



OctoPi

Projekthandbuch
Softwareentwicklungsprojekt SS 2023

Version: 3.0 (13.06.2023)

Auftraggeber: sovanta AG

Verantwortlich:

Thomas Martin

Julian Wernz

Inhaltsverzeichnis

1 Versionsverzeichnis.....	4
2 Einleitung.....	6
2.1 Projekthintergrund.....	6
2.2 Projektziel.....	6
3 Stakeholder.....	7
3.1 Team.....	7
3.1.1 Entwickler.....	7
3.1.2 Designer.....	7
3.2 Auftraggeber.....	8
3.3 Betreuung.....	9
4 Projektmanagement.....	10
4.1 Organisation.....	10
4.1.1 Arbeitszeiten.....	10
4.1.2 Vorgehensmodell.....	10
4.2 Meetings.....	11
4.2.1 Sprint-Planning.....	11
4.2.2 Sprint-Retrospektive.....	12
4.2.3 Sprint-Daily.....	12
4.2.4 Kundensprechstunde.....	12
4.3 Tools.....	13
4.3.1 Jira.....	13
4.3.2 Dokumentenspeicherung (Google Docs).....	14
4.3.3 GitHub.....	14
4.3.4 Discord.....	14
4.3.5 WhatsApp.....	15
4.3.6 Microsoft Teams.....	15
4.4 Rollendefinition.....	15
4.4.1 Projektleiter.....	15
4.4.2 Kundenkontakt.....	15
4.4.3 Dokumentation.....	16
4.4.4 Qualitätssicherung.....	16
4.4.5 Entwicklung und Design.....	16
4.4.6 Moderatorenrolle.....	17
4.5 Risikomanagement.....	17
5 Dokumente / Unterlagen.....	19
5.1 Webseite.....	19
5.2 Protokolle.....	19
5.3 Vorträge.....	19
5.4 Kunden Deliverables.....	20



6 Glossar.....	21
7 Tabellenverzeichnis.....	24

1 Versionsverzeichnis

Dieses Dokument entsteht in einem iterativen Prozess und wird insbesondere an teaminterne Änderungen angepasst und erweitert.

Für die Versionierung dieses Dokuments gelten folgende Regeln, welche teamintern beschlossen wurden:

1. Die Versionsnummer besteht aus zwei Zahlen, getrennt durch einen Punkt.
2. Die erste Zahl der Versionsnummer wird ganzzahlig erhöht, sobald das Dokument für eine Abgabe bereit ist. Die Zahl nach dem Punkt wird zusätzlich auf 0 gesetzt.
3. Die Erhöhung der zweiten Zahl der Versionsnummer erfolgt, sobald der Inhalt um mindestens ein Kapitel erweitert oder der Inhalt bestehender Abschnitte geändert wurde.
4. Die initiale Version eines Dokuments ist 0.0.

Version	Änderung	Datum	Autor(en)
3.0	Feedback von Prof. Knauber angewendet	13.06.2023	Julian Wernz, Thomas Martin
2.0	Änderung von Formulierungen und Qualitätssicherung	23.05.2023	Thomas Martin, Steven Schmitt, Julian Wernz
1.7	Markierung der Veränderungen	22.05.2023	Thomas Martin
1.6	Überarbeitung Risikomanagement, Vorträge, Tools und Rollendefinitionen	19.05.2023	Thomas Martin, Julian Wernz
1.5	Überarbeitung der Einleitung und Meetings	17.05.2023	Julian Wernz
1.4	Überarbeitung des Projekthandbuchs	16.05.2023	Thomas Martin

	aufgrund des Professoren-Feedbacks		
1.3	Überarbeitung des Projekthandbuchs aufgrund des Feedbacks, Moderatorenrolle, hinzufügt	09.05.2023	Jasmin Tschernoch
1.2	Anpassung Termine mit Kunden	08.05.2023	Julian Wernz
1.1	Anpassung der Arbeitszeiten	27.04.2023	Philip Dell
1.0	Kleine Anpassungen und Qualitätssicherung	25.04.2023	Philip Dell, Thomas Martin, Julian Wernz,
0.4	Projektmanagement überarbeitet	24.04.2023	Thomas Martin, Julian Wernz
0.3	Projektmanagement (Tools, Rollenverteilung) hinzugefügt	23.04.2023	Thomas Martin
0.2	Projektmanagement (Organisation, Meetings, Projektplanung) hinzugefügt	21.04.2023	Thomas Martin
0.1	Projektbeschreibung und Stakeholder hinzugefügt	20.04.2023	Thomas Martin
0.0	Inhaltsverzeichnis und Struktur des Dokuments definiert	19.04.2023	Jasmin Tschernoch

Tab. 1 Versionsverzeichnis

2 Einleitung

Das vorliegende Projekthandbuch entsteht im Rahmen des Softwareentwicklungsprojektes im Sommersemester 2023 an der Hochschule Mannheim und dient allen Beteiligten des Projektes als Leitfaden. Dieses Dokument enthält alle projektrelevanten Informationen, bezogen auf die Vorgehensweise, die Teamstruktur und die Projektbeschreibung.

Das Softwareentwicklungsprojekt ist Teil des 4. Semester im Studiengang IB. Dabei arbeiten die Studenten in **Gruppen von fünf bis acht Studierenden** an der Realisierung eines Produktes, dessen Anforderungen von einem externen Kunden gestellt werden.

Über den Projektverlauf hinweg wird dieses Dokument, bei Veränderung der Teamstruktur oder Abläufen, kontinuierlich überarbeitet und gegebenenfalls erweitert.

Fachbegriffe sind beim ersten Auftreten kursiv geschrieben und werden im Glossar erklärt.

2.1 Projekthintergrund

Als Teil des Softwareentwicklungsprojektes im Sommersemester 2023 wurde die Firma **„sovanta AG“¹** (im Folgenden als sovanta adressiert) als externer Kunde gewählt. sovanta ist im Bereich der Softwareentwicklung und UX tätig. Ihre Spezialgebiete umfassen SAP User Experience, Data Science und die *SAP BTP*.

Viele Kunden von sovanta nutzen bereits SAP Software für eine Vielzahl ihrer verschiedenen Geschäftsbereiche. Dennoch wird die SAP BTP, wegen fehlender Aufklärung über die SAP BTP, laut sovanta, nicht ausreichend von ihren Kunden genutzt.

2.2 Projektziel

Das Projektziel besteht darin, ein überzeugendes Konzept und eine prototypische Applikation zu entwickeln, um Besuchern von Messen spielerisch die Möglichkeiten der SAP BTP vorzustellen und potenzielle Kunden von der ‘sovanta *Innovation Factory*’ überzeugen zu können. Die Software soll die Besucher auf den Stand von sovanta aufmerksam machen. Weiterhin besteht das Ziel, das Interesse an dem Unternehmen und deren Dienstleistungen, insbesondere der Innovation Factory for SAP BTP, zu wecken.

¹ <https://sovanta.com/>

3 Stakeholder

3.1 Team

Das Team besteht aus acht Mitgliedern, davon studieren zwei Kommunikationsdesign und die restlichen sechs Informatik. Die interne Kommunikation findet über *Discord* und *WhatsApp* statt (siehe Kap. 4.3.4 und 4.3.5).

Über die folgende E-Mail-Adresse ist das Team erreichbar: octopi-sep@proton.me

3.1.1 Entwickler

Name	Studiengang	Kontakt
Philip Dell	Informatik	philip.dell@stud.hs-mannheim.de
Thomas Martin	Informatik	thomas.martin@stud.hs-mannheim.de
Steven Schmitt	Informatik	steven.schmitt@stud.hs-mannheim.de
Jasmin Tschernoch	Informatik	jasmin.tschernoch@stud.hs-mannheim.de
Timo Wenz	Informatik	timo.wenz@stud.hs-mannheim.de
Julian Wernz	Informatik	julian.wernz@stud.hs-mannheim.de

Tab. 2 Entwickler

3.1.2 Designer

Name	Studiengang	Kontakt
Johanna Neuendorf	Kommunikationsdesign	johanna.neuendorf@stud.hs-mannheim.de
Julia Stumpe	Kommunikationsdesign	julia.stumpe@stud.hs-mannheim.de

Tab. 3 Designer

3.2 Auftraggeber

Der Kontakt zu den sovanta-Mitarbeitern wird über einen E-Mail-Verteiler geregelt. Die durchschnittliche Rückmeldezeit beträgt zwei bis drei Tage.

Name, Vorname	Arbeitsbereich	Kontakt
Frankenbach, Jakob	Head of Development	semesterprojekt2023@sovanta.com
Haas, Larissa	Senior Data Scientist	semesterprojekt2023@sovanta.com
Meiseberg, Alina	Senior Data Scientist	semesterprojekt2023@sovanta.com
Janßen, Nils	Senior Software Engineer	semesterprojekt2023@sovanta.com
Bechberger, Thomas	Senior User Experience Designer	semesterprojekt2023@sovanta.com
Rueda, Rueda	Marketing Specialist	semesterprojekt2023@sovanta.com
Hebestreit, Louise	Marketing Specialist	semesterprojekt2023@sovanta.com

Tab. 4 Auftraggeber

Die hier aufgeführten Termine wurden festgelegt.

Datum	Uhrzeit	Termin	Beschreibung
Jeder zweite Mittwoch, bis 14.06.2023	10 - 15 h	Review Session	Vortrag, was seit dem letzten Review geschehen ist, aktueller Prototyp/ Produktinkrement, Fragen beantworten, Vorstellung der Planung des aktuellen und nächsten Sprints
Jeder zweite Mittwoch, bis 21.06.2023	10 - 14 h	Jour fixe Session	Vortrag über den letzten Sprint, Beantwortung von Fragen und Vorstellung der Planung des aktuellen Sprints
Jeden Freitag, bis 23.06.2023	9:15 - 10:15 h	Kundensprechstunde	Beantwortung der Fragen
05.04.2023	9 - 17 h	Infusion Sessions	Vorträge zum Kennenlernen der SAP BTP
06.04.2023	10 - 15 h	Infusion Sessions	Vorträge zum Kennenlernen der SAP BTP
21.04.2023	10 - 12 h	Sprechstunde Sales	Fragen rund um den Ablauf von

			Kundenmessen werden beantwortet
04.05.2023	9 - 13 h	Kundenreview Session	Vorstellung des Konzeptes ink. Feedback
26.06.2023	9 - 15 h	Abschlusspräsentation	Präsentation und Ehrung des Gewinnerteams
09.06.2023	9:30 - 10:30 h	Kundenreview Session	Vorstellung des aktuellen Projektstands ink. Feedback

Tab. 5 Termine

3.3 Betreuung

Name, Vorname	Rolle	Kontakt
Prof. Dr. Schramm, Wolfgang	Professor	w.schramm@hs-mannheim.de
Prof. Dr. Knauber, Peter	Professor	p.knauber@hs-mannheim.de
D'Angelo, Marco	Lehrbeauftragter (TEW)	m.dangelo@lba.hs-mannheim.de
Haas, Daniel	Tutor	daniel.haas@stud.hs-mannheim.de

Tab. 6 Betreuung

4 Projektmanagement

4.1 Organisation

4.1.1 Arbeitszeiten

Die Informatiker arbeiten sieben Stunden pro Werktag im Raum L303 an der Hochschule Mannheim, mit Ausnahme freitags, an diesem Tag wird die Arbeit online verrichtet. Um 12:00 Uhr machen die Informatiker eine gemeinsame Mittagspause, die 45 Minuten dauert. Nachmittags ist es jedem freigestellt, eine fünfzehnminütige Pause zu machen.

Mittwoch bis Freitag besteht eine Gleitzeitregelung von 8:30 bis 9:30 Uhr. Die Arbeitszeiten sind in der nachfolgenden Tabelle zu finden.

Werktag	Arbeitszeit
Montag	9 bis 17 h (+/- 30 min Gleitzeit)
Dienstag	9:30 bis 17:30 h
Mittwoch	8 bis 18 h (+/- 30 min Gleitzeit)
Donnerstag	8:30 bis 18 h (+/- 30 min Gleitzeit)
Freitag	8:30 bis 18 h (+/- 30 min Gleitzeit)

Tab. 7 Arbeitszeiten

Da sich das Team freitags online trifft, finden jeweils um 10:30 und 16:00 Uhr kurze Besprechungen statt.

Zusätzlich gibt es jeden Tag, mit Ausnahme von Dienstag, die Möglichkeit ab 15:00 online zu arbeiten.

Die Designer arbeiten in der Woche jeweils fünf Stunden. Diese verrichten sie donnerstags und freitags.

4.1.2 Vorgehensmodell

Als Vorgehensmodell nutzen wir einen abgewandelten *Scrum*-Prozess, in dem die Aufgaben des Scrum-Masters und des Product Owners abgeändert wurden (siehe Kap. 4.4).

Damit zu jeder wöchentlichen Präsentation, die mittwochs stattfindet, ein Sprint in seiner Gesamtheit vorgestellt werden kann, ist der Anfang und Ende eines Sprints jeweils auf Dienstag 13 Uhr gelegt.

Jeden Dienstag beginnt wie zuvor erwähnt ein neuer Sprint, worauf wir direkt die Zeiten für die geplanten **Arbeitspakete** innerhalb des Teams schätzen (siehe Kap. 4.2.1). Während des Sprints wird täglich zusätzlich ein Sprint-Daily-Meeting (siehe Kap. 4.2.3) gehalten, bei dem unser Fortschritt innerhalb des Sprints, die Arbeitsteilung für den Tag, als auch Probleme bei der Bewältigung eines Arbeitsschrittes, festgehalten wird. Am Ende eines Sprints findet dienstags um 11:30 Uhr das Sprint-Retro-Meeting (siehe Kap. 4.2.2) statt, in welchem **teaminterne** Probleme aufgegriffen und mögliche Lösungsansätze vorgestellt werden.

4.2 Meetings

4.2.1 Sprint-Planning

Das Sprint-Planning-Meeting ist für 13 Uhr dienstags vor Ort an der Hochschule im Raum H303 angesetzt. In diesem Meeting wird das Ziel, welches in kleinere Arbeitspakete aufgeteilt ist, für den anliegenden Sprint definiert. Die Aufgabenteilung in Arbeitspakete durch das Team und im Folgenden die Arbeitsaufwand-Schätzung der Pakete findet im Sprint-Planning-Meeting statt. Die Schätzung verläuft nach der *Delphi-Fibonacci-Methode*, wobei die Teammitglieder anonym nur Fibonacci-Zahlen für ihre Schätzung nutzen dürfen. Bei Diskrepanz zwischen den einzelnen Schätzungen der Teammitglieder wird innerhalb des Teams diskutiert und sich auf einen Kompromiss in Form einer Fibonacci-Zahl geeinigt. Arbeitspakete, die im letzten Sprint nicht vervollständigt wurden, bekommen eine neue Schätzung. Der Product Owner führt das etwa einstündige Meeting.

Arbeitspakete werden im laufenden Sprint und während dem Sprint Planning Meeting angelegt. Die Arbeitspakete werden mithilfe von *Jira* verfolgt. Jedes Arbeitspaket wird mit den folgenden Inhalten beschrieben:

- Inhalt - Was alles im Arbeitspaket umgesetzt werden soll.
- Begründung - Aus welchen Gründen das Arbeitspaket wichtig ist und warum der Inhalt dazu gehört.

- Nach Fertigstellung dieser Änderung - Was die Resultate nach Beendigung des Arbeitspaketes sind.
- Akzeptanzkriterium - Wann oder woran festgemacht werden kann, ob das Arbeitspaket beendet ist.
- Was nicht enthalten ist - Was explizit nicht in diesem Arbeitspaket enthalten ist.
- Betroffene Entscheidungen - Wurde beim Bearbeiten eine besondere oder wichtige Entscheidung getroffen?

Außerdem ist jedes Arbeitspaket einem **Meilenstein**, in dem mehrere Arbeitspakete gruppiert werden, untergeordnet. Dies kann sich auf eine Aktion beziehen, die in jedem Sprint ähnlich ausgeführt wird (z.B. Organisation), oder einen Produktzustand, der nur einmal während des Projektverlaufs bearbeitet wird (z.B. 'Vertikaler Prototyp'). Die Arbeitspakete werden während des Sprint-Planning-Meetings erstellt und in Jira eingetragen.

4.2.2 Sprint-Retrospektive

Dienstags um 11:30 beginnt das Sprint-Retro-Meeting. Es werden alle teaminternen Probleme, die im letzten Sprint aufgetreten sind, aufgelistet und besprochen. Teammitglieder stellen Lösungsansätze vor, mit denen unter anderem die Kommunikation und die angesprochenen Probleme in Zukunft verbessert werden können.

Die Retro dauert ca. 30 Minuten, kann aber auch, wenn mehr Zeit benötigt wird, in der Mittagspause weitergeführt werden, da die Teammitglieder immer zusammen in der Hochschulmensa essen.

4.2.3 Sprint-Daily

Im Sprint-Daily-Meeting stellt jedes Teammitglied kurz seinen vorherigen Arbeitstag vor. Es wird außerdem geklärt, ob sich jeder der Arbeitsteilung für den Tag bewusst ist und ob jemand Probleme bei der Bewältigung einer Aufgabe hat.

Das Daily-Meeting wird jeden Tag um 10 Uhr gehalten und ist maximal 15 Minuten lang.

4.2.4 Kundensprechstunde

Es wird einmal in der Woche eine Kundensprechstunde von der sovanta angeboten, in der Teams ihre Ideen vorstellen können und Fragen stellen. An der Sprechstunde beteiligen sich diejenigen, die für den Kundenkontakt verantwortlich sind. Danach stellen diese die Ergebnisse der Sprechstunde dem Rest des Teams vor und es wird besprochen, wie wir das Kundenfeedback in unser Projekt einbauen können.

Die Sprechstunde beginnt um 9.15 Uhr freitags und dauert meist eine Stunde.

4.3 Tools

Jeder aus dem Team hat Zugriff auf die aufgeführten Tools. Ansprechpartner, um Zugang zu bekommen, sind Philip Dell und Julian Wernz.

4.3.1 Jira

Wir nutzen Jira, um unsere Arbeitspakete zu verteilen und unseren Fortschritt zu verfolgen. Bei den Sprint-Planning-Meetings entscheiden wir, welche Pakete in den aktuellen Sprint aufgenommen werden. Auf Jira gibt es zu jedem Paket eine detaillierte Beschreibung, was alles ein Paket beinhaltet und eine *DOD* (Definition of Done). Außerdem erlaubt Jira, mithilfe des Add-ons '*Agile Poker*', den Arbeitsaufwand für ein Arbeitspaket, nach der Delphi-Fibonacci-Methode, zu schätzen. Jedes Paket im aktuellen Sprint hat einen von vier Zuständen, welche unterhalb aufgelistet sind:

- 'ZU ERLEDIGEN' : Das Arbeitspaket wurde noch nicht angefangen, sollte aber bis zum Sprintende bearbeitet sein. Beim Sprintanfang sind alle Pakete in diesem Zustand, auch wenn ein Paket im vorherigen Sprint nicht fertiggestellt wurde. Diese werden bei Sprintanfang überarbeitet und bekommen eine neue Schätzung.
- 'IN ARBEIT' : Ein Arbeitspaket, an dem bereits gearbeitet wurde, aber noch nicht fertiggestellt werden konnte, ist in diesem Status.
- 'REVIEW' : Nach der Fertigstellung eines Arbeitspaketes, wird es in den 'Review'-Zustand gestellt. Danach muss ein anderes Teammitglied, welches nicht am Arbeitspaket gearbeitet hat, prüfen, ob die gemachte Arbeit mit dem Akzeptanzkriterium des Arbeitspakets übereinstimmt. Falls sie übereinstimmen, wird das Arbeitspaket als 'FERTIG' eingestuft, falls nicht, wird es noch einmal überarbeitet und mit 'IN ARBEIT' gekennzeichnet.
- 'FERTIG' : Ein fertiges Arbeitspaket stimmt mit dessen DOD überein.. Ein als überflüssiges erwiesenes Arbeitspaket wird mit null Arbeitsstunden protokolliert und auf 'FERTIG' gestellt.

Um den Arbeitsaufwand für jedes Paket zu messen, benutzen wir 'Clockify'. Clockify ist eine Zeiterfassungssoftware, mit der wir unsere Zeiten zu den jeweiligen Arbeitspaketen protokollieren können. Die Zeiten lassen sich entweder live messen oder manuell eintragen.

Ein weiteres Add-on, welches wir nutzen, ist 'Epic Sum Up'. Mit diesem Tool lassen sich eingetragene Zeiten summieren und vergleichen und somit auch die protokollierten Arbeitsstunden einfacher überwachen.

4.3.2 Dokumentenspeicherung (*Google Docs*)

Google Docs wird genutzt, um unsere verschiedenen Dokumente untereinander auszutauschen. Das genannte Textverarbeitungsprogramm ist unser Speicherort für Präsentationsfolien, Feedback, Pflichtenheft, Projekthandbuch, Protokolle und Risikomanagement.

Außerdem messen wir unsere Motivation täglich mithilfe eines Google Forms, welches die Ergebnisse in einer Excel Tabelle speichert. Die Abstimmung erfolgt anonym und wird auf einer Skala von 1 bis 5 gemessen.

4.3.3 GitHub

Für die Versionskontrolle nutzen wir *GitHub*. Es gibt eine Organisation, 'OctoPi-Organization', auf der zwei Repositories liegen. Das eine ist für das Hosten der Website. Das andere ist für das Hauptprojekt.

Link: <https://github.com/OctoPi-Team>

4.3.4 Discord

Wir nutzen *Discord* als Hauptkommunikationsplattform für die teaminterne Kommunikation, inklusive des Tutors. Auf unserem eigenen Server sprechen wir uns über Teamaktivitäten ab, die Designer haben für ihre Ressourcen einen eigenen Kanal. Falls das Team von zu Hause aus arbeitet, was beispielsweise freitags der Fall ist, treffen sich alle Mitglieder auf dem Discord Server.

Täglich schickt ein Bot eine Erinnerung in den dafür angelegten Kanal für die Motivationsabstimmung.

4.3.5 WhatsApp

WhatsApp wird über das laufende Projekt als Kommunikationsplattform für kleinere organisatorische Absprachen genutzt. Diese betreffen beispielsweise projektexterne Treffen, das Zuspätkommen einzelner Teammitglieder oder um die Designer schneller und besser zu erreichen.

4.3.6 Microsoft Teams

Bei Besprechungen mit dem Kunden, für die der E-Mail-Kontakt nicht ausreicht, wird Microsoft Teams für Online-Meetings genutzt. Dies bezieht sich auf die wöchentliche Kundensprechstunde (siehe Kap. 4.2.4) als auch auf Kunden-Review-Meetings (siehe Kap. 5.3).

4.4 Rollendefinition

Innerhalb des Teams gibt es eine Rollenverteilung. Jedes Mitglied ist einer Rolle zugewiesen, wobei für jede Rolle ein oder zwei Backups festgelegt werden. Während der Sprint-Retrospektive passt das Team bei Bedarf die Rollenverteilung als Ergebnis von neuen Erkenntnissen an.

4.4.1 Projektleiter

Der Projektleiter ist verantwortlich für die Leitung der Team-Meetings, Aufgaben zu delegieren und Aufgaben in Arbeitspakete aufzuteilen. Gerät das Projekt in Verzug, ist es die Aufgabe des Projektleiters, Pakete neu zu priorisieren und neue Maßnahmen zu ergreifen, damit das Projekt sich wieder in dem Projektplan befindet. Hinzu kommt, dass der Projektleiter die Verantwortung für den Sprint Backlog trägt, sodass die Arbeitspakete auf Jira zu jeder Zeit verständlich sind und die protokollierten Arbeitszeiten mit den geplanten Stunden nicht extrem abweichen.

Hauptverantwortlicher: Thomas Martin

Backup: Philip Dell

4.4.2 Kundenkontakt

Damit während des Projektverlaufs zwischen den Vorstellungen des Kunden und des Projektziels keine Diskrepanzen entstehen, ist es die Aufgabe des Kundenkontaktes, den Kunden immer über den aktuellen Projektstatus zu informieren. Der Kundenkontakt bereitet

die Fragen für die Kundensprechstunden freitags vor und repräsentiert dort das Team. Die Verantwortlichen pflegen via E-Mail den Kontakt mit sovanta.

Hauptverantwortlicher: Jasmin Tschernoch

Backup: Timo Wenz

4.4.3 Dokumentation

Die unten genannten Teammitglieder mit der Dokumentationsrolle sind verantwortlich für die kontinuierliche Erstellung von Protokollen. Es wird täglich in einem Protokoll erfasst, welche Arbeit für den Tag geplant war, verrichtet wurde und welche Arbeit danach noch offen ist. Dazu wird noch festgehalten, was während dem Sprint-Daily-Meeting besprochen wurde. Außerdem ist es deren Aufgabe, abzugebende Dokumente, wie z.B. Präsentationsfolien, rechtzeitig auf Moodle und unserer Webseite hochzuladen.

Hauptverantwortlicher: Philip Dell

Backup: Steven Schmitt

4.4.4 Qualitätssicherung

Der Qualitätsbeauftragte trägt die Verantwortung, dass alle Abgaben der Kunden oder Professoren ihren Anforderungen entsprechen. Dazu gehören zum Beispiel die rechtzeitige Abgabe für die Jour fixe und Reviews und die Dokumente (Projekthandbuch, Anforderungsspezifikation, Architekturdokument). Die Zuständigen achten unter anderem auf Rechtschreibfehler und auf Vollständigkeit des Inhaltes. Der Prototyp wird vor den Vorträgen auf Lauffähigkeit geprüft. Weiterhin ist es die Aufgabe des Qualitätsbeauftragten, Checklisten für wichtige Abgaben zu erstellen, mit denen man die generelle Richtigkeit von Dokumenten sicherstellen kann. Diese Checklisten befinden sich in einem Ordner auf dem teaminternen Google Docs.

Hauptverantwortlicher: Steven Schmitt

Backup: Thomas Martin

4.4.5 Entwicklung und Design

Da der aktuelle Projektverlauf Entwicklerrollen Tester und Software Designer verlangt, werden diese vergeben. Für Aufgaben der technischen Seite, dazu gehören beispielsweise die Softwarearchitektur oder die Programmierung, tragen die Designer keine Verantwortung. Die Designer haben die Verantwortung, dass die Benutzeroberflächen des Endprodukts den 'User Experience Heuristiken' der sovanta entsprechen.

Verantwortung: Philip Dell, Jasmin Tschernoch, Julian Wernz, Timo Wenz

4.4.6 Moderatorenrolle

Die Einführung der Moderatorenrolle ist Ergebnis des Teamentwicklung-Workshops. Die Hauptaufgabe des Moderators besteht darin, bei Meetings auf die Zeit zu achten und das Team bei Themenabschweifung auf das angesetzte Thema zurückzubringen.

Hauptverantwortung: Jasmin Tschernoch

Backup: Julian Wernz

4.5 Risikomanagement

In dem dafür vorgesehenen Google Docs Spreadsheet „Risikomanagement“ sind bisher alle gefundenen Risiken aufgelistet. In diesem Dokument ist zu jedem Risiko noch eine Beschreibung, die präventiven Maßnahmen, ein messbarer Indikator, als auch die reaktiven Maßnahmen, für den Fall, dass das jeweilige Risiko eingetreten ist, formuliert. Außerdem besitzt jedes Risiko eine Risikobewertung, die sich aus der Auswirkung auf das Projekt und der Eintrittswahrscheinlichkeit ergibt. **Diese werden über den Projektverlauf von den Teammitgliedern gefunden, definiert und in das Spreadsheet eingetragen.**

Jeden Dienstag, nachdem der nächste Sprint geplant wurde, ist es die Aufgabe der Präsentierenden, Risiken, die für den nächsten Sprint relevant sind, aus dem bestehenden Spreadsheet auszuwählen. **Diese werden dann auch während der Präsentation am Mittwoch vorgestellt.**

		Auswirkung			
		Unbe- deutend	Gering	Kritisch	Existen- ziell
Eintrittswahrscheinlichkeit	Unwahr- scheinlich				
	Selten				
	Möglich				
	Sehr wahr- scheinlich				

Risikomatrix

Eintrittswahrscheinlichkeit:

- Unwahrscheinlich: Es ist unwahrscheinlich, dass das Risiko eintritt.
- Selten: Das Risiko tritt selten auf.
- Möglich: Es besteht eine realistische Möglichkeit, dass das Risiko eintritt.
- Sehr wahrscheinlich: Es ist sehr wahrscheinlich, dass das Risiko eintritt.

Auswirkung:

- Unbedeutend: Die Auswirkungen des Risikos sind gering oder kaum wahrnehmbar.
- Gering: Die Auswirkungen des Risikos sind spürbar, aber nicht schwerwiegend.
- Kritisch: Die Auswirkungen des Risikos sind schwerwiegend und haben Auswirkungen auf den Projektverlauf.
- Existenziell: Das Risiko hat katastrophale Auswirkungen auf den Projektverlauf und stellt eine existenzielle Bedrohung für das Endprodukt dar.

5 Dokumente / Unterlagen

5.1 Webseite

Die Octopi-Webseite bietet eine grundlegende Übersicht für die Professoren und Tutoren, um vorab Informationen über das Team zu erhalten. Sie wird über Github-Pages gehostet und mit den Sprachen *Javascript*, *CSS* und *HTML 5* erstellt worden.

Die Website besteht aus vier verschiedenen Seiten:

1. Das Gruppenbild und die Namen der Mitglieder sind auf der Homepage zu finden.
2. Die Seite „Über uns“ sagt aus, für welche Firma das Team arbeitet und wann die Arbeitszeiten sind.
3. Unter „Kontakt“ findet man die Kontaktdaten sowie eine direkte Verknüpfung, um direkt eine Mail an das Team zu senden.
4. Auf der Dokumentenseite sind die Dokumente mit dem Download für unsere Jour fixe und Review Folien und Steckbriefe aufgelistet. So können sich Besucher der Webseite einen besseren Eindruck von der Arbeit und den Fähigkeiten des Teams verschaffen.

Link zu der Seite: <https://octopi-team.github.io/OctoPi-Website/index.html>

5.2 Protokolle

Das Team führt täglich Protokolle, welche die relevanten Ereignisse des Tages enthalten. Jedes Protokoll enthält das Datum, den Namen des Protokollanten, sowie Informationen darüber, ob die Arbeit online oder vor Ort durchgeführt wurde. Darüber weist das Protokoll eine Agenda auf, indem alle besprochenen Themen sowie geplanten Aktivitäten aufgelistet sind.

5.3 Vorträge

Das Team hält jeden Mittwoch einen Vortrag, in dem über den Projektstand berichtet wird. Es gibt zwei verschiedene Arten von Vorträgen, die regelmäßig gehalten werden. Der kürzeste Vortrag ist das „Jour fixe“. Dieser berichtet Professoren und Tutoren über die aktuelle Sprintplanung, gibt einen Rückblick auf den vorherigen Sprint und nennt Risiken, die

für den aktuellen oder nächsten Sprint relevant sind. Dieser Vortrag sollte nicht länger als fünf Minuten dauern.

Die zweite Vortragsart ist das „Review“, in dem alle vorherigen „Jour fixe“, seit dem letzten Review zusammengefasst werden. Hierbei wird auch die Planung bis zum nächsten Review sowie der aktuelle Prototyp präsentiert.

Die Verantwortlichen für die Dokumente (siehe Kap. 4.4.3 Dokumentation) laden die Folien sowie abzugebende Dokumente dienstags vor 17 Uhr auf Moodle und auf unserer Webseite hoch.

Damit die Produktinkremente nicht zu weit von den Anforderungen des Kunden abweichen, werden außerdem auch Kunden-Review-Vorträge gehalten, wo das Team ihren Fortschritt jeweils den sovielen Mitarbeitern vorstellt. Der Schwerpunkt der Vorträge bezieht sich auf die Implementierung und das erarbeitete UX-Konzept.

Diese Reviews haben keine festen Termine und werden unabhängig von den Professoren mit dem Kunden geplant.

5.4 Kunden Deliverables

Die Auslieferung der Deliverables an den Kunden werden vollständig im Anforderungsspezifikation-Dokument beschrieben (siehe Kap.8 Auslieferung Anforderungsspezifikation-Dokument v2 OctoPi).

6 Glossar

Begriff	Definition
Backend	Ein Server, der für die Maschineninteraktion gedacht ist. Es findet keine direkte Interaktion vom Benutzer zum Backend statt.
CDS	SAP Core Data Services https://cap.cloud.sap/docs/cds/
Clockify	eine kostenlose App zum Zeiterfassen https://clockify.me/
CSS	Cascading Stylesheet
Delphi-Fibonacci-Methode	In der Agilen Softwareentwicklung, beschreibt die Delphi-Fibonacci-Methode eine Art des Planungs Pokers, mit dem man den Arbeitsaufwand eines Arbeitsschrittes schätzen kann. https://de.wikipedia.org/wiki/Scrum#Planungspoker
Discord	Eine Plattform, die zur Kommunikation genutzt wird. https://discord.com/
DOD	'Definition of Done' : Das Akzeptanzkriterium, welches beschreibt, wann ein Arbeitspaket bewältigt ist.
Epic Sum Up	Epic Sum Up ist ein Add-on für Jira, mit der man eingetragenen Zeiten einfach summieren kann
FA	Funktionale Anforderungen
Frontend	Teil der Software, welcher mit dem Nutzer interagiert und auf dem Rechner des Nutzers läuft.
Github	Softwareversionierungsservice, welcher Git benutzt und weitere Zusatzfunktionen bietet. https://github.com/

Google Docs	Online-Service, welcher Mehrbenutzerbetrieb für Word-Dokumente, Sheets und Slides anbietet. https://docs.google.com
Human Centered Design	reale Personen sind in der Mitte der Entwicklung
Innovation Factory	Die Innovation Factory besteht aus sechs Bereichen und wird mit den vorhandenen Features der BTP realisiert. Die sechs Bereiche lauten Design, Engineering, Produktion, Parts, Shipment und Monitoring. (https://sovanta.com/innovation-factory-for-sap-btp/)
JavaScript	Eine Programmiersprache, die für den Browser entwickelt wurde und mittlerweile auch mithilfe von Node.js im Backend laufen kann.
Jira	Online Projektmanagement Tool, welches primär für Softwareentwicklung eingesetzt wird. https://www.atlassian.com/software/jira
NFA	Nichtfunktionale Anforderungen
Node.js	Eine plattformübergreifende Javascript Laufzeitumgebung. https://nodejs.org
R	Randbedingung
React.js	Frontend Framework basierend auf Node.js Javascript. https://react.dev/
SAP BTP	SAP Business Technology Platform: „Eine Technologieplattform, die Daten und Analysen, künstliche Intelligenz, Anwendungsentwicklung, Automatisierung und Integration in einer einheitlichen Umgebung vereint.“ (https://www.sap.com/germany/products/technology-platform/what-is-sap-business-technology-platform.html)
Scrum	Scrum ist ein agiler Software Prozess https://www.scrum.org/

SEP	Softwareentwicklungsprojekt
Three.js	JavaScript Frontend Library für 3D Rendering. https://threejs.org/
UX	Nutzererfahrung - Die Erfahrung, die ein Nutzer mit einem Produkt macht.
VS Code	Visual Studio Code ist ein kostenloser Quelltext-Editor. https://code.visualstudio.com/
Web-API	Application Programming Interface, welches Interaktionen über http Requests bietet.
WhatsApp	Internet Nachrichten App für Mobilgeräte. https://www.whatsapp.com/

7 Tabellenverzeichnis

Tab. 1 Versionsverzeichnis.....	5
Tab. 2 Entwickler.....	7
Tab. 3 Designer.....	7
Tab. 4 Auftraggeber.....	8
Tab. 5 Termine.....	9
Tab. 6 Betreuung.....	9
Tab. 7 Arbeitszeiten.....	10