Schränke, die prüfen, ob die Objekte unter verschiedenen klimatischen Bedingungen gut funktionieren, indem diese klimatischen Bedingungen ohne Verschlechterung simuliert werden.

1.2 Abkürzungen

KSS: Klimaschrank Software

SDP: Software Development Plan

SRP: Software Requirements Specifications

STP: Software Test Plan

SDD: Software Detailed Design

STR: Software Test Report

SAR: Software Analysis Report

GUI: Graphical User Interface

RAT: Risk Analysis Table

1.3 Verweise

1.3.1 Projektreferenzen

#	Document Identifier	Document Title
		Hier werden die Referenzen eingetragen.

2 Softwareentwicklungsaktivitäten

Der Abschnitt listet und beschreibt die Entwicklungsaktivitäten des "Klimaprüfschrank" Softwareentwicklungsprojekts.

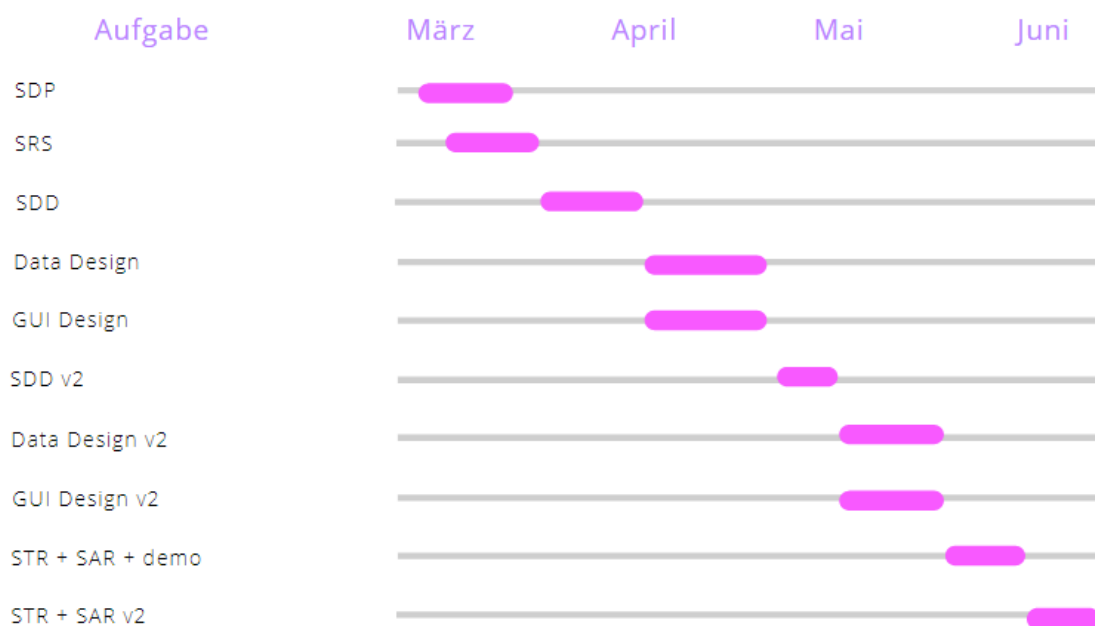
2.1 Softwareentwicklungsprozess

Der für das Projekt ausgewählte Softwareentwicklungsprozess ist das Wasserfall-Programmiermodell.

2.1.1 Übersicht der Prozessphasen

Der Softwareentwicklungsprozess für das Projekt besteht aus folgenden Phasen: Planung, Anforderungsanalyse, Design, Implementierung, Testen und Analysieren. Diese Phasen folgen nacheinander, wobei jede Phase unmittelbar nach Abschluss der vorherigen Phase beginnt. Das folgende Gantt-Diagramm zeigt das geplante Startdatum und die geplante Dauer für die Phasen.

Gantt Diagramm von Klimaschrank Project



2.1.2 Technische Dokumentation

Die folgende Dokumentation wird während der Softwareentwicklungsphase erstellt:

- Software specification: SRS
- Software detailed conception: SDD
- Software tests phases: STR
- Software analysis: SAR

2.1.3 Lieferung

Die folgenden Artikel werden am Ende des Prozesses geliefert:

- Technische Dokumentation,
- Benutzerdokumentation: Benutzerhandbuch, Administrationsverfahren und Installationsverfahren,
- Software und ihre Konfigurationsdateien

Softwareentwicklungsplan von Klimaschrank		
Gruppe04	Version: 01	Seite 4 / 6

2.2 Softwareentwicklungstools

2.2.1 Arbeitsstationen

LENOVO V330-15IKB
Windows 10 Pro x64
Intel(R) Core (TM) I5-8250U
8 GB Ram

Windows 10 Enterprise x64
Intel(R) Core(TM) i5-8600
16.0 GB RAM

MSI Gtx 1050 Gaming
Linux Ubuntu x64
Intel(R) Core(TM) i5-7300HQ
8Gb RAM

Dell Latitude E6440
Windows 10 x64
Intel(R) Core(TM) i7-4610M
8 Gb RAM

MSI Summit E15
Windows 10 x64
Intel(R) Core(TM) i7-1185G7
16GB Ram

2.2.2 Anforderungsmanagement und Dokumentation

Google Docs, Microsoft Excel, Microsoft Word

2.2.3 Software Design

Lucidchart (<https://www.lucidchart.com/>)

2.2.4 Codierung und automatisierte Tests

Java
MySQL Workbench 8.0
Testing Tool: JUnit

2.2.5 Konfigurationsmanagement

GitHub wird für die Verwaltung der Softwarekonfiguration und zur Verfolgung von Problemen iBezug auf die verwendet Software-Entwicklung. Zu diesem Zweck wird ein öffentliches Repository erstellt.

2.3 Regeln und Standards für die Softwareentwicklung

UML wird für die Dokumentation des Software-Designs verwendet.

[Java-Codierungskonventionen](#)

Softwareentwicklungsplan von Klimaschrank

Gruppe04

Version: 01

Seite 5 / 6

3 Verantwortungsbereiche

3.1 Aktivitäten und Verantwortlichkeiten

Activity	Responsibility
Projektmanagement	Yiğit Opan
Dokumentation	Yiğit Opan Zeynep Şenel
Codierung	Mehmet Leblebici Fırat Eray Erol Özlem Şahin
Erprobung	Yiğit Opan Zeynep Şenel Mehmet Leblebici Fırat Eray Erol Özlem Şahin
Database Design	Yiğit Opan
GUI Desing	Yiğit Opan Zeynep Şenel Mehmet Leblebici

4. Risikoabschätzung

4.1 Risikoanalyse

Risiko	Wahrscheinlichkeit	Auswirkungen
Der Zeitaufwand für die Entwicklung der Software wird unterschätzt	Mäßig	Ernst
Software-Tools können nicht integriert werden	Hoch	Tolerierbar
Die grosse von Software ist unterschätzt	Mäßig	Tolerierbar
Teammitglieder fehlen wegen der Krankheit	Hoch	Ernst
Der Preis der Software-Tools kann nicht angegeben werden	Niedrig	<u>Unerheblich</u>

Softwareentwicklungsplan von Klimaschrank

Gruppe04

Version: 01

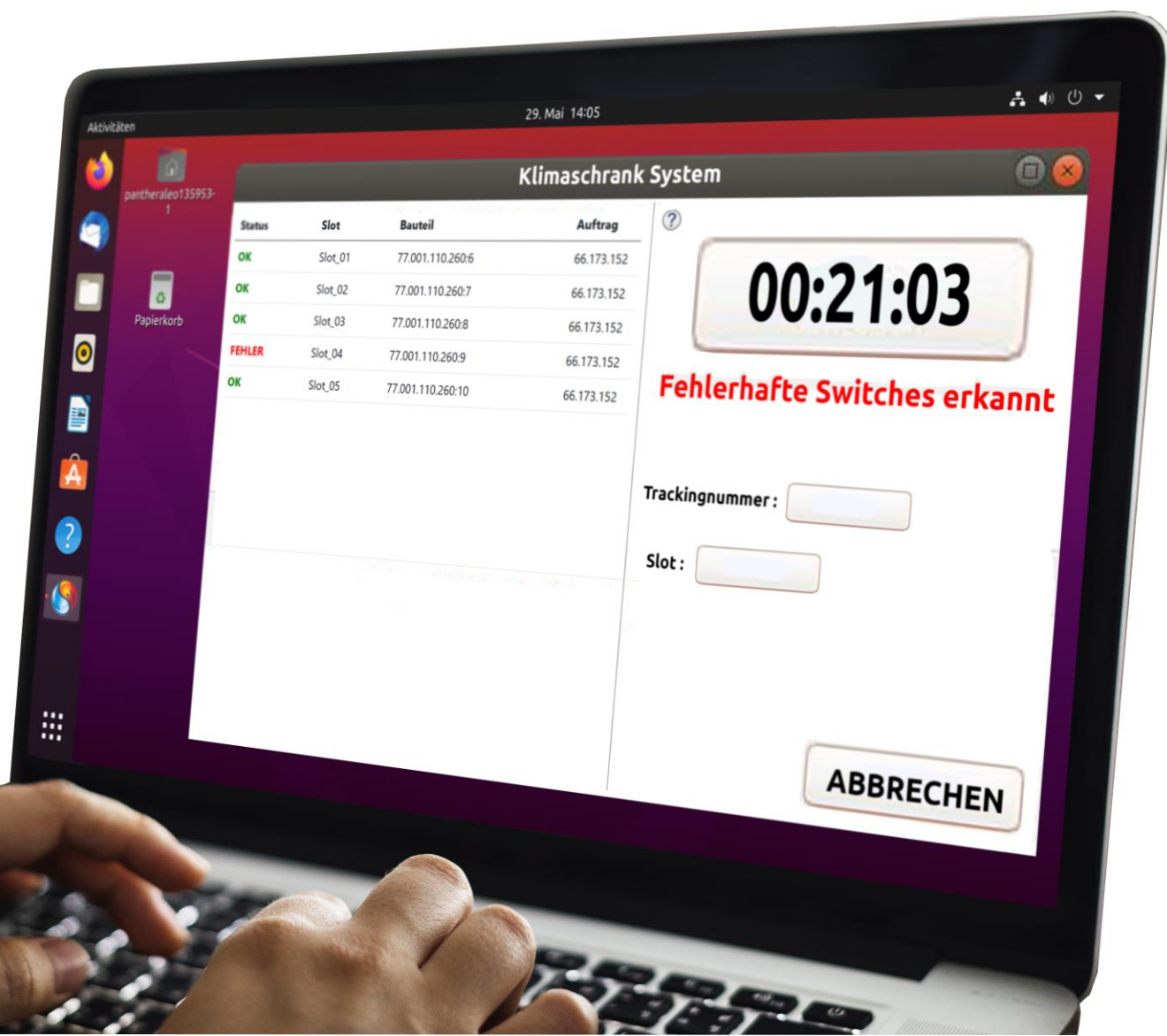
Seite 6 / 6

4.2. Riskikoplanung

<i>Risiko</i>	<i>Strategie</i>
Krankheit des Personals	Reorganisierung des Teams so, dass es mehr Überschneidungen bei der Arbeit gibt und die Mitarbeiter die Jobs des anderen verstehen.
Problem bei der Integration der Tools	Wendung zu anderen Softwaretools, die alle Mitglieder des Projekts mit minimalen Änderungen und Auswirkungen problemlos verwenden können.
Zeitliche Koordinierung	Steigerung des Engagements des Teams mit mehr Aufwand und mehr Arbeit, um den Zeitplan einzuhalten.
Softwaregröße	Entfernung von zusätzlichen Teilen des Projekts so, dass er nicht wirklich benötigt wird oder nicht unbedingt erforderlich ist.
Preis des Software-Tools	Suchung von anderen kostenlosen Tools, die dem aktuellen Entwicklungsplan entsprechen und in diesen integriert werden können.

INHALTSVERZEICHNIS

1 IDENTIFIZIERUNG	2
1.1 Dokumentübersicht	2
1.2 Systemübersicht	2
1.3 Abkürzungen	2
1.4 Verweise	2
1.5 Konventionen	2
 2 ANFORDERUNGEN	 3
2.1 Spezifikation der Benutzeranforderungen	3
2.2 Systemarchitektur	3
2.3 Anwendungsfälle und Verwendungsszenarien	4
2.4 Aktivitätsdiagramm	5
2.5 Spezifikation der Systemanforderungen	6



Softwareentwicklungsplan von Klimaschrank		
Gruppe04	Version: 02	Seite 2 / 8

1. IDENTIFIZIERUNG

1.1. Dokumentübersicht

Dieses Dokument enthält die Softwareanforderungsspezifikationen der KSS-Softwareentwicklung Projekt. Es beschreibt funktionale Anforderungen, Einschränkungen, Anwendungsfälle und Nutzungsszenarien.

1.2. Systemübersicht

Das KSS-Softwareprojekt wird für technologische Werkzeuge verwendet, die extremen Wetterbedingungen standhalten müssen. Es wird getestet, ob diese Werkzeuge bei der erforderlichen Temperatur funktionieren oder nicht und die Ergebnisse werden angezeigt.

1.3. Abkürzungen

KSS: Klimaschrank Software
SDP: Software Development Plan
SRP: Software Requirements Specifications
STP: Software Test Plan
SRS: Software Requirements Specifications
SDD: Software Detailed Design
STR: Software Test Report
SAR: Software Analysis Report
GUI: Graphical User Interface
RAT: Risk Analysis Table

1.4. Verweise

#	Dokumentenkennung	Dokumenttitel
[KSS]	KSS-SDP-1	SDP: Software Development Plan

1.5. Konventionen

Die in diesem Dokument aufgeführten Anforderungen sind nachfolgender Struktur aufgebaut:

SRS-KSS-001

Beschreibung der KSS-001-Anforderung

Jede Systemanforderungs-ID enthält die ID der entsprechenden Benutzeranforderung als Präfix.

SRS-KSS-001.1

Beschreibung der Systemanforderung KSS-001.1, die sich auf die Benutzeranforderung KSS-001 bezieht.

Softwareentwicklungsplan von Klimaschrank		
Gruppe04	Version: 02	Seite 3 / 8

2. ANFORDERUNGEN

2.1. Spezifikation der Benutzeranforderungen

- **SRS-KSS-001**

KSS-Software sollte in der Lage sein, Benutzer im System zu speichern, nachdem bestimmte Informationen angefordert hat.

- **SRS-KSS-002**

Benutzer sollen sich mit ihren Passwörtern beim KSS einloggen können.

- **SRS-KSS-003**

Benutzer sollten in der Lage sein, Testen zu führen.

- **SRS-KSS-004**

Benutzer sollten die Prüflingen aus dem System löschen und hinzufügen können.

- **SRS-KSS-005**

Das KSS sollte dem Benutzer die Testergebnisse anzeigen können.

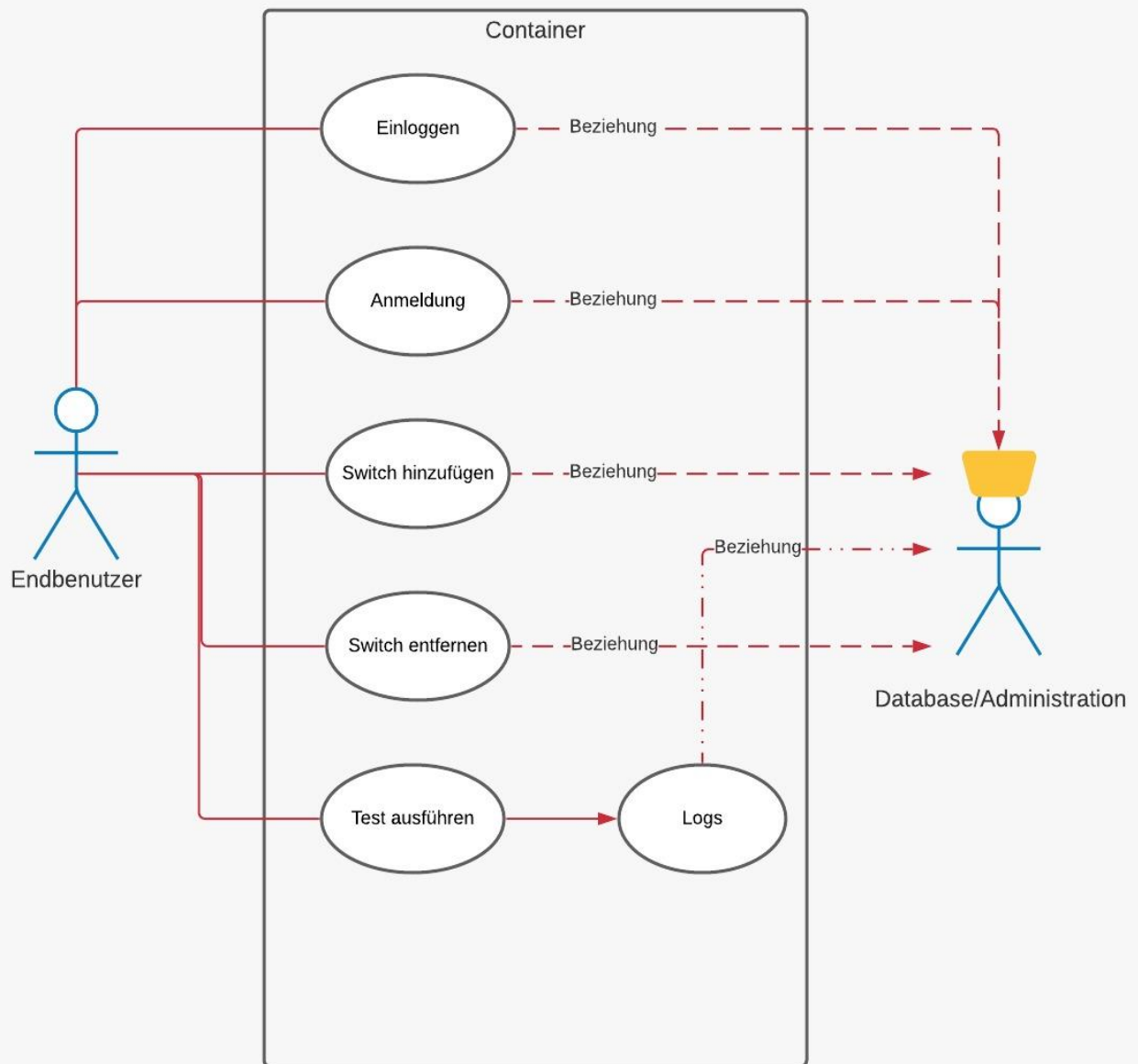
- **SRS-KSS-006**

Der Benutzer kann während des Haupttests in das KSS eingreifen.

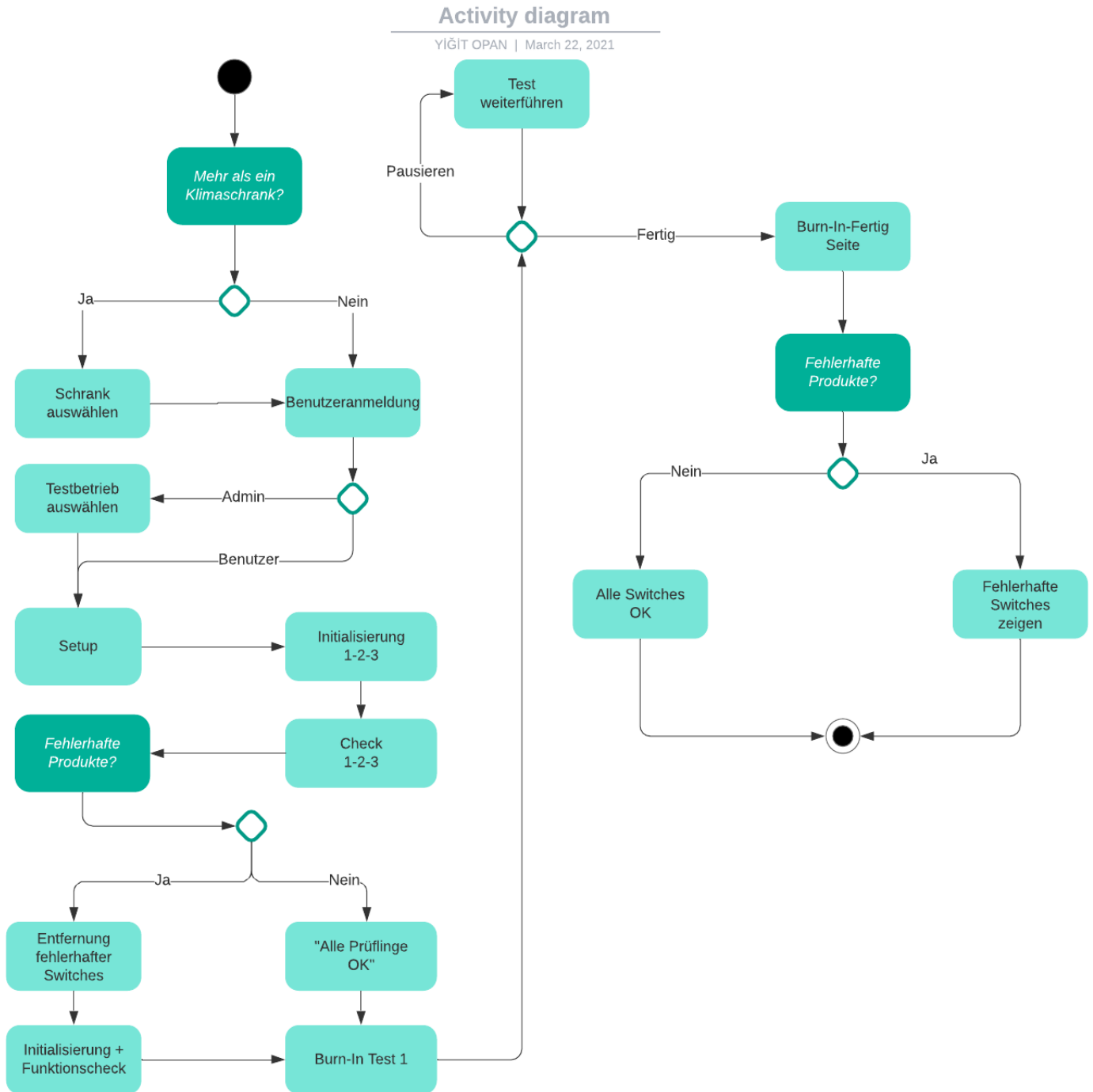
2.2 Systemarchitektur

Die wird während der detaillierten Entwurfsphase geteilt.

2.3 Anwendungsfälle und Verwendungsszenarien



2.4 Aktivitätsdiagramm



Softwareentwicklungsplan von Klimaschrank		
Gruppe04	Version: 02	Seite 6 / 8

2.5. Spezifikation der Systemanforderungen

▪ **SRS-KSS-001.1**

Wenn einen Benutzer gespeichert wird, sollte das KSS die folgenden Informationen:

- Name
- Nachname
- Passwort
- Personalnummer
- Rolle
- Status

vom Database abrufen.

▪ **SRS-KSS-001.2**

KSS sollte eine Benutzerkennung zum allen Benutzern geben.

▪ **SRS-KSS-002.1**

Wenn auf KSS eingeloggt wird, muss man seine Benutzererkennung und Passwort schreiben.

▪ **SRS-KSS-002.2**

Wenn der Benutzer das falsche Passwort eingibt, sollte das System eine Warnung ausgeben.

▪ **SRS-KSS-003.1**

Wenn der Benutzer ein Test führen möchte, sollte das KSS Verbindung mit dem Schrank stellen.

▪ **SRS-KSS-003.2**

Wenn die Verbindung gescheitert ist, sollte das System ein Feedback geben.

▪ **SRS-KSS-003.3**

KSS sollte maximal 20 Prüflinge zu setzen erlauben.

▪ **SRS-KSS-003.4**

Pre-Tests sollen von KSS gestartet werden.

Softwareentwicklungsplan von Klimaschrank		
Gruppe04	Version: 02	Seite 7 / 8

▪ **SRS-KSS-003.5**

KSS sollte die folgenden Informationen:

- Slot
- BauteilID
- Erfolgstatus
- Auftrag
- verstrichene Zeit

während des Pre-Tests darstellen.

▪ **SRS-KSS-003.6**

KSS sollte den Benutzer von Beginn des Haupttests nachfolgenden Informationen fragen:

- tempTarget
- minTarget
- TemperaturKurve

▪ **SRS-KSS-003.7**

KSS sollte dem Benutzer während des Haupttests die folgenden Informationen bereitstellen:

- istTemperatur
- sollTemperatur
- minTarget
- VerstricheneZeit

▪ **SRS-KSS-003.7**

Das KSS muss mit einem einzelnen Schrank verbunden sein und diese Informationen müssen in der Datenbank gefunden werden.

▪ **SRS-KSS-005.1**

Das KSS sollte die Haupttestergebnisse wie folgt speichern:

- Datum
- Uhrzeit
- Status (erfolgreich oder nicht)
- Test ID
- Die Person, die den Test durchführt

▪ **SRS-KSS-005.2**

Das KSS sollte wie folgt aufzeichnen, falls den Prüflingen während des Pretests und des Haupttests fehlgeschlagen sind:

- Testnummer
- Nummer der Person, die den Test durchführt
- Fehlerursache

Softwareentwicklungsplan von Klimaschrank		
Gruppe04	Version: 02	Seite 8 / 8

- **SRS-KSS-005.3**

Das KSS sollte nicht zulassen, dass die getestete Prüflingen erneut getestet werden.

- **SRS-KSS-005.4**

Das KSS sollte nach der Schrank-Temperatur fragen. Wenn die Schrank-Temperatur von der gewünschten Temperatur abweicht (mit 10% Toleranz), sollte der Schrank als defekt angesehen und der Benutzer informiert werden.

- **SRS-KSS-006.1**

KSS sollte während des Haupttests vom Benutzer abgebrochen und gestoppt werden können.

- **SRS-KSS-006.2**

KSS sollte nach dem Abbruch während des Haupttests wieder aufgenommen werden.