

SE I - Belegabgabe

Bauphysik

Fritzsche Felix, Lehmann Christian, Ullmann Max, Baburkin Yewgenij, Gieß
Christian, Grambole Lukas, Däbler Michael, Denis Klasowski

31. Januar 2020

1. Vision BauphysikSE1	1
1.1. Einführung	1
1.2. Positionierung	1
1.3. Stakeholder Beschreibungen	3
1.4. Produkt-/Lösungsüberblick	4
1.5. Zusätzliche Produktanforderungen	5
2. Use-Case Model BauphysikSE1	6
2.1. Identifizierte Use Cases	6
2.2. Use Case Diagramm	6
2.3. Ausgearbeitete Use Cases	7
2.4. Use Case: Reihenberechnung durchführen	7
2.5. Use Case: Plausibilität prüfen	12
2.6. Use Case: Daten drucken	12
3. BauphysikSE1 System-Wide Requirements Specification	17
3.1. Einführung	17
3.2. Systemweite funktionale Anforderungen	17
3.3. Qualitätsanforderungen für das Gesamtsystem	17
3.4. Zusätzliche Anforderungen	19
4. Glossar BauphysikSE1	20
4.1. Einführung	20
4.2. Begriffe	20
4.3. Akronyme & Abkürzungen	21
4.4. Physikalische Größen	21
4.5. Verzeichnis der Datenstrukturen	23
Projektdokumentation	24
5. Projektplan BauphysikSE1	25
5.1. Einführung	25
5.2. Projektorganisation	25
5.3. Praktiken und Bewertung	25
5.4. Meilesteine und Ziele	25
5.5. Deployment	27
5.6. Erkenntnisse (Lessons learned)	27
6. Risikoliste Bauphysik	28
7. Iterationsplan 1 BauphysikSE1	32
7.1. Meilensteine	32
7.2. Wesentliche Ziele	32
7.3. Aufgabenzuordnung	32
7.4. Probleme (optional)	33
7.5. Bewertungskriterien	34
7.6. Assessment	34
8. Iterationsplan 2 BauphysikSE1	35

8.1. Meilensteine	35
8.2. Wesentliche Ziele	35
8.3. Aufgabenzuordnung	35
8.4. Probleme (optional)	37
8.5. Bewertungskriterien	37
8.6. Assessment	37
Entwurfsdokumentation	39
9. Architecture Notebook Bauphysik	40
9.1. Zweck	40
9.2. Architekturziele und Philosophie	40
9.3. Annahmen und Abhängigkeiten	40
9.4. Architektur-relevante Anforderungen	40
9.5. Entscheidungen, Nebenbedingungen und Begründungen	40
9.6. Architekturmechanismen	41
9.7. Architektursichten (Views)	41
10. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]	42
10.1. Test Case ID: 001 - Berechnung Wärmewiderstand für zwei Schichten:	42
11. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]	43
11.1. Test Case ID: 002 - Berechnung Wärmewiderstand und Temperaturverlauf für zwei Schichten:	43
12. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]	45
12.1. Test Case ID: 003 - Berechnung Wärmewiderstand für zwei Schichten mit vorgeschichteten Daten:	45
13. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]	46
13.1. Test Case ID: 004 - Berechnung Wärmewiderstand für drei Schichten:	46
14. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]	48
14.1. Test Case ID: 005 - Berechnung Wärmewiderstand für vier Schichten mit vorgeschichteten Werten:	48
15. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]	50
15.1. Test Case ID: 006_1 - Berechnung nach unvollständiger Eingabe Wanddicke:	50
16. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]	52
16.1. Test Case ID: 006_2 - Berechnung nach unvollständiger Eingabe Wärmeleitfähigkeit: ...	52
17. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]	54
17.1. Test Case ID: 007 - Berechnung mit "leerer Schicht":	54
18. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]	56
18.1. Test Case ID: 008 - Partielle Nullen:	56
19. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]	58
19.1. Test Case ID: 009 - alle Felder "Null":	58
20. Projekt Bauphysik Test Case: [Use Case: Plausibilität_prüfen]	59
20.1. Test Case ID: 021 - Plausibilität korrekter Daten:	59
21. Projekt Bauphysik Test Case: [Use Case: Plausibilität_prüfen]	60

21.1. Test Case ID: 022 - Plausibilität inkorrektter Daten:	60
22. Design	61

1. Vision BauphysikSE1

1.1. Einführung

Der Zweck dieses Dokuments ist es, die wesentlichen Bedarfe und Funktionalitäten des Bauphysikrechner-Systems zu sammeln, zu analysieren und zu definieren. Der Fokus liegt auf den Fähigkeiten, die von Stakeholdern und adressierten Nutzern benötigt werden, und der Begründung dieser Bedarfe. Die Details, wie das Bauphysikrechner-System diese Bedarfe erfüllt, werden in der Use-Case und Supplementary Specification beschrieben.

1.1.1. Zweck

Der Zweck dieses Dokuments ist es, die wesentlichen Anforderungen an das System aus Sicht und mit den Begriffen der künftigen Anwender zu beschreiben.

1.1.2. Gültigkeitsbereich (Scope)

Dieses Visions-Dokument bezieht sich auf das Bauphysikrechner-System, das von Team 14 entwickelt wird. Das System wird es der Professorin Rhena Krawietz erlauben, bauphysikalische Berechnungen zum Wärmedurchgangskoeffizienten durchführen zu können, um den Arbeitsaufwand bei der Erstellung und Korrektur von Aufgaben zu minimieren.

1.1.3. Definitionen, Akronyme und Abkürzungen

Definitionen, Akronyme und Abkürzungen werden im Glossar beschrieben.

1.2. Positionierung

1.2.1. Fachliche Motivation

Auf eine hochschulinterne Anfrage von Professor Jürgen Anke, dem Verantwortlichen des Moduls Software Engineering, antwortete die Professorin Rhena Krawietz mit dem Wunsch nach einer Softwarelösung zur Berechnung des Wärmedurchgangskoeffizienten. Infolge eines Projektzuordnungsverfahrens hat Professor Jürgen Anke das Team 14 mit der Projektbearbeitung im Rahmen des vorgenannten Moduls beauftragt. Da die Kundin derzeit bei der Erstellung und Kontrolle von Aufgaben zum Wärmedurchgangskoeffizienten nur unter Verwendung von Microsoft Excel oder eines Taschenrechners arbeiten kann, sind diese Aufgaben mit einem hohen Zeit- und Arbeitsaufwand verbunden. Ebenfalls zeitaufwändig ist für sie auch das Zusammensuchen der notwendigen Materialdaten. Die Auftraggeberin erhofft sich, diese Aufwände mithilfe eines Softwaresystems verringern zu können.

1.2.2. Problem Statement

Das Problem	Die Erstellung und die Kontrolle von Aufgaben zum Wärmedurchgangskoeffizienten sind für die Auftraggeberin mit einem hohen Aufwand verbunden.
betrifft	die Auftraggeberin
die Auswirkung davon ist	<ul style="list-style-type: none"> • hoher Zeitbedarf bei der Erstellung und Kontrolle von Aufgaben zum Wärmedurchgangskoeffizienten • Notwendigkeit der Erarbeitung von individuellen Rechenwegen für jede Problemstellung • erhöhte Gefahr von Rechenfehlern durch viele manuelle Eingaben • erhöhtes Risiko von fehlerhaften Prüfungsbewertungen
eine erfolgreiche Lösung wäre	Ein Softwaresystem, welches häufig verwendete Materialdaten enthält, alle notwendigen Berechnungen durchführt, Zwischen- und Endergebnisse ausgibt und Ausgaben visualisiert.

1.2.3. Positionierung des Produkts

Für	die Auftraggeberin
die	Berechnungen zum Wärmedurchgangskoeffizienten durchführen möchte
Das Produkt ist ein	Softwaresystem, welches Berechnungen zum Wärmedurchgangskoeffizienten ermöglicht
das	selbstständig und schnell alle notwendigen Berechnungen durchführt und bei Bedarf Zugriff auf häufig verwendete Materialdaten bietet
Im Gegensatz zu	einer Berechnung mithilfe von Microsoft Excel oder einem Taschenrechner
Unser Produkt	bietet die Möglichkeit, ohne eigenen Berechnungsaufwand direkt von den Materialdaten zu den Wärmewiderstandsdaten zu kommen

1.3. Stakeholder Beschreibungen

1.3.1. Zusammenfassung der Stakeholder

Name	Beschreibung	Verantwortlichkeiten
Auftraggeberin (Kundin)	Prof. Dr.-Ing. Rhena Krawietz, Professorin für Technische Physik an der HTW Dresden	führt Berechnungen mit dem Softwaresystem aus, um Aufgaben erstellen und kontrollieren zu können
Gesetzgeber	Vorgabe von rechtlichen Rahmenbedingungen	gibt Gesetze vor und überwacht deren Einhaltung, insbesondere im Hinblick auf die Lizenzierung
Lehrende	Lehrende der Hochschule für Technik und Wirtschaft Dresden mit Bezug zum Bauwesen	führen möglicherweise in Zukunft Berechnungen mit dem Softwaresystem aus
Projektteam	Team 14	versucht Kundenwünsche bei der Entwicklung des Softwaresystems umzusetzen
Qt	Lizenzgeber	legt Grundlagen für die Lizenzierung des Softwaresystems fest
Studierende	Studierende des Bauingenieurwesens oder anderer Studiengänge mit Studienschwerpunkt im Bauwesen	erhalten Prüfungsbewertungen auf Grundlage der Berechnungen des Softwaresystems, führen möglicherweise in Zukunft Berechnungen mit dem Softwaresystem aus
Team-Coach	Prof. Dr.-Ing. Jürgen Anke, Professor für Softwaretechnologie und Informationssysteme an der HTW Dresden	unterstützt und bewertet die Arbeit des Projektteams

1.3.2. Benutzerumgebung

Benutzerumgebung der Auftraggeberin

- Die Auftraggeberin wünscht, dass das Softwaresystem ausschließlich an ihrer Benutzerumgebung ausgerichtet wird. Die Anforderungen potenzieller zukünftiger Systemnutzer an die Benutzerumgebung sollen vernachlässigt werden.
- Die Auftraggeberin möchte das Softwaresystem alleine benutzen.

- Die Auftraggeberin möchte das Softwaresystem auf Laptops und Desktop-PCs verwenden.
- Zum Zeitpunkt der Auslieferung wird die Auftraggeberin das Betriebssystem Windows 10 nutzen.
- Das Softwaresystem soll von einem USB-Speichermedium aus lauffähig sein.
- Die Auftraggeberin möchte das Softwaresystem offline nutzen können.
- Das Softwaresystem soll mit einer handelsüblichen Computermouse und einer handelsüblichen Tastatur bedient werden können.

1.4. Produkt-/Lösungsüberblick

1.4.1. Bedarfe und Hauptfunktionen

Bedarf	Priorität	Features	Geplantes Release
Reihenberechnung durchführen	hoch	Berechnung und Ausgabe von j , R_{ges} , R_i , R_T , U , $\Delta\vartheta_k$ und ϑ_k für in Reihe angeordnete Wandschichten	xx
Berechnungsdaten laden	hoch	Laden von gespeicherten Eingabedaten und Berechnungsergebnissen	xx
Berechnungsdaten speichern	hoch	Speicherung von Eingabedaten und Berechnungsergebnissen	xx
Daten drucken	hoch	Druck auswählbarer Eingabedaten und Berechnungsergebnisse	xx
Fehleingaben verhindern	hoch	Verhinderung der Eingabe von offensichtlich falschen Eingabedaten (Buchstaben, Sonderzeichen, ...)	xx
Materialdaten eingeben	hoch	Ermöglichung der Eingabe der Materialdaten von bis zu 10 Wandschichten	xx
Einheit ändern	mittel	Möglichkeit der Änderung der Einheit von d_i	xx

Bedarf	Priorität	Features	Geplantes Release
Materialdaten vormerken	mittel	Möglichkeit der Speicherung von Materialdaten von bis zu 200 Werkstoffen	xx
Temperaturverlauf ausgeben	mittel	Ausgabe des grafisch visualisierten Temperaturverlaufs über die verschiedenen Wandschichten	xx
Parallelberechnung durchführen	niedrig	Berechnung und Ausgabe von Wärmewiderstandsdaten von parallel angeordnete Wandschichten	xxx
Tauwasserfreiheit berechnen	niedrig	Berechnung der Tauwasserfreiheit auf der Innenoberfläche einer Außenwand	xx

1.5. Zusätzliche Produktanforderungen

Anforderung	Priorität	Geplantes Release
Ausgabe von j , R_{ges} , R_i , R_T , U , $\Delta\vartheta_k$ und ϑ_k mit vier Nachkommastellen	hoch	xx
Deutsche Benutzeroberfläche	hoch	xx
System auf Windows 10 lauffähig	hoch	xx
System kann offline genutzt werden	hoch	xx
einfache Bedienbarkeit	mittel	xx
gute Verständlichkeit (Erklärungen)	mittel	xx

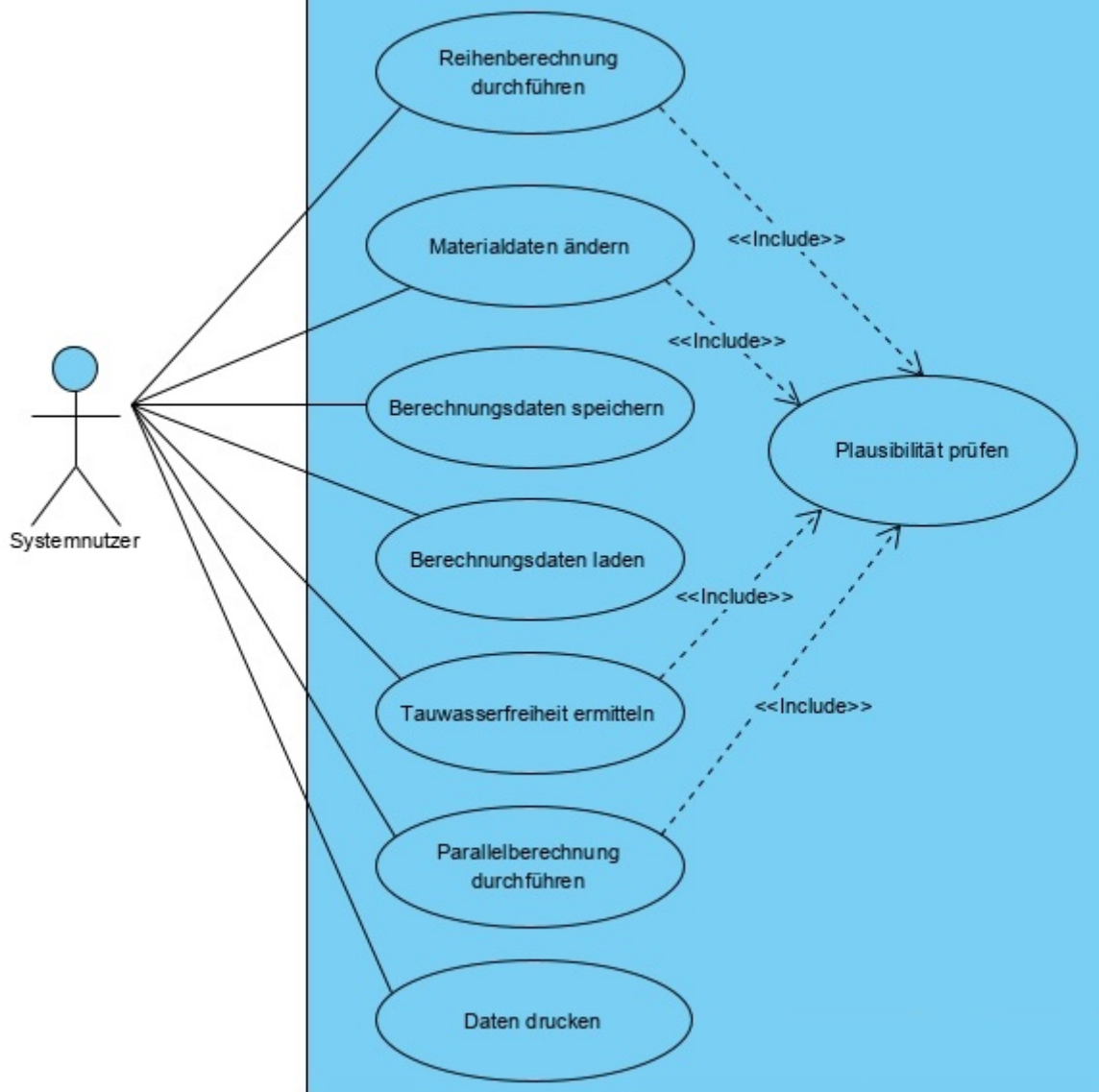
2. Use-Case Model BauphysikSE1

2.1. Identifizierte Use Cases

Hinweis: Die Use Cases wurden nach ihrer Priorität sortiert.

Kurzbezeichnung	Name	Akteur	Beschreibung
UC1	Reihenberechnung durchführen	Systemnutzer	Berechnung und Ausgabe von j , R_{ges} , R_i , R_T , U , $\Delta\vartheta_k$ und ϑ_k für in Reihe angeordnete Wandschichten
UC2	Plausibilität prüfen	Systemnutzer	Überprüfung der Gültigkeit von Eingabedaten
UC3	Daten drucken	Systemnutzer	Druck auswählbarer Informationen
UC4	Materialdaten ändern	Systemnutzer	Änderung von Materialdaten
UC5	Berechnungsdaten speichern	Systemnutzer	Speicherung von Eingabe- und Ergebnisdaten
UC6	Berechnungsdaten laden	Systemnutzer	Laden von gespeicherten Eingabe- und Ergebnisdaten
UC7	Parallelberechnung durchführen	Systemnutzer	Berechnung und Ausgabe von j , R_{ges} , R_i , R_T , U , $\Delta\vartheta_k$ und ϑ_k für Bauteile mit parallelen Wandschichten
UC8	Tauwasserfreiheit ermitteln	Systemnutzer	Berechnung der Kondenswasserbildung an Bauteilen

2.2. Use Case Diagramm



2.3. Ausgearbeitete Use Cases

2.4. Use Case: Reihenberechnung durchführen

2.4.1. Kurzbeschreibung

Der Use Case beschreibt einen Berechnungsvorgang für die Wärmewiderstandsdaten von in Reihe geschalteten Wandschichten.

2.4.2. Kurzbeschreibung der Akteure

Systemnutzer

will Wärmewiderstandsberechnungen durchführen.

Vorbedingungen

Der Systemnutzer hat das Softwaresystem gestartet.

Standardablauf (Basic Flow)

1. Der Use Case beginnt, wenn der Systemnutzer die Funktion der Reihenberechnung ausgewählt hat.
2. *WHILE* Das System bietet eine Eingabemöglichkeit für die Materialdaten.
 - a. *WHILE* Der Systemnutzer gibt Materialdaten ein.
 - i. *INCLUDE* Plausibilität prüfen
3. Der Systemnutzer startet die Berechnung durch das Softwaresystem.
4. Das Softwaresystem berechnet die Wärmewiderstandsdaten.
5. Das Softwaresystem gibt die Wärmewiderstandsdaten aus.
6. *IF* Systemnutzer will einen Temperaturverlauf ermitteln
 - a. *WHILE* Der Systemnutzer gibt Temperaturdaten ein.
 - i. *INCLUDE* Plausibilität prüfen
 - b. Der Systemnutzer bestätigt seine Eingaben.
 - c. Das Softwaresystem ermittelt den Temperaturverlauf.
 - d. Das Softwaresystem gibt den Temperaturverlauf aus.
7. Der Use Case ist abgeschlossen.

Alternative Abläufe

Alternativer Ablauf #2.1

Wenn der Systemnutzer im Schritt 2 des Standardablaufs vorgespeicherte Materialdaten verwenden möchte, dann

- i. Das Softwaresystem stellt dem Systemnutzer gespeicherte Materialdaten zur Verfügung.
- ii. Der Systemnutzer wählt gespeicherte Materialdaten aus.
- iii. Der Use Case wird im Schritt 2 des Standardablaufs fortgesetzt.

Alternativer Ablauf #2.2

Wenn der Systemnutzer im Schritt 2 des Standardablaufs weitere Materialschichten hinzufügen möchte, dann

- i. Der Systemnutzer fügt eine Schicht hinzu.
- ii. Das System schafft Eingabemöglichkeiten für die Daten einer zusätzlichen Materialschicht.
- iii. Der Use Case wird im Schritt 2 des Standardablaufs fortgesetzt.

Alternativer Ablauf #2.3

Wenn der Systemnutzer im Schritt 2 des Standardablaufs eine Materialschicht entfernen möchte, dann

- i. Der Systemnutzer entfernt eine Materialschicht.

- ii. Das System entfernt die Eingabemöglichkeiten für die Daten einer Materialschicht.
- iii. Der Use Case wird im Schritt 2 des Standardablaufs fortgesetzt.

Alternativer Ablauf 3a

Wenn im Schritt 3 des Standardablaufs berechnungsnotwendige Daten fehlen, dann

- i. Das Softwaresystem weist den Systemnutzer auf fehlende Daten hin.
- ii. Der Systemnutzer bestätigt die Kenntnisnahme der Information des Softwaresystems.
- iii. Der Use Case wird im Schritt 2 des Standardablaufs fortgesetzt.

Wesentliche Szenarios

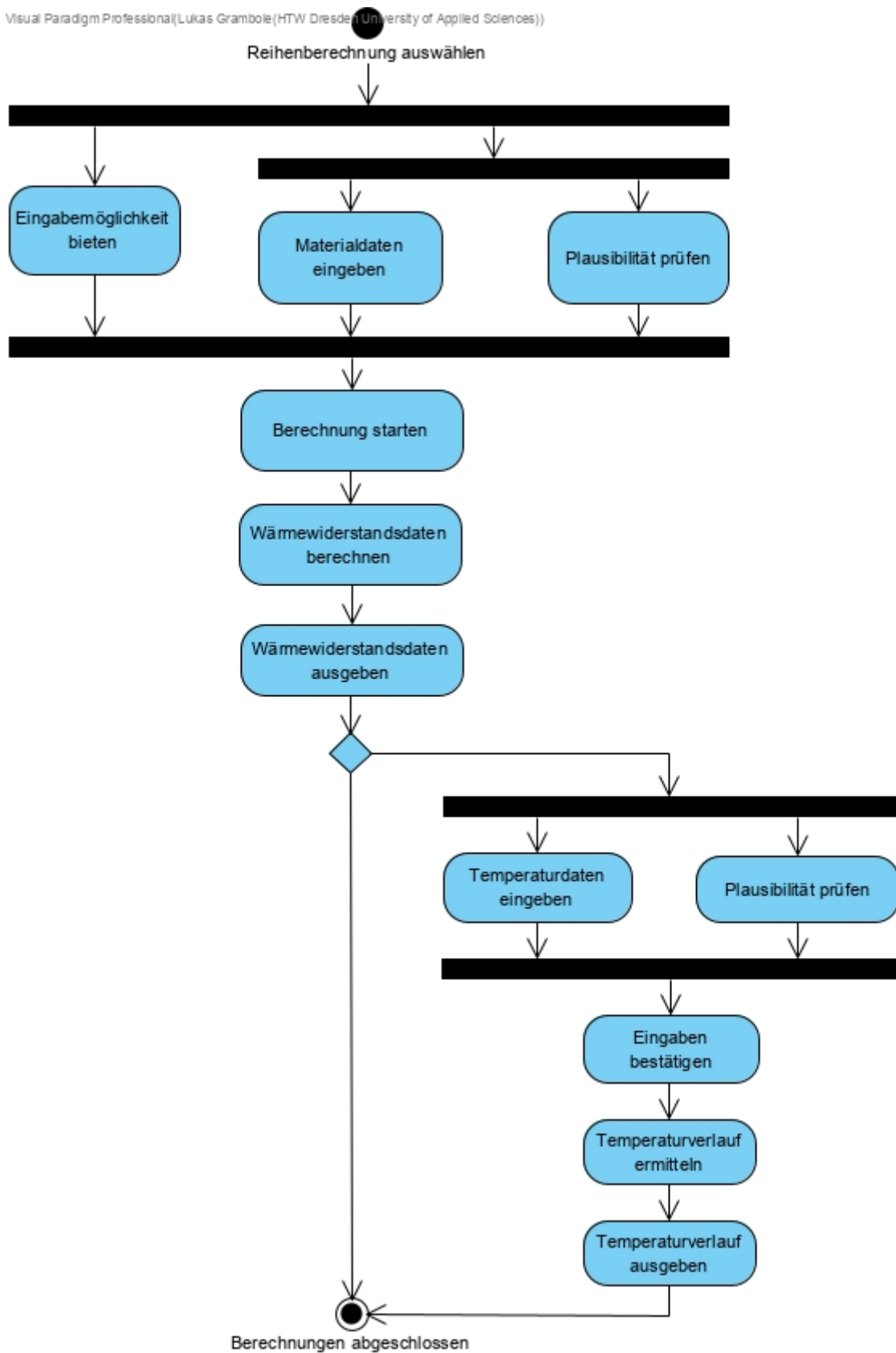
- **SC1:** Der Systemnutzer wählt die Funktion der Wärmewiderstandsberechnung aus und gibt die Materialdaten von zwei Schichten ein. Nachdem der Systemnutzer die Berechnung startet, berechnet das Softwaresystem die Wärmewiderstandsdaten und gibt diese aus. Die Berechnung wurde erfolgreich abgeschlossen.
- **SC2:** Der Systemnutzer wählt die Funktion der Wärmewiderstandsberechnung aus und gibt die Materialdaten von zwei Schichten ein. Nachdem der Systemnutzer die Berechnung startet, berechnet das Softwaresystem die Wärmewiderstandsdaten und gibt diese aus. Anschließend gibt der Systemnutzer Temperaturdaten ein und lässt einen Temperaturverlauf ermitteln. Die Berechnung wurde erfolgreich abgeschlossen.
- **SC3:** Der Systemnutzer wählt die Funktion der Wärmewiderstandsberechnung aus und wählt vorgespeicherte Materialdaten von zwei Schichten aus. Nachdem der Systemnutzer die Berechnung startet, berechnet das Softwaresystem die Wärmewiderstandsdaten und gibt diese aus. Die Berechnung wurde erfolgreich abgeschlossen.
- **SC4:** Der Systemnutzer wählt die Funktion der Wärmewiderstandsberechnung aus, fügt eine Schicht hinzu und gibt die Materialdaten von drei Schichten ein. Nachdem der Systemnutzer die Berechnung startet, berechnet das Softwaresystem die Wärmewiderstandsdaten und gibt diese aus. Die Berechnung wurde erfolgreich abgeschlossen.
- **SC5:** Der Systemnutzer wählt die Funktion der Wärmewiderstandsberechnung aus und gibt unvollständige Materialdaten von zwei Schichten ein. Nachdem der Systemnutzer die Berechnung startet, gibt das Softwaresystem eine Fehlermeldung aus. Der Systemnutzer ergänzt die fehlenden Daten und startet die Berechnung erneut. Dann berechnet das Softwaresystem die Wärmewiderstandsdaten und gibt diese aus. Die Berechnung wurde erfolgreich abgeschlossen.

2.4.3. Nachbedingungen

Bei erfolgreicher Durchführung des Use Case muss folgende Nachbedingungen erfüllt sein:

- Die Berechnungsergebnisse werden ausgegeben.

2.4.4. Aktivitätsdiagramm



2.4.5. Wireframes

2.5. Use Case: Plausibilität prüfen

2.5.1. Kurzbeschreibung

Der Use Case beschreibt einen Vorgang zur Prüfung der Gültigkeit von Daten.

2.5.2. Kurzbeschreibung der Akteure

Systemnutzer

will korrekte Eingabedaten eingeben.

2.5.3. Vorbedingungen

Der Systemnutzer hat einen Use Case gestartet, der eine Dateneingabe erfordert.

2.5.4. Standardablauf (Basic Flow)

1. Der Use Case beginnt, wenn der Systemnutzer Eingabedaten eingibt.
2. *WHILE* Systemnutzer gibt Eingabedaten ein
 - a. Das Softwaresystem prüft die Gültigkeit der Eingabedaten.
 - b. *IF* Softwaresystem erkennt eine ungültige Eingabe
 - i. Das Softwaresystem verhindert die Eingabe.
 - ii. Das Softwaresystem weist den Systemnutzer auf eine Fehleingabe hin.
3. Der Use Case ist abgeschlossen.

2.5.5. Wesentliche Szenarios

- **SC1:** Der Systemnutzer gibt d_i ein. Der Eingabewert beträgt 260. Es wird keine ungültige Eingabe erkannt. Der Use Case ist abgeschlossen.
- **SC2:** Der Systemnutzer gibt d_i ein. Er versucht 1o einzugeben. Das Softwaresystem verhindert die Eingabe des Buchstaben o und weist den Systemnutzer auf eine ungültige Eingabe hin. Der Use Case ist abgeschlossen.

2.5.6. Nachbedingungen

Bei erfolgreicher Durchführung des Use Case muss folgende Nachbedingungen erfüllt sein:

- Die eingegebenen Daten wurden im Hinblick auf ihre Plausibilität validiert.

2.6. Use Case: Daten drucken

2.6.1. Kurzbeschreibung

Der Use Case beschreibt die Erteilung eines Druckauftrags für den Druck von ausgewählten Daten.

2.6.2. Kurzbeschreibung der Akteure

Systemnutzer

will auswählbare Eingabedaten und Berechnungsergebnisse drucken.

2.6.3. Vorbedingungen

- Es liegen Berechnungsergebnisse vor.
- Es ist ein Drucker oder ein PDF-Drucker verfügbar.

2.6.4. Standardablauf (Basic Flow)

1. Der Use Case beginnt, wenn der Systemnutzer die Druckfunktion auswählt.
2. Das Softwaresystem gibt eine Vorauswahl von druckbaren Elementen aus.
3. *WHILE* Softwaresystem gibt eine Auswahl von Elementen für den Druck aus
 - a. *IF* Systemnutzer möchte andere Elemente drucken
 - i. Der Systemnutzer ändert die Auswahl.
 - b. *ELSE*
 - i. Der Systemnutzer belässt die Auswahl.
4. Der Systemnutzer löst den Druckauftrag aus.
5. Das Softwaresystem sendet den Druckauftrag an das Betriebssystem.
6. Der Use Case ist abgeschlossen.

2.6.5. Alternative Abläufe

Alternativer Ablauf #3.1

Wenn der Systemnutzer im Schritt 3 des Standardablaufs seine Auswahl als Vorauswahl festlegen möchte, dann

- i. Der Systemnutzer wählt die Funktionalität zur Festlegung der aktuellen Auswahl als Vorauswahl.
- ii. Das Softwaresystem speichert die aktuelle Auswahl als Vorauswahl.
- iii. Der Use Case wird im Schritt 3 des Standardablaufs fortgesetzt.

Alternativer Ablauf #3.2

Wenn der Systemnutzer im Schritt 3 des Standardablaufs eine Druckvorschau sehen möchte, dann

- i. Der Systemnutzer wählt die Funktionalität zur Erzeugung einer Druckvorschau.
- ii. Das Softwaresystem erstellt eine Druckvorschau.
- iii. Der Systemnutzer schließt die Druckvorschau.
- iv. Der Use Case wird im Schritt 3 des Standardablaufs fortgesetzt.

Alternativer Ablauf 4a

Wenn der Systemnutzer im Schritt 4 des Standardablaufs keine Elemente für den Druck ausgewählt hat, dann

- i. Das Softwaresystem gibt eine Fehlermeldung aus.
- ii. Der Systemnutzer bestätigt die Kenntnisnahme der Fehlermeldung.
- iii. Der Use Case wird im Schritt 2 des Standardablaufs fortgesetzt.

2.6.6. Wesentliche Szenarios

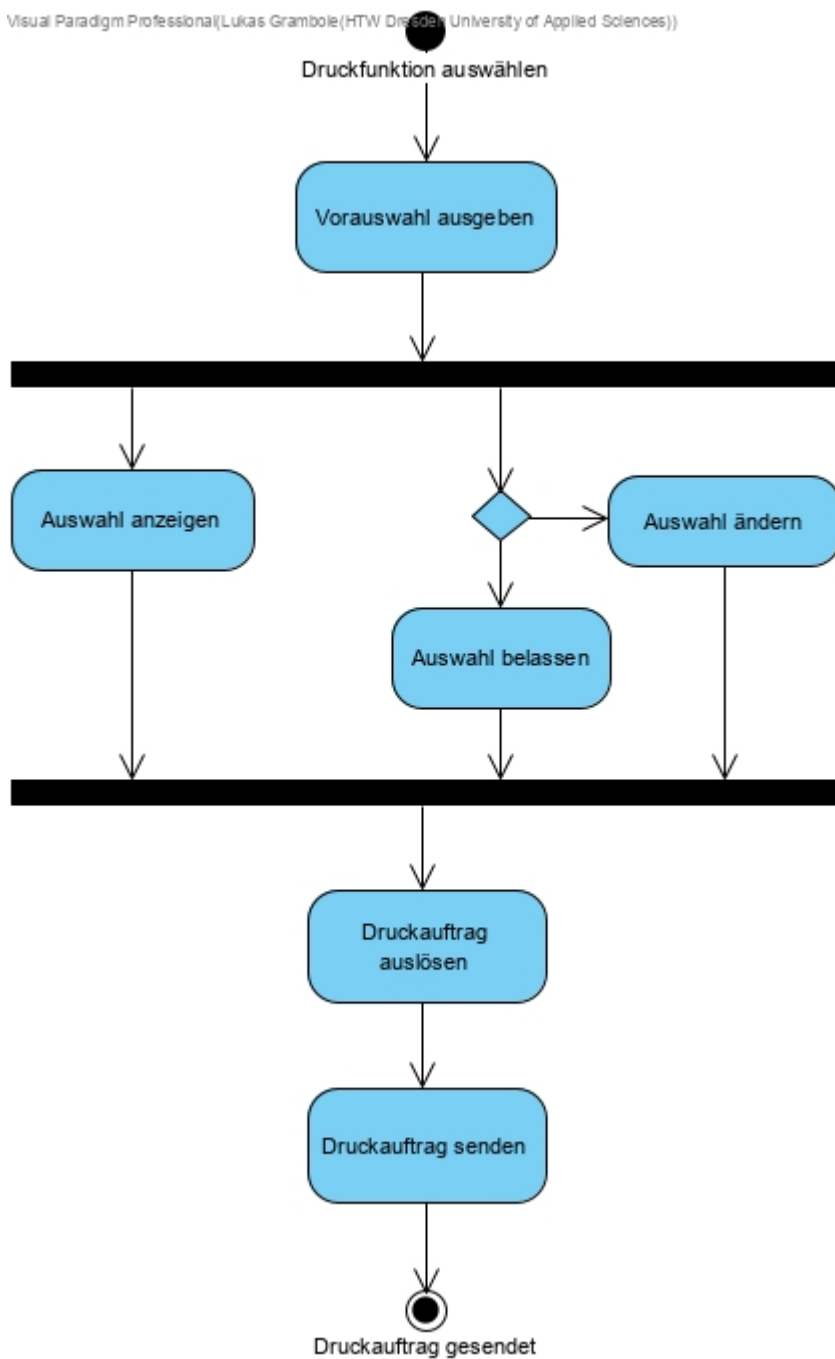
- **SC1:** Der Systemnutzer wählt die Druckfunktion aus. Er löst den Druckauftrag aus. Das Softwaresystem übermittelt den Druckauftrag an das Betriebssystem. Der Druckauftrag wurde erfolgreich erteilt.
- **SC2:** Der Systemnutzer wählt die Druckfunktion aus. Er wählt zusätzliche Elemente für den Druck aus und speichert seine aktuelle Auswahl als Vorauswahl. Er löst den Druckauftrag aus. Das Softwaresystem übermittelt den Druckauftrag an das Betriebssystem. Der Druckauftrag wurde erfolgreich erteilt.
- **SC3:** Der Systemnutzer wählt die Druckfunktion aus und lässt sich eine Druckvorschau anzeigen. Er löst den Druckauftrag aus. Das Softwaresystem übermittelt den Druckauftrag an das Betriebssystem. Der Druckauftrag wurde erfolgreich erteilt.
- **SC4:** Der Systemnutzer wählt die Druckfunktion aus. Er möchte andere Elemente drucken und wählt anschließend kein Element für den Druck aus. Das Softwaresystem weist den Systemnutzer darauf hin, dass kein Element für den Druck ausgewählt wurde. Der Systemnutzer fügt der Auswahl ein Element hinzu. Der Systemnutzer löst den Druckauftrag aus. Das Softwaresystem übermittelt den Druckauftrag an das Betriebssystem. Der Druckauftrag wurde erfolgreich erteilt.

2.6.7. Nachbedingungen

Bei erfolgreicher Durchführung des Use Case muss folgende Nachbedingung erfüllt sein:

- Der Druckauftrag wurde an das Betriebssystem gesendet.

2.6.8. Aktivitätsdiagramm



2.6.9. Wireframe

Frame

Prüfung WS 19/20

Neuer Tab

+

Auswahl zum Drucken:

Eingabedaten

☐ Schichtdicken

☐ Wärmeübergangswiderstände

☐ Wärmeleitfähigkeiten

☐ Temperaturen

Ergebnisdaten

☐ Wärmedurchgangskoeffizient

☐ Wärmedurchlasswiderstände

☐ Summierte Wärmedurchlasswiderstände

☐ Temperaturen an Schichtgrenzen

☐ Wärmestromdichte

Visualisierung

☐ Temperaturverlauf

☐ Visualisierung der Schichten

Als Vorauswahl festlegen

Druckvorschau

Drucken

3. BauphysikSE1 System-Wide Requirements Specification

3.1. Einführung

In diesem Dokument werden die systemweiten Anforderungen für das Bauphysikrechner-System spezifiziert. Die Gliederung erfolgt nach der FURPS+ Anforderungsklassifikation:

- Systemweite funktionale Anforderungen (F),
- Qualitätsanforderungen für Benutzbarkeit, Zuverlässigkeit, Effizienz und Wartbarkeit (URPS) sowie
- zusätzliche Anforderungen (+) für technische, rechtliche, organisatorische Randbedingungen



Die funktionalen Anforderungen, die sich aus der Interaktion von Nutzern mit dem System ergeben, sind als Use Cases in einem separaten Dokument festgehalten.

3.2. Systemweite funktionale Anforderungen

Alle funktionalen Anforderungen wurden als Use Cases ausgedrückt.

3.3. Qualitätsanforderungen für das Gesamtsystem

3.3.1. Benutzbarkeit (Usability)

NFAU-1: Das System muss Fehlermeldungen in einer Sprache ausgeben können, welche für den Systemnutzer verständlich ist.

- Zur Überprüfung werden durch einen Testnutzer fehlerhafte Eingaben getätigt, um anschließend zu testen, ob der Testnutzer durch Fehlermeldungen zu einem korrekten Vorgehen geleitet wird.

NFAU-2: Das System soll Hilfestellungen zu Eingabemöglichkeiten beinhalten.

- Diese NFA bezieht sich auf UC1, UC2, UC3, UC4, UC7 und UC8.
- Zur Überprüfung wird durch einen Testnutzer eine Beispielaufgabe mit dem Softwaresystem gelöst, um dabei zu testen, ob Hilfestellungen angezeigt werden.

NFAU-3: Das System muss eine deutsche Benutzeroberfläche beinhalten.

- Zur Überprüfung wird eine Beispielberechnung durch einen Testnutzer mit der Muttersprache Deutsch durchgeführt.

NFAU-4: Das System muss von einem Nutzer mit Fachwissen im Bereich der Bauphysik bedient werden können.

- Zur Überprüfung wird ein Testnutzer mit Fachwissen im Bereich der Bauphysik eine Beispielaufgabe mit dem System lösen.

3.3.2. Zuverlässigkeit (Reliability)

NFAR-1: Das System soll korrekte Ergebnisdaten ausgeben.

- Diese NFA bezieht sich auf UC1, UC7 und UC8.
- Zur Überprüfung wird mithilfe von mehreren Beispielaufgaben getestet, ob die Berechnungsergebnisse des Systems stimmen.

NFAR-2: Das System soll die Eingabe von unblausiblen Werten verhindern.

- Diese NFA bezieht sich auf UC2.
- Zur Überprüfung wird getestet, ob die Eingabe von ungültigen Zahlen, Buchstaben, oder Sonderzeichen in Eingabefelder verhindert wird.

NFAR-3: Das System soll gespeicherte Eingabe- und Ergebnisdaten nach einem Absturz wiederherstellen können.

- Diese NFA bezieht sich auf UC5 und UC6.
- Zur Überprüfung provoziert ein Testnutzer einen Systemabsturz und überprüft anschließend die Verfügbarkeit gespeicherter Eingabe- und Ergebnisdaten.

3.3.3. Effizienz (Performance)

NFAP-1: Das System soll Berechnungen mit bis zu 10 Materialschichten unterstützen.

- Diese NFA bezieht sich auf UC1, UC7 und UC8.
- Zur Überprüfung wird getestet, ob eine Beispielaufgabe mit 10 Wandschichten korrekt gelöst werden kann.

NFAP-2: Das System soll die Speicherung der Materialdaten von 200 Materialien ermöglichen.

- Diese NFA bezieht sich auf UC4.
- Zur Überprüfung wird getestet, ob die Materialdaten von 200 Materialien eingegeben werden können.

NFAP-3 Das System soll Ergebnisdaten mit vier Nachkommastellen ausgeben.

- Diese NFA bezieht sich auf UC1, UC7 und UC8.
- Zur Überprüfung löst ein Testnutzer eine Beispielaufgabe und prüft die Anzahl der Nachkommastellen bei den Ergebnisdaten.

3.3.4. Unterstützbarkeit (Supportability)

NFAS-1: Das System soll eine portable Software sein.

- Zur Überprüfung wird getestet, ob die Software ohne Installation genutzt werden kann.

NFAS-2: Das System soll den Wechsel der Eingabeeinheit von d_i ermöglichen.

- Diese NFA bezieht sich auf UC1 und UC7.
- Zur Überprüfung wird getestet, ob die Eingabeeinheit von d_i geändert werden kann und dabei korrekte korrekte Ergebnisdaten ausgegeben werden.

3.4. Zusätzliche Anforderungen

3.4.1. Einschränkungen / Constraints

- Das System muss offline nutzbar sein.
- Das System soll auf einem handelsüblichen Computer benutzbar sein.
- Das System muss mit dem Betriebssystem Windows 10 nutzbar sein.
- Das System muss von einem handelsüblichen USB-Speichermedium aus lauffähig sein.
- Eingaben in das System sollen mit einer handelsüblichen Maus und einer handelsüblichen Tastatur möglich sein.
- Es muss eine Open-Source-Anwendung entwickelt werden.

3.4.2. Interface Requirements

- Das System soll bekannte Symbole verwenden. Das bedeutet, dass zum Beispiel ein Drucker als Symbol für die Druckfunktion verwendet werden soll.
- Das System soll für verschiedene Bildschirmgrößen nutzbar sein.
- Hinweistexte zu Eingabefeldern sollen ausgeblendet werden können.

4. Glossar BauphysikSE1

4.1. Einführung

In diesem Dokument werden die wesentlichen Begriffe des Bauphysikrechner-Systems aus dem Anwendungsgebiet der Bauphysik definiert. Zur besseren Übersichtlichkeit sind Begriffe, Abkürzungen, physikalische Größen und Datendefinitionen gesondert aufgeführt.

4.2. Begriffe

Begriff	Definition und Erläuterung	Synonyme
Auswahl	Gruppe von Elementen, welche für eine nachfolgende Aktivität ausgewählt werden können	
Berechnungsergebnisse	Physikalische Größen und Diagramme, welche im Rahmen eines Berechnungsvorgangs ermittelt wurden	Berechnete Elemente, Ergebnisdaten
Eingabedaten	Berechnungsabhängige Daten, welche ein Systemnutzer in das Softwaresystem eingibt	
Eingabeeinheit	Physikalische Einheit von Eingabedaten	
Celsiuskala	Temperaturskala, bei der der Abstand zwischen dem Gefrier- und dem Siedepunkt des Wassers 100 Einheiten entspricht	
Kelvinskala	Temperaturskala, deren Nullpunkt der absolute Nullpunkt ist (-273,15°C)	
Systemnutzer	Bezeichnung für einen Akteur, der Berechnungen mit dem Softwaresystem durchführen möchte. Neben der Auftraggeberin sind auch Studierende und Lehrende potenzielle Systemnutzer.	
Tauwasser	Wasser, welches aus Wasserdampf kondensiert ist	
Tauwasserfreiheit	bezeichnet den Zustand einer tauwasserfreien Wand	

Begriff	Definition und Erläuterung	Synonyme
Temperaturdiagramm	Diagramm mit der Wanddicke auf der x-Achse und der Temperatur auf der y-Achse	Temperaturverlauf
Vorauswahl	Gruppe von Elementen, welche vorab in der Vergangenheit ausgewählt wurden	
Wärmestrom	gibt die übertragene Wärmemenge pro Zeiteinheit an	Wärmefluss

4.3. Akronyme & Abkürzungen

Abkürzung	Bedeutung	Erläuterung
NFA	Nichtfunktionale Anforderung	Beschränkung der vom System angebotenen Funktionalität
NFAP	Nichtfunktionale Anforderung Performance	Nichtfunktionale Anforderung in Bezug auf die Leistung eines Softwaresystems
NFAR	Nichtfunktionale Anforderung Reliability	Nichtfunktionale Anforderung in Bezug auf die Zuverlässigkeit eines Softwaresystems
NFAS	Nichtfunktionale Anforderung Supportability	Nichtfunktionale Anforderung in Bezug auf die Unterstützbarkeit eines Softwaresystems
NFAU	Nichtfunktionale Anforderung Usability	Nichtfunktionale Anforderung in Bezug auf die Benutzbarkeit eines Softwaresystems
SC	Szenario	Konkrete Instanz eines Use Case
UC	Use Case	Funktionale Anforderung eines Systems

4.4. Physikalische Größen

Abkürzung	Bedeutung	Einheit	Erläuterung
d_i	Dicke des Materials	mm, cm, dm, m	Beschreibung der Dicke der jeweiligen Wandschicht

Abkürzung	Bedeutung	Einheit	Erläuterung
i	Index-Nummer des Materials	1	Nummerierung eines Materials innerhalb einer Rechenaufgabe
j	Wärmestromdichte	$\text{W}\cdot\text{m}^{-2}$	physikalische Größe zur Beschreibung der Änderung der thermischen Energie eines Bauteils, Produkt aus U und $\Delta\vartheta$
n	Schichtanzahl	1	Anzahl der in Reihe geschalteten Wandschichten
R_{ges}	Summe aller Wärmedurchlasswiderstände	$\text{m}^2\cdot\text{K}\cdot\text{W}^{-1}$	Zwischenwert, Summe aus R_1, R_2, \dots, R_n
R_i	Wärmedurchlasswiderstand des Materials i	$\text{m}^2\cdot\text{K}\cdot\text{W}^{-1}$	Widerstand, den eine Materialschicht dem Wärmestrom entgegengesetzt, Quotient aus Materialdicke und Wärmeleitfähigkeit
R_{se}	Wärmeübergangswiderstand außen	$\text{m}^2\cdot\text{K}\cdot\text{W}^{-1}$	Wärmeübergangswiderstand des Materials an der äußeren Oberfläche
R_{si}	Wärmeübergangswiderstand innen	$\text{m}^2\cdot\text{K}\cdot\text{W}^{-1}$	Wärmeübergangswiderstand des Materials an der inneren Oberfläche
R_T	Wärmedurchgangswiderstand	$\text{m}^2\cdot\text{K}\cdot\text{W}^{-1}$	Widerstand, welcher dem Wärmestrom vom gesamten Bauteil inklusive der Oberflächen entgegengesetzt wird, Summe aus $R_{\text{si}}, R_{\text{se}}$ und R_{ges}
U	Wärmedurchgangskoeffizient	$\text{W}\cdot\text{m}^{-2}\cdot\text{K}^{-1}$	Maß für die Wärmedurchlässigkeit eines Bauteils, Kehrwert von R_T

Abkürzung	Bedeutung	Einheit	Erläuterung
$\Delta\vartheta$	Temperaturunterschied	K	Temperaturunterschied zwischen Innen- und Außentemperatur, Betrag der Differenz aus ϑ_i und ϑ_e
$\Delta\vartheta_k$	Temperaturunterschied an einer Schichtgrenze	K	Produkt aus j und dem überwundenen Wärmewiderstand (R_{si} , R_i oder R_{se})
λ_i	Wärmeleitfähigkeit des Materials i	$W\cdot m^{-1}\cdot K^{-1}$	Stoffeigenschaft, welche den Wärmestrom durch ein Material bestimmt
ϑ_e	Außentemperatur	$^{\circ}C$	Temperatur an der Wandaußenseite
ϑ_i	Innentemperatur	$^{\circ}C$	Temperatur an der Wandinnenseite
ϑ_k	Temperatur an der Schichtgrenze k	$^{\circ}C$	Berechnung von Lage der wärmeren Seite abhängig (vgl. Beispielrechnung der Auftraggeberin)

4.5. Verzeichnis der Datenstrukturen

Bezeichnung	Definition	Format	Gültigkeitsregeln
Materialdaten	$d_i, \lambda_i, R_{se}, R_{si}$	Double	aus Zahlen, ≥ 0
Temperaturdaten	$\Delta\vartheta_k, \vartheta_k, \Delta\vartheta_e, \vartheta_i$	Double	aus Zahlen, ≥ 0 (bei Kelvinskala), beziehungsweise $\geq -273,15$ (bei Celsiuskala)
Wärmewiderstandsdaten	j, R_{ges}, R_i, R_T, U	Double	aus Zahlen, ≥ 0

Projektdokumentation

- Projektplan
- Risikoliste
- Iteration Plan (für zwei ausgewählte Iterationen)

5. Projektplan BauphysikSE1

5.1. Einführung

Dieses Dokument ist der Projektplan für das SE1 Projekt Bauphysik. Ziel des Dokumentes ist es die Organisation und Methoden des Projektes im Groben wiederzugeben.

5.2. Projektorganisation

Die Tätigkeiten sind nach Rollen und Themengebiet aufgeteilt. Eine Person in Ihrer Rolle trägt Verantwortung in diesem Themenbereich und trifft hier nach eigenem Ermessen Entscheidungen. Alle anstehenden und abgeschlossenen Tätigkeiten werden flexibel in einer work-item-list festgehalten und soweit einer Iteration zugeordnet, bearbeitet.

Die Rollenverteilung sieht wie folgt aus:

Name, Vorname	Primäre Rolle	Sekundäre Rolle
Klassowski, Denis	Project Manager	Analyst
Däbler, Michael	Analyst	Technical Writer
Grambole, Lukas	Analyst	---
Grieß, Christian	Architect	Developer
Baburkin, Yewgenji	Developer	Project Manager
Ullmann, Max	Developer	Tester
Lehmann, Christian	Tester	Architect
Fritzsche, Felix	Deployment Eng	Developer

5.3. Praktiken und Bewertung

Die Lösung für das Projekt wird iterativ entwickelt. Hierfür Orientiert sich das Team am Open-UP. Für wichtige Dokumente werden die vom Coach bereitgestellten Templates genutzt und ausgefüllt. Alle Aufgaben und Probleme werden zunächst festgehalten und je nach Priorität einer Iteration zugeordnet. Ist eine Iteration abgeschlossen, wird anhand der erledigten Aufgaben die Geschwindigkeit (Velocity) berechnet um zu überprüfen, ob unser Ziel der Geschwindigkeit gehalten werden konnte.

Desweiteren finden sich in den Iteration-Plans die Assessments, wo das Team die Iteration bewertet.

Alle Dokumente zum Projekt befinden sich im Github-Repository.

5.4. Meilesteine und Ziele

Größere Ziele die wir einnerhalb einer Iteration erreichen möchten.

Iteration	Primary objectives (risks and use case scenarios)	Scheduled start or milestone	Target velocity
It. 0	<ul style="list-style-type: none"> • Vertraut Machen mit OpenUP • Kennenlernen des Teams • Kundengespräch abhalten 	02.12.2019	-
It1	<ul style="list-style-type: none"> • Kundengespräch analysieren • Elementare Projektdaten erarbeiten (ver.1) • Technische Vision erarbeiten • erste USE_CASES • erste TEST_CASES • Architecture (ver.1) • Risiko: Missverständnisse 	16.12.2019	16
It2	<ul style="list-style-type: none"> • Zusammen mit dem Kunden den Scope erweitern • Vision aktualisieren • Test-Cases aktualisieren • Use-Cases aktualisieren • Architecture Notebook aktualisieren • Technische Vision anpassen • Risiko: Scope zu weit erweitert 	05.01.2020	12
It3	<ul style="list-style-type: none"> • Erweiterten Scope umsetzen und Projektdaten verfeinern • Vision und Scope im Meeting besprechen / alle auf einen Stand bringen • Wireframes erstellen • Vertikale Prototypen beginnen • Technische Vision finalisieren • Risiko: Zeit pro Iteration gekürzt -> Verlust der Übersicht 	19.01.2020	12

Iteration	Primary objectives (risks and use case scenarios)	Scheduled start or milestone	Target velocity
It4	<ul style="list-style-type: none"> • Auf eine Präsentation für den Kunden vorbereiten • Clickdummy erstellen • Visuellen Prototyp optimieren • Technischen Prototyp anfangen (Berechnung) 	26.01.2020	12

5.5. Deployment

N/A

5.6. Erkenntnisse (Lessons learned)

N/A (Einzelne Erkenntnisse und Bewertung der Iterationen befinden sich im Iteration_plan)

6. Risikoliste Bauphysik

In diesem Dokument sind die wesentlichen Risiken des Projekts aufgeführt. Dabei werden folgende Attribute verwendet:

- 1. **Typ:** Ressourcen, Geschäftlich, Technisch, Zeitlich
- 2. **Auswirkung (IMP):** Wert zwischen 1 (niedrig) und 5 (hoch), der die Auswirkungen auf das Projekt angibt, wenn das Risiko eintritt
- 3. **Wahrscheinlichkeit (PRB):** Prozentangabe für die Eintrittswahrscheinlichkeit des Risikos
- 4. **Stärke (MAG):** Produkt aus Auswirkung und Wahrscheinlichkeit (damit kann die Liste sortiert werden)

Die Risiken sind in folgender Tabelle dargestellt. Das Datum des Dokuments oben gibt an, wann die Risikoliste zuletzt aktualisiert wurde.

ID	Datum	Name	Beschreibung	Typ	IMP	PRB	MAG	Owner	Gegenmaßnahme
1	01.12.2019	Teammitglied fällt aus	Mitglied des Teams kann (dauerhaft oder zeitweise) nicht mehr am Projekt teilnehmen	Ressourcen	3	100%	3	Klassowski, Denis	Backup-Rollen nutzen und Know-How zwischen diesen Personen kontinuierlich teilen

ID	Datum	Name	Beschreibung	Typ	IMP	PRB	MAG	Owner	Gegenmaßnahme
10	15.01.2020	Fehlinterpretationen	Fachbegriffe und projektspezifische Bezeichnungen werden falsch verstanden	Geschäftlich	4	50%	2	Däbler, Michael	Erklärung von spezifischen Bezeichnungen im Glossar, regelmäßige Durchführung von Team-Meetings
7	13.01.2020	Stakeholder	Veränderungen im Bereich der Stakeholder erzeugen veränderte Anforderungen	Geschäftlich	2	60%	1,2	Däbler, Michael	iterative Vorgehensweise, regelmäßige Durchführung von Kundengesprächen
8	13.01.2020	Benutzumgebung	Änderungen der Benutzumgebung gefährden die Nutzbarkeit des Software systems	Technisch	3	40%	1,2	Däbler, Michael	Kundenanforderungen zur Benutzumgebung feststellen und abnehmen lassen

ID	Datum	Name	Beschreibung	Typ	IMP	PRB	MAG	Owner	Gegenmaßnahme
9	13.01.2020	Projektmfang	Auftraggeberin wünscht sich nach den ersten Präsentationen zusätzliche Funktionalität	Geschäftlich	3	40%	1,2	Klassowski, Denis	Erweiterungen u.U. ablehnen
5	06.12.2019	Zieldefinition	Das Problem der Auftraggeberin wurde nicht verstanden	Geschäftlich	5	20%	1	Däbler, Michael	Abnahme von Projektbestandteilen durch Auftraggeberin, regelmäßige Durchführung von Kundengesprächen
4	03.12.2019	Einsatz unbekannter Technologien	Es werden neue Tools eingesetzt, zu denen die Projektmitglieder keine Erfahrungen haben	Technisch	1	60%	0,6	Baburkin, Yewgenij	

ID	Datum	Name	Beschreibung	Typ	IMP	PRB	MAG	Owner	Gegenmaßnahme
3	01.12.2019	Terminfindung	Terminfindung mit der Auftraggeberin gestaltet sich schwierig	Zeitlich, Geschäftlich	2	30%	0,6	Klassowski, Denis	alternative Kommunikation mit dem Kunden
2	01.12.2019	Rollenverteilung	Unklare Definitionen von Rollen im Projekt führen zu Verzögerungen	Ressourcen	2	20%	0,4	Klassowski, Denis	Rollenverteilung gemäß Open Unified Process, regelmäßige Durchführung von Team-Meetings
6	16.12.2019	Ressourcenproblem	Es sind nicht genügend Entwickler vorhanden um die Programmierung im angestrebten Zeitraum umzusetzen	Zeitlich	2	10%	0,2	Baburkin, Yewgenij	Backup-Rollen

7. Iterationsplan 1 BauphysikSE1

7.1. Meilensteine

Meilenstein	Datum
Beginn der Iteration	16.12.2019
Kundengespräch analysieren	
Elementare Projektdateien erarbeiten (ver.1)	
Technische Vision erarbeiten	
Ende der Iteration	04.01.2020

7.2. Wesentliche Ziele

- Einen Überblick über das Projekt, Projektumfang und Möglichkeiten erarbeiten
- Arbeitsweise des Open-UP direkt auf Problemstellung anwenden
- Besprechen des Problems und Vision im Team (möglichst gleiches Verständnis des Problems erarbeiten)
- Erste Versionen der Templates erstellen
- Mehrere mögliche Technische Optionen besprechen.

7.3. Aufgabenzuordnung

Die folgenden Aufgaben werden in dieser Iteration bearbeitet:

Aufgabe bzw. Beschreibung	Priorität	Schätzung der Größe (Punkte)	Status	Referenzen	geplante Iteration	Zugewiesen (Name)	Gearbeitete Stunden	Schätzung der verbleibenden Stunden
Vision V0.1 erstellen	hoch	7	DONE	x	I1	Michael Däbler, Lukas Grambole	4	3
Glossar V0.1 erstellen	hoch	4	DONE	x	I1	Michael Däbler, Lukas Grambole	2	2

Aufgabe bzw. Beschreibung	Priorität	Schätzung der Größe (Punkte)	Status	Referenzen	geplante Iteration	Zugewiesen (Name)	Gearbeitete Stunden	Schätzung der verbleibenden Stunden
Architekten Notizbuch V0.1 erstellen	mittel	4	DONE	x	I1	Christian Griefß	2	0
Projektplan V0.1 erstellen	hoch	2	DONE	x	I1	Denis Klassowski	2	0
Erste USE-CASE	mittel	2	DONE	x	I1	Christian Lehmann	3	0
Erste TEST-CASE	niedrig	1	DONE	x	I1	Christian Lehmann	2	0
Iteration Plan It1 erstellen	hoch	1	DONE	x	I1	Denis Klassowski	1	0
Iteration Plan It2 erstellen	niedrig	1	DONE	x	I1	Denis Klassowski	1	0
Erste Technische Ideen intern vorstellen	mittel	3	DONE	x	I1	Yewgenji Baburkin, Max Ullmann, Christian Griefß	3	0

7.4. Probleme (optional)

Problem	Status	Notizen
Unterschiedliche Auffassung der Problemstellung	NO LONGER A PROBLEM	In mehreren Meetings wurde die Vision besprochen und im Laufe der Iterationen verfeinert
Kunde wirkt unsicher bei Vorstellung der Idee / Kunde weiß nicht genau was er will	NO LONGER A PROBLEM	Unsicherheiten wurden in It2 beseitigt

7.5. Bewertungskriterien

- Sind die erstellten Dokumente abgestimmt
- Gibt es negatives Feedback bei Gruppenarbeiten
- Postive Rückmeldung vom Kunden
- Wird die Work-Item-List von jedem eingehalten
- Wurde die geplante Velocity eingehalten

7.6. Assessment

Assessment Ziel	Iteration 1
Assessment Datum	23.12.2019
Teilnehmer	Alle Teamteilnehmer
Projektstatus	Ist angelaufen. Keine großen Probleme

- Beurteilung im Vergleich zu den Zielen
Alle Ziele wurden erreicht.
- Geplante vs. erledigte Aufgaben
Fast alle für Iteration 1 geplanten Aufgaben wurden erfolgreich bearbeitet. Noch nicht viele geplante Aufgaben für Iteration 2.
- Beurteilung im Vergleich zu den Bewertungskriterien
Gutes Feedback vom Kunden erhalten. Team war sehr gut vorbereitet und Kunde war positiv überrascht.
Teamarbeit erfolgte eigeninitiativ, reibungslos und lieferte gute Ergebnisse.
Das gesamte Team zieht an einem Strang. Alle sind sich über Projektziele im Klaren und wollen ein gutes Ergebnis erzielen.
Geplante Velocity ist um 1 niedriger als tatsächliche Velocity
- Andere Belange und Abweichungen
N/A

8. Iterationsplan 2 BauphysikSE1

8.1. Meilensteine

Meilenstein	Datum
Beginn der Iteration	05.01.2020
Der Scope muss erweitert werden	
Dokumente müssen an der neuen Scope angepasst werden	
Übersicht der Risiken erstellen	
Erste Idee der Visualisierung fertigstellen	
Ende der Iteration	18.01.2020

8.2. Wesentliche Ziele

- Kundengespräch analysieren
- Inception Phase möglichst abschließen
- Visueller Prototyp muss erstellt werden
- Use-Cases müssen präsentierbar sein
- Entscheidung bei der technischen Umsetzung treffen

8.3. Aufgabenzuordnung

Die folgenden Aufgaben werden in dieser Iteration bearbeitet:

Aufgabe bzw. Beschreibung	Priorität	Schätzung der Größe (Punkte)	Status	Referenzen	geplante Iteration	Zugewiesen (Name)	Gearbeitete Stunden	Schätzung der verbleibenden Stunden
Architekten Notizbuch aktualisieren	hoch	2	DONE	x	I2	Grieß, Christian	2	0
Vision und Glossar aktualisieren	hoch	3	DONE	x	I2	Grambolle, Lukas; Däbler, Michael	2	3

Use-Case identifizieren und ausarbeiten	hoch	8	DONE	x	I2	Grambol e, Lukas Däbler, Michael	7	0
NFA identifizieren	hoch	3	DONE	x	I2	Grambol e, Lukas; Däbler, Michael	2	1
Architecture Notebook und Tech. Vision aktualisieren	hoch	2	DONE	x	I2	Grieß, Christian	4	0
(Kunden-)Gesprächsprotokoll erfassen und nachbearbeiten	hoch	2	DONE	x	I2	Baburkin , Yewgenji; Fritzsche, Felix	3	0
(Coach-)Gesprächsprotokoll erfassen und nachbearbeiten	mittel	1	DONE	x	I2	Grambol e, Lukas	1	0
UI Entwurf Wireframe erstellen	hoch	4	DONE	x	I2	Fritzsche, Felix; Baburkin , Yewgenji	4	0
Risk List aktualisieren	hoch	2	DONE	x	I2	Klassowski, Denis	3	0
Clickdummy erstellen	hoch	5	NOT DONE	x	I2	Ullmann, Max	0	5

Entscheiden welche Technologien verwendet werden	hoch	1	DONE	x	I2	Ullmann Max; Baburkin, Yewgenji; Grief, Christian	1	0
TEST-CASE anpassen	niedrig	1	DONE	x	I2	Christian Lehmann	2	0

8.4. Probleme (optional)

Problem	Status	Notizen
Frage zu Aktualisierungsschnittstellen nicht protokolliert	PENDING	Diese Frage wurde im Rahmen des 2. Kundengesprächs gestellt, jedoch nicht ausreichend protokolliert.
Aufgrund der Neujahrspause muss die Iteration 3 verkürzt werden	NO LONGER A PROBLEM	Team ist einverstanden und mit dieser Maßnahme bleiben wir weiterhin im Zeitplan.

8.5. Bewertungskriterien

- Ist der Scope ausreichend erweitert
- Ist die Qualität der Use-Cases gut
- Wurden die neuen Wünsche des Kunden erfolgreich in alten Dokumente eingearbeitet

8.6. Assessment

Assessment Ziel	Ist der neue Scope ausreichend
Assessment Datum	13.01.2020
Teilnehmer	Alle bis auf Ullmann Max
Projektstatus	Scope und damit das Projekt steht in Frage

- Beurteilung im Vergleich zu den Zielen
Das Ziel des visuellen Prototyps wurde auf Iteration 3 verschoben. Alle anderen Ziele wurden größtenteils erreicht.
- Geplante vs. erledigte Aufgaben
Ziel und damit auch die Aufgabe des visuellen Prototyps verschoben.
Alle anderen Aufgaben wurden größtenteils erfüllt.

- Beurteilung im Vergleich zu den Bewertungskriterien

Coach war mit dem neuen Scope einverstanden.

Bisherige Probleme des Kunden werden weiterhin, wie in Iteration 1 erarbeitet, lösbar sein.
Kunde wirkt damit zufrieden.

Use-Cases sind verfeinert worden. Qualitätsprüfung noch ausstehend.

- Andere Belange und Abweichungen

Leider fand das Kundengespräch aufgrund von organisatorischen Problemen relativ spät statt.

Um im Zeitplan zu bleiben haben wir als Team entschieden die Iteration 3 und 4 zu verkürzen.

Entwurfsdokumentation

- Architektur-Notizbuch
- Test Cases
- Design

9. Architecture Notebook Bauphysik

9.1. Zweck

Dieses Dokument beschreibt die Philosophie, Entscheidungen, Nebenbedingungen, Begründungen, wesentliche Elemente und andere übergreifende Aspekte des Systems, die Einfluss auf Entwurf und Implementierung haben.

9.2. Architekturziele und Philosophie

Folgende Architektur bezieht sich auf eine Softwarelösung die ohne Installation, als Desktopanwendung, über ein USB-Speichermedium auszuführen sein soll.

Die Auftraggeberin wünscht sich ein vom Internet, unabhängiges Programm.

Sie möchte ein wartungsfreies Produkt.

Außerdem wird nicht erwartet, dass sich in Zukunft etwas an der Systemumgebung ändern wird.

Es ist nur für die Eigenbenutzung gedacht (kurzfristige Nutzung durch andere möglich).

Die Software ist für Spezialisten in dem Gebiet der Bauphysik und soll für diese einfach zu bedienen sein.

Das Drucken von ausgeführten Rechnungen muss möglich sein.

9.3. Annahmen und Abhängigkeiten

1. Nutzer verwenden Desktop-PC oder Laptop mit USB-Port.
2. Endgeräte laufen unter Windows 10.
3. Eingabegeräte sind Maus (Touchpad) und Tastatur.
4. Keine Maßnahmen zur Barrierefreiheit nötig.
5. Nutzung ausschließlich durch Spezialisten der Bauphysik.
6. Keine Internetanbindung vorhanden.
7. Dokumente werden über das Betriebssystem gedruckt/erstellt.

9.4. Architektur-relevante Anforderungen

1. Das System muss auf Windows 10 Rechnern ausführbar sein.

9.5. Entscheidungen, Nebenbedingungen und Begründungen

1. Software wird mit dem GUI tool PyQT in Python geschrieben (GNU / GPL License).
 - a. Obwohl unsere Developer keine Erfahrung mit Python/QT haben, wollen wir dieses Projekt nutzen, um uns Grundkenntnisse anzueignen.
2. PyInstaller oder Py2Exe um Standalone zu erzeugen.

9.6. Architekturmechanismen

1. Persistenz

Die erstellten Rechnungen sollen bei Absturz des Systems nicht verloren gehen(Ausnahme: Aktuelle Rechnung).

2. Lokalisation

Die Software wird nur in deutsch gehalten, auf jeden Fall bezüglich der ersten Version.

Aber jede der individuellen Komponenten sollte technisch auch die englische Sprache unterstützen.

9.7. Architektursichten (Views)

10. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]

10.1. Test Case ID: 001 - Berechnung Wärmewiderstand für zwei Schichten:

10.1.1. Beschreibung

Dieser Test Case evaluiert das Verhalten des entworfenen Softwaresystems, nachdem der Systemnutzer die Funktion der Wärmewiderstandsberechnung ausgewählt und die Materialdaten von zwei Schichten eingegeben hat. Nachdem der Systemnutzer die Berechnung startet, berechnet das Softwaresystem die Wärmewiderstandsdaten und gibt diese aus.

Ergebnis dieses Tests sind die korrekt berechneten Werte für den Wärmewiderstand. Die Berechnung wurde erfolgreich abgeschlossen.

10.1.2. Vorbedingungen

<Vorbedingung 1> Die Software ist gestartet und einsatzbereit.

<Vorbedingung 2> Der Systemnutzer hat die Eingabefelder der Wärmewiderstandsberechnung für zwei Schichten mit folgenden Werten befüllt:

- Wanddicke der ersten Schicht: $d_1 = \dots$ cm
- Wanddicke der zweiten Schicht: $d_2 = \dots$ cm
- Wärmeleitfähigkeit der ersten Schicht: $\lambda_1 = \dots$ W/(m*K)
- Wärmeleitfähigkeit der zweiten Schicht: $\lambda_2 = \dots$ W/(m*K)

<Vorbedingung 3> Der Systemnutzer löst die Berechnung aus.

10.1.3. Nachbedingungen

<Nachbedingung 1> Die Software hat keinen Fehler gemeldet.

<Nachbedingung 2> Die Software zeigt die berechneten Ergebnisse an.

10.1.4. Benötigte Daten

Als Testdaten werden Werte benötigt, die idealerweise aus realen Aufgaben zur Berechnung des Wärmeübergangs stammen. Bestenfalls Aufgaben mit geprüften selbst berechneten oder vorgegebenen korrekten Ergebnissen, um den Output der Software entsprechend validieren zu können.

11. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]

11.1. Test Case ID: 002 - Berechnung Wärmewiderstand und Temperaturverlauf für zwei Schichten:

11.1.1. Beschreibung

Dieser Test Case evaluiert das Verhalten des entworfenen Softwaresystems, nachdem der Systemnutzer die Funktion der Wärmewiderstandsberechnung ausgewählt und die Materialdaten von zwei Schichten eingegeben hat. Nachdem der Systemnutzer die Berechnung startet, berechnet das Softwaresystem die Wärmewiderstandsdaten und gibt diese aus. Anschließend gibt der Systemnutzer Temperaturdaten ein und lässt einen Temperaturverlauf ermitteln.

Ergebnis dieses Tests sind die korrekt berechneten Werte für den Wärmewiderstand und den Temperaturverlauf, sowie eine grafische Veranschaulichung des Temperaturverlaufs. Die Berechnung wurde erfolgreich abgeschlossen

11.1.2. Vorbedingungen

<Vorbedingung 1> Die Software ist gestartet und einsatzbereit

<Vorbedingung 2> Der Systemnutzer hat die Eingabefelder der Wärmewiderstandsberechnung für zwei Schichten mit folgenden Werten befüllt:

- Wanddicke der ersten Schicht: $d_1 = \dots$ cm
- Wanddicke der zweiten Schicht: $d_2 = \dots$ cm
- Wärmeleitfähigkeit der ersten Schicht: $\lambda_1 = \dots$ W/(m*K)
- Wärmeleitfähigkeit der zweiten Schicht: $\lambda_2 = \dots$ W/(m*K)

<Vorbedingung 3> Der Systemnutzer löst die Berechnung aus

<Vorbedingung 4> Der Systemnutzer möchte den Temperaturverlauf ermitteln und gibt dafür folgende Werte ein:

- Innentemperatur: $T_i = \dots$ °C
- Außentemperatur: $T_a = \dots$ °C

<Vorbedingung 5> Der Systemnutzer bestätigt die Eingabe der Temperaturdaten

11.1.3. Nachbedingungen

<Nachbedingung 1> Die Software hat keinen Fehler gemeldet.

<Nachbedingung 2> Die Software zeigt die berechneten Ergebnisse für den Wärmewiderstand und den Temperaturverlauf an.

<Nachbedingung 3> Die Software zeigt eine grafische Veranschaulichung des Temperaturverlaufs an.

11.1.4. Benötigte Daten

Als Testdaten werden Werte benötigt, die idealerweise aus realen Aufgaben zur Berechnung des Wärmeübergangs stammen. Bestenfalls Aufgaben mit geprüften selbst berechneten oder vorgegebenen korrekten Ergebnissen, um den Output der Software entsprechend validieren zu können.

12. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]

12.1. Test Case ID: 003 - Berechnung Wärmewiderstand für zwei Schichten mit vorgespeicherten Daten:

12.1.1. Beschreibung

Dieser Test Case evaluiert das Verhalten des entworfenen Softwaresystems, nachdem der Systemnutzer die Funktion der Wärmewiderstands Berechnung ausgewählt und die Materialdaten von zwei Schichten eingegeben hat. Dabei werden die Werte für die Wärmeleitfähigkeit aus vorgespeicherten Materialdaten ausgewählt. Nachdem der Systemnutzer die Berechnung startet, berechnet das Softwaresystem die Wärmewiderstandsdaten und gibt diese aus.

Ergebnis dieses Tests sind die korrekt berechneten Werte für den Wärmewiderstand. Die Berechnung wurde erfolgreich abgeschlossen

12.1.2. Vorbedingungen

<Vorbedingung 1> Die Software ist gestartet und einsatzbereit

<Vorbedingung 2> Der Systemnutzer hat die Eingabefelder der Wärmewiderstands Berechnung für zwei Schichten mit folgenden Werten befüllt:

- Wanddicke der ersten Schicht: $d_1 = \dots$ cm
- Wanddicke der zweiten Schicht: $d_2 = \dots$ cm
- Wärmeleitfähigkeit der ersten Schicht: es wird der Koeffizient für das Material x gewählt
- Wärmeleitfähigkeit der zweiten Schicht: es wird der Koeffizient für das Material y gewählt

<Vorbedingung 3> Der Systemnutzer löst die Berechnung aus

12.1.3. Nachbedingungen

<Nachbedingung 1> Die Software hat keinen Fehler gemeldet.

<Nachbedingung 2> Die Software zeigt die berechneten Ergebnisse an.

12.1.4. Benötigte Daten

Als Testdaten werden Werte benötigt, die idealerweise aus realen Aufgaben zur Berechnung des Wärmeübergangs stammen. Bestenfalls Aufgaben mit geprüften selbst berechneten oder vorgegebenen korrekten Ergebnissen, um den Output der Software entsprechend validieren zu können.

13. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]

13.1. Test Case ID: 004 - Berechnung Wärmewiderstand für drei Schichten:

13.1.1. Beschreibung

Dieser Test Case evaluiert das Verhalten des entworfenen Softwaresystems, nachdem der Systemnutzer die Funktion der Wärmewiderstandsberechnung ausgewählt, eine Schicht hinzugefügt und die Materialdaten von dann drei Schichten eingegeben hat. Nachdem der Systemnutzer die Berechnung startet, berechnet das Softwaresystem die Wärmewiderstandsdaten und gibt diese aus.

Ergebnis dieses Tests sind die korrekt berechneten Werte für den Wärmewiderstand. Die Berechnung wurde erfolgreich abgeschlossen.

13.1.2. Vorbedingungen

<Vorbedingung 1> Die Software ist gestartet und einsatzbereit.

<Vorbedingung 2> Der Systemnutzer hat eine zusätzliche Schicht hinzugefügt.

<Vorbedingung 3> Der Systemnutzer hat die Eingabefelder der Wärmewiderstandsberechnung für die drei Schichten mit folgenden Werten befüllt:

- Wanddicke der ersten Schicht: $d_1 = \dots$ cm
- Wanddicke der zweiten Schicht: $d_2 = \dots$ cm
- Wanddicke der dritten Schicht: $d_3 = \dots$ cm
- Wärmeleitfähigkeit der ersten Schicht: $\lambda_1 = \dots$ W/(m*K)
- Wärmeleitfähigkeit der zweiten Schicht: $\lambda_2 = \dots$ W/(m*K)
- Wärmeleitfähigkeit der dritten Schicht: $\lambda_3 = \dots$ W/(m*K)

<Vorbedingung 3> Der Systemnutzer löst die Berechnung aus

13.1.3. Nachbedingungen

<Nachbedingung 1> Die Software hat keinen Fehler gemeldet.

<Nachbedingung 2> Die Software zeigt die berechneten Ergebnisse an.

13.1.4. Benötigte Daten

Als Testdaten werden Werte benötigt, die idealerweise aus realen Aufgaben zur Berechnung des Wärmeübergangs stammen. Bestenfalls Aufgaben mit geprüften selbst berechneten oder

vorgegebenen korrekten Ergebnissen, um den Output der Software entsprechend validieren zu können.

14. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]

14.1. Test Case ID: 005 - Berechnung Wärmewiderstand für vier Schichten mit vorgespeicherten Werten:

14.1.1. Beschreibung

Dieser Test Case evaluiert das Verhalten des entworfenen Softwaresystems, nachdem der Systemnutzer die Funktion der Wärmewiderstands Berechnung ausgewählt, zwei Schichten hinzugefügt und die Materialdaten von dann vier Schichten eingegeben hat. Dabei wird für zwei der vier Schichten der Koeffizient für die Wärmeleitfähigkeit aus vorgespeicherten Materialdaten ausgewählt. Nachdem der Systemnutzer die Berechnung startet, berechnet das Softwaresystem die Wärmewiderstandsdaten und gibt diese aus.

Ergebnis dieses Tests sind die korrekt berechneten Werte für den Wärmewiderstand. Die Berechnung wurde erfolgreich abgeschlossen.

14.1.2. Vorbedingungen

<Vorbedingung 1> Die Software ist gestartet und einsatzbereit.

<Vorbedingung 2> Der Systemnutzer hat zwei zusätzliche Schichten hinzugefügt.

<Vorbedingung 3> Der Systemnutzer hat die Eingabefelder der Wärmewiderstands Berechnung für die vier Schichten mit folgenden Werten befüllt:

- Wanddicke der ersten Schicht: $d_1 = 2 \text{ cm}$
- Wanddicke der zweiten Schicht: $d_2 = 24 \text{ cm}$
- Wanddicke der dritten Schicht: $d_3 = 5 \text{ cm}$
- Wanddicke der vierten Schicht: $d_4 = 1 \text{ cm}$
- Wärmeleitfähigkeit der ersten Schicht: $\lambda_1 = 0,350 \text{ W/(m}^*\text{K)}$
- Wärmeleitfähigkeit der zweiten Schicht: Nutzer wählt als Material Kalksandstein-Mauerwerk [$\lambda_2 = 0,560 \text{ W/(m}^*\text{K)}$]
- Wärmeleitfähigkeit der dritten Schicht: $\lambda_3 = 0,045 \text{ W/(m}^*\text{K)}$
- Wärmeleitfähigkeit der vierten Schicht: Nutzer wählt als Material Kunstharzputz [$\lambda_4 = 0,700 \text{ W/(m}^*\text{K)}$]

<Vorbedingung 3> Der Systemnutzer löst die Berechnung aus

14.1.3. Nachbedingungen

<Nachbedingung 1> Die Software hat keinen Fehler gemeldet.

<Nachbedingung 2> Die Software zeigt die berechneten Ergebnisse an.

14.1.4. Benötigte Daten

Als Testdaten werden Werte benötigt, die idealerweise aus realen Aufgaben zur Berechnung des Wärmeübergangs stammen. Bestenfalls Aufgaben mit geprüften selbst berechneten oder vorgegebenen korrekten Ergebnissen, um den Output der Software entsprechend validieren zu können.

15. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]

15.1. Test Case ID: 006_1 - Berechnung nach unvollständiger Eingabe Waddicke:

15.1.1. Description

Dieser Test Case evaluiert, wie das entworfene Softwaresystem bei unvollständiger Eingabe von Werten in die Eingabefelder durch den Benutzer reagiert. In diesem Fall wird für eine Schicht die Eingabe der Waddicke weggelassen.

Das erwartete Ergebnis ist die Ausgabe eines Fehlers durch die Software.

15.1.2. Pre-conditions

<Vorbedingung 1> Die Software ist gestartet und einsatzbereit.

<Vorbedingung 2> Der Systemnutzer hat eine dritte und eine vierte Schicht hinzugefügt.

<Vorbedingung 3> Der Systemnutzer hat bis auf die Waddicker der Schicht drei alle Eingabefelder der vorhandenen Schichten mit folgenden Werten befüllt:

- Waddicke der ersten Schicht: $d_1 = 2 \text{ cm}$
- Waddicke der zweiten Schicht: $d_2 = 24 \text{ cm}$
- Waddicke der dritten Schicht: d_3 **bleibt leer**
- Waddicke der vierten Schicht: $d_4 = 1 \text{ cm}$
- Wärmeleitfähigkeit der ersten Schicht: $\lambda_1 = 0,350 \text{ W/(m}^*\text{K)}$
- Wärmeleitfähigkeit der zweiten Schicht: $\lambda_2 = 0,560 \text{ W/(m}^*\text{K)}$
- Wärmeleitfähigkeit der dritten Schicht: $\lambda_3 = 0,045 \text{ W/(m}^*\text{K)}$
- Wärmeleitfähigkeit der vierten Schicht: $\lambda_4 = 0,700 \text{ W/(m}^*\text{K)}$

<Vorbedingung 3> Der Benutzer löst die Berechnung aus.

15.1.3. Post-conditions

<Nachbedingung 1> Die Software meldet einen Fehler bei der Überprüfung der eingegebenen Werte, da ein unbefülltes Feld vorhanden ist.

<Nachbedingung 2> Berechnung wird nicht durchgeführt und keine Ergebnisse angezeigt.

<Nachbedingung 3> Die Eingabefelder sind nach der versuchten Berechnung und dem ausgegebenen Fehler nach wie vor mit den bisher eingegebenen Werten befüllt.

15.1.4. Data required

Als Testdaten werden Werte benötigt, die idealerweise aus realen Aufgaben zur Berechnung des Wärmeübergangs stammen. Diese werden dann zum Teil weggelassen.

16. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]

16.1. Test Case ID: 006_2 - Berechnung nach unvollständiger Eingabe Wärmeleitfähigkeit:

16.1.1. Description

Dieser Test Case evaluiert, wie das entworfene Softwaresystem bei unvollständiger Eingabe von Werten in die Eingabefelder durch den Benutzer reagiert. In diesem Fall wird für eine Schicht die Eingabe der Wärmeleitfähigkeit weggelassen.

Das erwartete Ergebnis ist die Ausgabe eines Fehlers durch die Software.

16.1.2. Pre-conditions

<Vorbedingung 1> Die Software ist gestartet und einsatzbereit.

<Vorbedingung 2> Der Systemnutzer hat eine dritte und eine vierte Schicht hinzugefügt.

<Vorbedingung 3> Der Systemnutzer hat bis auf die Waddicke der Schicht drei alle Eingabefelder der vorhandenen Schichten mit folgenden Werten befüllt:

- Waddicke der ersten Schicht: $d_1 = 2 \text{ cm}$
- Waddicke der zweiten Schicht: $d_2 = 24 \text{ cm}$
- Waddicke der dritten Schicht: $d_3 = 5 \text{ cm}$
- Waddicke der vierten Schicht: $d_4 = 1 \text{ cm}$
- Wärmeleitfähigkeit der ersten Schicht: $\lambda_1 = 0,350 \text{ W/(m}^*\text{K)}$
- Wärmeleitfähigkeit der zweiten Schicht: $\lambda_2 = 0,560 \text{ W/(m}^*\text{K)}$
- Wärmeleitfähigkeit der dritten Schicht: λ_3 **bleibt leer**
- Wärmeleitfähigkeit der vierten Schicht: $\lambda_4 = 0,700 \text{ W/(m}^*\text{K)}$

<Vorbedingung 3> Der Benutzer löst die Berechnung aus.

16.1.3. Post-conditions

<Nachbedingung 1> Die Software meldet einen Fehler bei der Überprüfung der eingegebenen Werte, da ein unbefülltes Feld vorhanden ist.

<Nachbedingung 2> Berechnung wird nicht durchgeführt und keine Ergebnisse angezeigt.

<Nachbedingung 3> Die Eingabefelder sind nach der versuchten Berechnung und dem ausgegebenen Fehler nach wie vor mit den bisher eingegebenen Werten befüllt.

16.1.4. Data required

Als Testdaten werden Werte benötigt, die idealerweise aus realen Aufgaben zur Berechnung des Wärmeübergangs stammen. Diese werden dann zum Teil weggelassen.

17. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]

17.1. Test Case ID: 007 - Berechnung mit "leerer Schicht":

17.1.1. Description

Dieser Test Case evaluiert, wie das entworfene Softwaresystem bei einer Eingabe durch den Benutzer reagiert, bei der die Eingabefelder, die eine einzige Wandschicht repräsentieren, gänzlich unbesetzt bleiben.

Das erwartete Ergebnis ist eine Berechnung des Wärmeübergangs ohne die "leere Schicht" oder die Ausgabe eines Fehlers durch die Software. Idealerweise mit einer genaueren Beschreibung der Art des Eingabefelders.

17.1.2. Pre-conditions

<Vorbereitung 1> Die Software ist gestartet und einsatzbereit

<Vorbereitung 2> Der Systemnutzer hat eine dritte und vierte Schicht hinzugefügt.

<Vorbereitung 2> Der Benutzer hat alle Eingabefelder, bis auf alle Felder für Schicht drei mit folgenden Werten befüllt:

- Wandschicht der ersten Schicht: $d_1 = \dots$ cm
- Wandschicht der zweiten Schicht: $d_2 = \dots$ cm
- Wandschicht der dritten Schicht: d_3 **bleibt leer**
- Wandschicht der vierten Schicht: $d_4 = \dots$ cm
- Wärmeleitfähigkeit der ersten Schicht: $\lambda_1 = \dots$ W/(m*K)
- Wärmeleitfähigkeit der zweiten Schicht: $\lambda_2 = \dots$ W/(m*K)
- Wärmeleitfähigkeit der dritten Schicht: λ_3 **bleibt leer**
- Wärmeleitfähigkeit der vierten Schicht: $\lambda_4 = \dots$ W/(m*K)

<Vorbereitung 3> Der Benutzer löst die Berechnung aus.

17.1.3. Post-conditions

<Nachbedingung 1> Die Software meldet einen Fehler bei der Überprüfung der eingegebenen Werte.

<Nachbedingung 2> Berechnung wird nicht durchgeführt und keine Ergebnisse angezeigt.

<Nachbedingung 3> Die Eingabefelder sind nach der versuchten Berechnung und dem

ausgegebenen Fehler nach wie vor mit den Werten befüllt.

17.1.4. Data required

Als Testdaten werden Werte benötigt, die idealerweise aus realen Aufgaben zur Berechnung des Wärmeübergangs stammen.

18. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]

18.1. Test Case ID: 008 - Partielle Nullen:

18.1.1. Description

Dieser Test Case evaluiert, wie das Softwaresystem reagiert, wenn alle Felder vom Benutzer befüllt wurden, einzelne oder mehrere eine Null enthalten. Beispielsweise das Verhalten, wenn

008.1 Wanddicke gleich Null oder

008.2 Übertragungskoeffizient gleich Null ist.

Das erwartete Ergebnis ist die Ausgabe

008.1 t.b.d

008.2 eines Fehlers und idealerweise Beschreibung des Fehlers.

18.1.2. Pre-conditions

<Vorbedingung 1> Die Software ist gestartet und einsatzbereit

<Vorbedingung 2.1> Der Benutzer hat alle Eingabefelder befüllt, bei einer Wanddicke steht der Wert Null

<Vorbedingung 2.2> Der Benutzer hat alle Eingabefelder befüllt, bei einem Übertragungskoeffizient steht der Wert Null

<Vorbedingung 3> Der Benutzer löst die Berechnung aus

18.1.3. Post-conditions

<Nachbedingung 1.1> t.b.d

<Nachbedingung 2.1> t.b.d

<Nachbedingung 1.2> Das Softwaresystem hat einen Fehler bei den eingegebenen Daten gemeldet

<Nachbedingung 2.2> Die Berechnung der Werte wurde nicht durchgeführt, es sind keine Ergebnisse vorhanden

<Nachbedingung 3> Die Eingabefelder sind nach der Berechnung nach wie vor mit den Werten befüllt

18.1.4. Data required

Als Testdaten werden Werte benötigt, die idealerweise aus realen Aufgaben zur Berechnung des

Wärmeübergangs stammen.

19. Projekt Bauphysik Test Case: [Use Case: Reihenberechnung_durchführen]

19.1. Test Case ID: 009 - alle Felder "Null":

19.1.1. Description

Dieser Test Case evaluiert, wie das Softwaresystem reagiert, wenn alle Eingabefelder eine Null enthalten.

Das erwartete Ergebnis ist die Ausgabe...t.b.d

19.1.2. Pre-conditions

<Vorbedingung 1> Die Software ist gestartet und einsatzbereit

<Vorbedingung 2> Alle Eingabefelder sind mit Nullen befüllt

<Vorbedingung 3> Der Benutzer löst die Berechnung aus

19.1.3. Post-conditions

<Nachbedingung 1> t.b.d.

<Nachbedingung 2> t.b.d.

<Nachbedingung 3> Die Eingabefelder sind nach der Berechnung nach wie vor mit den Werten befüllt

19.1.4. Data required

Keine Daten benötigt.

20. Projekt Bauphysik Test Case: [Use Case: Plausibilität_prüfen]

20.1. Test Case ID: 021 - Plausibilität korrekter Daten:

20.1.1. Beschreibung

Dieser Test Case evaluiert das Verhalten des entworfenen Softwaresystems, nachdem der Systemnutzer die Funktion Wärmewiderstandsberechnung ausgewählt hat. Diese Funktion erfordert die Eingabe von Daten. Während der Nutzer seine Daten in die vorgesehenen Felder eingibt, prüft das Softwaresystem diese auf Plausibilität.

Ergebnis dieses Tests ist, dass bei Eingabe sinnvoller Zahlenwerte kein Fehler gemeldet wird. Damit wird die Plausibilitätsprüfung erfolgreich abgeschlossen.

20.1.2. Vorbedingungen

<Vorbedingung 1> Die Software ist gestartet und einsatzbereit.

<Vorbedingung 2> Der Systemnutzer hat die Eingabefelder der Wärmewiderstandsberechnung für zwei Schichten mit folgenden Werten befüllt:

- Wanddicke der ersten Schicht: $d_1 = 20 \text{ cm}$
- Wanddicke der zweiten Schicht: $d_2 = 10,5 \text{ cm}$
- Wärmeleitfähigkeit der ersten Schicht: $\lambda_1 = 0,454 \text{ W/(m}\cdot\text{K)}$
- Wärmeleitfähigkeit der zweiten Schicht: $\lambda_2 = 0,369 \text{ W/(m}\cdot\text{K)}$

20.1.3. Nachbedingungen

<Nachbedingung 1> Die Software hat keinen Fehler gemeldet.

<Nachbedingung 2> Die Software hat die Eingabedaten validiert.

20.1.4. Benötigte Daten

Als Testdaten werden Werte benötigt, die idealerweise aus realen Aufgaben zur Berechnung des Wärmeübergangs stammen. Generell eignen sich für diesen Fall auch zufällig gewählte Zahlenwerte.

21. Projekt Bauphysik Test Case: [Use Case: Plausibilität_prüfen]

21.1. Test Case ID: 022 - Plausibilität inkorrektter Daten:

21.1.1. Beschreibung

Dieser Test Case evaluiert das Verhalten des entworfenen Softwaresystems, nachdem der Systemnutzer die Funktion Wärmewiderstandsberechnung ausgewählt hat. Diese Funktion erfordert die Eingabe von Daten. Während der Nutzer seine Daten in die vorgesehenen Felder eingibt, prüft das Softwaresystem diese auf Plausibilität. In diesem Fall werden vom System nicht erwartete Daten eingegeben.

Ergebnis dieses Tests ist, dass bei Eingabe sinnvoller Zahlenwerte kein Fehler gemeldet wird. Damit wird die Plausibilitätsprüfung erfolgreich abgeschlossen.

21.1.2. Vorbedingungen

<Vorbedingung 1> Die Software ist gestartet und einsatzbereit.

<Vorbedingung 2> Der Systemnutzer will die Eingabefelder der Wärmewiderstandsberechnung für zwei Schichten mit folgenden Werten befüllen:

- Wanddicke der ersten Schicht: $d_1 = 20 \text{ cm}$
- Wanddicke der zweiten Schicht: $d_2 = 10,5 \text{ cm}$
- Wärmeleitfähigkeit der ersten Schicht: $\lambda_1 = 0,454 \text{ W/(m*K)}$
- Wärmeleitfähigkeit der zweiten Schicht: $\lambda_2 = 0,369 \text{ W/(m*K)}$

<Vorbedingung 3> Anstatt für die Dicke der zweiten Wandschicht $d_2 = 10,5 \text{ cm}$ gibt der Nutzer $d_2 = 10,5 \text{ cm}$ ein

21.1.3. Nachbedingungen

<Nachbedingung 1> Die Software meldet einen Fehler.

<Nachbedingung 2> Die Software verhindert die Eingabe des "o".

<Nachbedingung 3> Die Software weist den Nutzer auf eine Fehleingabe hin.

21.1.4. Benötigte Daten

Als Testdaten werden Werte benötigt, die idealerweise aus realen Aufgaben zur Berechnung des Wärmeübergangs stammen. Generell eignen sich für diesen Fall auch zufällig gewählte Zahlenwerte.

22. Design

Dialog

?

×

OK

Cancel

Tab 1

Tab 2

Temp

außen

0,00

▲

▼

°C

Show Graph

Schicht x

Dicke

0,00

▲

▼

Schicht x+1

delete

ADD