

Abschlussprüfung Winter 2023

Ausbildungsberuf

Fachinformatiker/Fachinformatikerin (VO 2020)

Fachrichtung: Anwendungsentwicklung

Prüfungsbezirk

Braunschweig 1201 FIA 4 (AP T2V1)

Prüfungsteilnehmer

Herr Julian Thiele

Identnummer: 734117

Ausbildungsbetrieb

Bredex GmbH

Lindentwete 1

38100 Braunschweig

Projektbetreuer

Herr Christian Rucinski

Thema der Projektarbeit

Digitalisierung des internen Onboarding Prozesses neuer Mitarbeiter*innen

Inhaltsverzeichnis

1	Einleitung	1
1.1	Ausbildungsbetrieb	1
1.2	Projektumfeld	1
1.3	Projektabgrenzung	1
1.4	Projektziel	1
1.5	Abweichungen vom Projektantrag	1
1.6	weitere Anmerkungen	2
2	Projektplanung	2
2.1	Identifizierung der Arbeitspakete	2
2.2	Zeitplan	2
2.3	Ressourcenplanung	3
2.4	Kostenplanung	3
2.5	Nutzwertanalyse	3
3	Analysephase	4
3.1	Anforderungsgespräch	4
3.2	IST-Zustand Analyse	5
3.3	SOLL-Konzept	5
4	Designphase	6
4.1	Benutzeroberfläche	6
4.2	Datenbank	6
5	Implementierungsphase	7
5.1	Frontend	7
5.2	Backend	8
6	Testphase	9
7	Abnahme	10
8	Fazit	10
8.1	Soll/Ist-Vergleich	10
8.2	Lessons learned	11
	Glossar	12
	Tabellenverzeichnis	14
	Abbildungsverzeichnis	14

Listings	14
A genutzte Ressourcen	19
Benutzerdokumentation	20
Projektantrag	28
Bestätigung über die durchgeführte Projektarbeit	32

1 Einleitung

1.1 Ausbildungsbetrieb

Die BREDEX GmbH wurde im Jahr 1987 in Braunschweig gegründet. Den Schwerpunkt bildet die individuelle Softwareentwicklung, BREDEX führt jedoch auch Beratungen zu Datenschutz, Datensicherheit, Qualitätssicherung von Software und Schulungen durch. Die BREDEX GmbH besitzt, mit ihrer Tochterfirma BREDEX HUNGARY KFT. zusammen, ca. 200 Mitarbeiter. Davon sind ungefähr 12 Mitarbeiter Auszubildenden und dualen Studenten.

1.2 Projektumfeld

Das Projekt wird unter der Aufsicht von Herr Christian Rucinski in den Geschäftsräumen der BREDEX GmbH entwickelt. Bei der Anwendung handelt es sich um ein BREDEX internes Tool. Sie soll für den Onboardingprozess neuer Mitarbeiter eingesetzt werden und den sogenannten BREDEX-Buddy, einen Ansprechpartner neuer Mitarbeiter, bei seiner Arbeit unterstützen.

Das Frontend wird mit dem Framework Vue, bei welchem die Scriptsprache TypeScript Anwendung findet, geschrieben. Als Backend wird Firebase als "Backend As a Service" genutzt, in welche die NoSQL-Datenbank Firestore integriert ist.

1.3 Projektabgrenzung

Durch die Fachinformatikerausbildungsverordnung wird das Projekt zeitlich auf 80 Stunden begrenzt.

Die Anwendung ist ein alleinstehendes Tool und wird daher in kein Projekt eingegliedert. Daher gibt es keine weiteren Beschränkungen, auf die geachtet werden muss.

1.4 Projektziel

Um neuen Mitarbeitern den Start in das Unternehmen zu vereinfachen, stellt BREDEX GmbH neuen Mitarbeitern den sogenannten BREDEX-Buddy zur Verfügung. Um diesen in Zukunft die Arbeit zu vereinfachen und ihnen die Aufgaben klarzustellen, soll es in Zukunft eine Webanwendung geben. Diese soll eine Checkliste mit den verschiedenen Aufgaben zur Verfügung stellen, die ein BREDEX-Buddy erledigen soll, sowie eine Rangliste, um die Einarbeitung mithilfe von Gamification interessanter zu gestalten.

1.5 Abweichungen vom Projektantrag

Es gibt keine Abweichungen vom Projektantrag.

1.6 weitere Anmerkungen

In der Dokumentation wird die männliche Form von Mitarbeitern, Kollegen, Kunden, etc genutzt. Gemeint wird jedoch m/w/d.

2 Projektplanung

Bei der Projektplanung werden die benötigten Arbeitspakete identifiziert und geplant. Zudem werden Kosten und Ressourcen zunächst abgeschätzt.

2.1 Identifizierung der Arbeitspakete

Zu Beginn fand ein Anforderungsgespräch statt. Hierbei wurden die Projektziele definiert, damit es keine Differenzen im Verständnis dieser gibt. Außerdem wurde definiert, welche Anforderungen erfüllt werden müssen.

Im Anschluss folgte die Analysephase. In dieser Phase wurde der IST-Zustand analysiert und der Soll-Zustand niedergeschrieben, welcher sich aus den vorab definierten Anforderungen ergab. Dabei wurden die Anforderungen verfeinert.

Auf den Verfeinerungen der Analysephase baute die folgende Designphase auf. Hierbei wurde die Benutzeroberfläche geplant und ein Datenmodell erstellt. Dadurch konnten benötigte Ressourcen genauer abgeschätzt werden.

Daraufhin folgte die Entwicklung der Anwendung, bei der die Ergebnisse der vorherigen Arbeitspaketen umgesetzt wurden. Um mögliche Fehler während der Implementierung frühzeitig erkennen und beheben zu können, fanden Tests statt.

Am Ende der Entwicklungsphase wurde ein Abnahmegespräch mit dem Projektbetreuer durchgeführt. In diesem Gespräch wurde die Anwendung mit den Anforderungen verglichen und entschieden, ob diese erfüllt wurden.

Im letzten Arbeitsschritt wurde die Dokumentation des Projektes erstellt.

2.2 Zeitplan

Das vorliegende Projekt wurde im Zeitraum vom 22.09.2023 bis zum 22.11.2023 bearbeitet. Für die unter 2.1 genannten Arbeitspakete wurde folgend der Zeitplan (Tabelle 1) erstellt.

Arbeitspaket	Dauer (in Stunden)
Anforderungsgespräch	3
Analysephase	6
Planungsphase	6
Designphase	6
Umsetzung	40
Test und Abnahme	7
Dokumentation	12
Gesamt	80

Tabelle 1: Zeitplan

2.3 Ressourcenplanung

Um die Kosten so gering wie möglich zu halten, wurde nach Möglichkeit nur kostenlose oder bereits vorhandene Software, sowie Hardware genutzt. Als Arbeitsgerät wurde durch die BREDEX GmbH ein Laptop mit Windows 10 und der Entwicklungsumgebung VisualStudio Code bereitgestellt. Des Weiteren wurde Firebase als Backend-As-A-Service mit Firestore als Datenbank verwendet. Git wird als Versionsverwaltung eingesetzt.

Eine Liste mit allen genutzten Ressourcen befindet sich im Anhang.

2.4 Kostenplanung

Für den Prüfungsteilnehmer wurde ein Stundenlohn von 9 EUR angesetzt. Ihm standen für die Bearbeitung 80 Stunden zur Verfügung. Der Projektbetreuer führte das Anforderungsgespräch durch und stand dazu noch für weitere Fragen während der Bearbeitung zur Verfügung. Daher wurden für ihn 6 Stunden angesetzt, bei einem geschätzten Stundenlohn von 45 EUR. Da ansonsten nur vorhandene oder kostenlose Ressourcen genutzt wurden, ergaben sich keine weiteren Lizenz- oder Nutzungskosten.

Als Gesamtkosten ergaben sich hiermit 990 EUR, wie im Kostenplan (s. Tabelle 2) festgehalten.

2.5 Nutzwertanalyse

Das Ziel der Anwendung liegt darin, den Prozess der Einarbeitung neuer Mitarbeiter zu erleichtern. Unter diesem Gesichtspunkt wird im Folgenden der alte Prozess mit dem neuen Prozess, der die hier entwickelte Anwendung nutzt, verglichen. Die Nutzwerte wurden ermittelt und im Anhang in Tabelle 3 festgehalten.

Am Wichtigsten ist der Aspekt der Benutzerfreundlichkeit (Usability). Im alten Prozess mussten alle Aufgaben per Hand aus einer Word-Datei in eine eigene Liste kopiert werden und bei Erfüllung manuell durch Emojis abgehakt oder aus der eigenen Liste gelöscht werden. Außerdem mussten

alle Aufgaben für nicht-deutschsprachige Mitarbeiter vom BREDEX-Buddy selbst übersetzt werden. In der Anwendung werden die Aufgaben über die Achievements direkt in einer Liste gesammelt und können durch einen einfachen Klick abgehakt werden. Die Sprache kann über einen Button in der Anwendung einfach umgestellt werden.

Ein weiterer Wert der Anwendung liegt in der Vereinheitlichung des Prozesses. Derzeit gibt es unterschiedliche Listen mit verschiedenen Aufgaben. Je nachdem, welche Liste der BREDEX-Buddy nutzt, werden unterschiedliche Aufgaben erledigt.

Mit der Anwendung gibt es eine einheitliche Liste an Achievements, die von den BREDEX-Buddys genutzt wird. Diese Liste kann jederzeit von zuständigen Personen angepasst werden, sollte es Änderungen im Einarbeitungsprozess geben.

Die Motivation die Aufgaben zu erledigen steigt vorraussichtlich mit der Anwendung deutlich an. Der Vorgang, die Aufgaben zu suchen, und alles manuell kopieren zu müssen ist umständlich, was die Motivation senkt, sich darum zu kümmern. Mit der Anwendung müssen die Aufgaben nicht mehr gesucht werden, sondern sind einfach in der Applikation einsehbar. Außerdem erhält die Einarbeitung durch die Rangliste eine Gamification. Dadurch kann die Motivation weiterhin gesteigert werden, da man sich mit seinen Kollegen messen kann und ein kleiner Wettbewerb entsteht.

Schließlich wird der Aspekt der Sicherheit betrachtet. Die alten Dateien an Aufgaben lagen in einem zentralen Cloudspeicher und konnten von allen Mitarbeitern bearbeitet oder neu erstellt werden. Mit der neuen Anwendung können nur noch zuständige Personen auf die Liste zugreifen, um diese zu bearbeiten.

3 Analysephase

In der Analysephase werden die Anforderungen definiert, sowie der IST-Zustand analysiert und das SOLL-Konzept entwickelt. Mit den Erkenntnissen aus dieser Phase kann im Anschluss die Struktur der Anwendung erstellt werden.

3.1 Anforderungsgespräch

Als Erstes wurde ein Anforderungsgespräch mit dem Kunden durchgeführt. In dem Gespräch wird der IST-Zustand ermittelt und das SOLL-Konzept definiert. Außerdem wurden die vom Kunden gewünschten Features der Anwendung übermittelt. Die wichtigsten Features der Anwendung sind:

- Achievements, die abgehakt werden können
- Rangliste, um sich mit Kollegen vergleichen zu können
- Als Administrator Achievementliste bearbeiten können
- Mehrsprachigkeit in Deutsch und Englisch

Die Punkte wurden in dem Gespräch noch weiter konkretisiert, um die Anwendung im Anschluss möglichst nach den Wünschen des Kunden implementieren zu können.

3.2 IST-Zustand Analyse

Damit neue Mitarbeiter der BREDEX GmbH besser in das Unternehmen finden, wird ihnen ein Ansprechpartner, der sogenannte BREDEX-Buddy, an die Seite gestellt. Dieser soll dem Mitarbeiter das Unternehmen näherbringen. Allerdings ist nicht genau klar, was die Aufgaben des BREDEX-Buddys beinhalten. Es gibt unterschiedliche Orte, an denen man Listen mit verschiedenen Inhalten findet. Dadurch führen die BREDEX-Buddys ihre Aufgabe unterschiedlich aus. Daraus resultiert, dass Mitarbeiter unterschiedlich eingearbeitet werden und es besteht die Möglichkeit, dass Punkte vergessen oder vernachlässigt werden. Dadurch erfahren neue Mitarbeiter beispielsweise nicht von Angeboten, die die BREDEX GmbH zur Verfügung stellt, bekommen Ereignisse nicht mit oder lernen ihre Kollegen nicht kennen, weshalb sie sich schwerer in das Unternehmen einfinden, wodurch auch die Produktivität leiden kann.

Zudem ist der Prozess sehr umständlich. Die Aufgaben müssen aus der Datei in eine eigene kopiert werden, um sie abhaken zu können. Das Abhaken der erledigten Aufgaben erfolgt händisch über Löschen der Einträge, Einfügen eines Haken-Emojis oder Ähnlichem.

3.3 SOLL-Konzept

Um in Zukunft dafür zu sorgen, dass alle BREDEX-Buddys ihre Aufgaben auf die gleiche Art und Weise erledigen und den Prozess im Allgemeinen zu vereinfachen, soll eine Webanwendung programmiert werden. In der Anwendung soll es eine Checkliste an Achievements geben, die neue Mitarbeiter im Verlauf ihrer Einarbeitung mit ihren BREDEX-Buddys zusammen abarbeiten können. Diese Achievements sollen verschiedene Aufgaben beinhalten, wie beispielsweise bestimmte Personen aus dem Unternehmen kennen zu lernen oder sich über Angebote oder Abläufe zu informieren. Manche Achievements beinhalten zudem Referenzen zu Insidern aus der Unternehmenskultur, wodurch man diese besser kennen lernt.

Zudem soll es eine Rangliste geben, in der sich Kollegen, die zur selben Zeit angefangen haben, gegenseitig sehen können. Diese Gamification erhöht die Motivation, Achievements zu erledigen. Dadurch lernen neue Mitarbeiter das Unternehmen schneller kennen und finden sich so schneller zurecht und lernen ihre Kollegen schneller kennen.

Um die Achievements aktualisieren zu können, soll es noch eine Seite geben, bei der ausgewählte Personen neue Achievements anlegen, bearbeiten oder auch löschen können. Dadurch können die Achievements besser aktuell gehalten werden und bei Änderungen im Unternehmen oder anderem Bedarf angepasst werden. Hierbei müssen beide Sprachen von den Nutzern gepflegt werden. Nicht autorisierte Personen sollen diese Seite weder sehen, noch darauf zugreifen können.

4 Designphase

In der Designphase werden Benutzeroberfläche und Datenmodelle der Anwendung geplant. Dadurch können benötigte Ressourcen genauer abgeschätzt sowie Schwierigkeiten oder Fehler im Voraus erkannt werden.

4.1 Benutzeroberfläche

Bei dem Design der Benutzeroberfläche (im folgenden GUI genannt) wird besonders darauf geachtet, dass alle Daten möglichst übersichtlich dargestellt werden. Dazu werden für jede der großen Anforderungen jeweils einzelne Unterseiten (im folgenden View genannt) unterteilt.

Hierbei gibt es eine View, um alle Achievements einsehen und abhaken zu können. Achievements werden zu verschiedenen Zeitpunkten freigeschaltet. Manche sind direkt verfügbar, manche nach einer Woche, einem Monat oder erst nach einem halben Jahr. Entsprechend dieser Zeitstempel sind die Achievements in Akkordeons gruppiert. Dies führt dazu, dass Achievements übersichtlich einsehbar sind.

Eine weitere View gibt es für die Ansicht der Rangliste. Hier wird der derzeit eingeloggten Person eine Rangliste mit den Personen, die zu einer ähnlichen Zeit angefangen haben, angezeigt. Jeder wird mit Platzierung, Punktzahl, Name, sowie der Email-Adresse angezeigt (Abbildung 1).

Eine letzte View dient der Verwaltung von Achievements. Die View ist ähnlich aufgebaut, wie die Achievements-View. Neben den Akkordeon-Headern befindet sich zusätzlich jeweils ein weiterer Button, um neue Achievements für diesen Zeitpunkt hinzufügen zu können. Anstatt Achievements abzuhaken zu können, findet der Nutzer hier einen Button, um Achievements zu bearbeiten oder zu löschen (Abbildung 2).

Beim Hinzufügen oder Bearbeiten eines Achievements öffnet sich ein Modaldialog. In diesem Dialogfenster kann der deutsche und englische Text, sowie der Zeitpunkt für die Freischaltung des Achievements und die Punktzahl eingestellt werden (Abbildung 3). Bei Bestätigung des Dialoges werden die Daten in die Datenbank übertragen und dort gespeichert. Fehlende Eingaben werden dem Nutzer über eine Eingabeaufforderung am jeweiligen Eingabefeld gekennzeichnet.

Um ein einheitliches Design mit anderen internen Applikationen der BREDEX GmbH herzustellen, wurde der Stil der Software am BREDEX-Styleguide orientiert. Hier sind grundsätzliche Designentscheidungen für BREDEX-Software, wie z.B. Farben oder Schrift normiert. Auch die Login-Seite ist dem standardisierten Design der BREDEX GmbH angepasst.

4.2 Datenbank

Bei der Datenbank handelt es sich um die NoSQL-Datenbank Firebase von Firestore. Die Daten werden im JSON-Format gespeichert. Hierbei wird von Sammlungen und Attributen gesprochen. Sammlungen sind mehrere JSON-Objekte, mit einer ID und einem Datensatz. Dieser Datensatz

kann weitere Sammlungen beinhalten, sowie Attribute wie beispielsweise Zahlen, Zeichenketten oder Arrays.

Um die Achievements und die Daten der Nutzer der Anwendung speichern zu können, wurden in der Datenbank jeweils Sammlungen angelegt. Die Referenzen der Nutzer zu ihren erledigten Achievements werden in einer ID-Liste gespeichert, anstatt eine Benutzereigene Sammlung für Achievements anzulegen. Dies entspricht eher einem relationalen Ansatz, der eher unüblich für NoSQL-Datenbanken ist. Die Entscheidung wurde getroffen, um die Redundanz von Daten zu verringern und das Pflegen von Achievements einfacher umsetzen zu können.

5 Implementierungsphase

In der Implementierungsphase wird die Anwendung programmiert. Hierbei werden die, in der Planungs- und Designphase erstellten, Strukturen und Modelle genutzt, um die Software möglichst ohne Komplikationen umsetzen zu können.

5.1 Frontend

Im Frontend wird das Webframework Vue eingesetzt. Dieses baut auf den standard Websprachen HTML, JavaScript und CSS auf. Vue nutzt komponentenbasierte Programmierung, indem es HTML um die Möglichkeit erweitert, Templates zu erzeugen. Durch diese Templates wird dann, entsprechend dem zugrunde liegenden Java-Scripts, die GUI erzeugt. In dieser Anwendung wird Vue-Routing genutzt, um eine Single-Page Application zu erstellen. Die Webseite besteht bei einer Single-Page Application nur aus einer HTML-Datei. Diese Datei wird von Vue im Buildprozess der Anwendung erstellt. Der Inhalt des HTML kann dynamisch mit den verschiedenen Komponenten ausgetauscht werden.

Standardmäßig wird in Vue die Scriptsprache JavaScript genutzt. Das Framework kann auch mit der Sprache TypeScript genutzt werden, wofür sich in diesem Projekt entschieden wurde, um die Vorteile von Typisierung nutzen zu können. Dadurch können gewisse Fehler bereits zur Kompilierzeit vom Compiler entdeckt werden und die Anwendung gegenüber Fehlern robuster gemacht werden.

Vue stellt seit Version 3.0 auch eine neue API zur Verfügung: die Composition API. Die Vorteile gegenüber der, in Vue standardmäßig genutzten, Options API, ist die Möglichkeit, den Code variabler und besser strukturieren zu können. Dies verbessert die Lesbarkeit und Qualität des Codes. Außerdem erleichtert die Composition API die Wiederverwendbarkeit von Code in verschiedenen Komponenten, was eine große Auswirkung auf Codequalität und Fehlerrobustheit hat.

Für Styling wird das CSS-Framework Bootstrap genutzt, um einfacher ein modernes Design für die Anwendung erstellen zu können und sich auf die Programmierung des Projektes konzentrieren zu können.

Als erstes wurde die Achievement-View erstellt, da es sich bei diesem Feature um die Hauptanforderung der Website handelt. Hier werden alle Achievements angezeigt, die der Benutzer

erledigen kann. Achievements, die noch zu erledigen sind, werden hierbei hervorgehoben, um leichter den Überblick behalten zu können (Abbildung 4). Am oberen Rand der Website gibt es eine kurze Einführung in die Website und eine Fortschrittsleiste zeigt an, wie viel der Nutzer bereits erledigt hat. Jede Vue-Komponente hat ihren eigenen Lifecycle. Hierzu gehören beispielsweise das Erstellen, Mounten, Aktualisieren oder Unmounten der Komponente. Der OnMounted-Lifecycle Hook wird hier genutzt, um kurz vor dem Anzeigen der Komponente, die Daten aus der Datenbank zu laden. Diese Vorgehensweise wird in anderen Komponenten ebenfalls genutzt, um Daten zu laden.

Als nächstes wurde die Login-View erstellt. Da in der BREDEX GmbH Microsoft Accounts genutzt werden, kann sich der Nutzer hier mit diesem Anmelden. Außerdem werden neue Nutzer erkannt, indem UUIDs verglichen werden und für jeweilige Nutzer neue Einträge in der Datenbank erstellt.

Im Anschluss wurde die Rangliste erstellt. Hier werden Mitarbeiter, die bis zu einem Monat vor oder nach dem aktuellen Nutzer angefangen haben, tabellenartig mit Name, Punktzahl, Email und Platzierung dargestellt.

Die letzte View dient der Verwaltung der Achievements. Auf diese View sollen nur Personen aus bestimmten Gruppen, wie beispielsweise der Personalabteilung, Zugriff erhalten. Dieser Zugriff kann mit dem Vue-Router implementiert werden, welcher eine Möglichkeit hat, eine Bedingung an die Weiterleitung an eine Adresse zu binden. In dieser Bedingung wird die Rolle des Nutzers abgefragt. Da diese bei der Anmeldung über Firebase nicht mit dem Access-Token übergeben wird, muss sich separat mit dem Azure Portal verbunden werden, um diese abzufragen.

Um die Mehrsprachigkeit umzusetzen, wird das Vue-Plugin Vue I18n genutzt. In einer separaten Datei wird zu jedem Text der Anwendung im JSON-Format ein Key gespeichert, zusammen mit den jeweiligen Übersetzungen. In dem HTML-Template wird nun anstelle des Textes der Key eingetragen, welcher beim Erstellen der Komponente, je nach aktuell eingestellter Sprache, durch die Übersetzung ersetzt wird.

Um die Anwendung bei Änderungen von Werten automatisch zu aktualisieren, gibt es in Vue ein Attribut, das eine Variable erhalten kann: das Ref. Vue rendert das DOM, bevor es auf dem Bildschirm dargestellt wird. Dadurch werden im Nachhinein veränderte Werte nicht mehr dargestellt. Refs haben hierbei einen Sonderstatus. Sobald sich die Variable mit dem Attribut ändert, wird die Komponente mit dem neuen Wert neu berechnet.

5.2 Backend

Für das Backend wird das Backend-As-A-Service Firebase genutzt. Darin sind verschiedene Optionen zur Benutzeranmeldung, wovon nur die Anmeldung über Microsoft genutzt wird, sowie die NoSQL-Datenbank Firestore enthalten. Als erstes muss das Backend mit der Software verbunden werden. Dafür wird als erstes die Firebase SDK mit dem Befehl

```
npm install firebase
```

über Node.js installiert und im Anschluss in den Abhängigkeiten der Vue-Anwendung hinzugefügt.

Anschließend wird die Vue-App mit den Firebase Einstellungen konfiguriert.

Als nächstes wird die Datenbank erstellt. Die NoSQL-Datenbank basiert auf JSON-ähnlichen Dateistrukturen. Man kann Sammlungen mit verschiedenen Elementen erstellen. Diese Elemente haben jeweils eine ID, welche als Zeichenkette gespeichert wird. Für Nutzer und Achievements wird jeweils eine Sammlung angelegt.

Die Achievements müssen für beide Sprachen ihren Text enthalten, welcher jeweils als Zeichenkette gespeichert wird. Des Weiteren muss der Zeitpunkt gespeichert werden, in dem das Achievement freigeschaltet wird. Da diese an vier verschiedenen möglichen Zeitpunkten freigeschaltet werden können, wurde die Entscheidung getroffen, für das Attribut einen Wert von 0 bis 3 zu speichern. Dadurch können die Achievements leicht gruppiert und mit einem Zähler iteriert werden. Die Punkte, die jedes Achievement hat, werden in einer Ganzzahl von 0 bis 5 gespeichert.

Für die ID des Nutzers wird dessen Microsoft-UUID genutzt. So kann nach der Anmeldung leicht wieder auf die Daten eines bestimmten Nutzers zugegriffen werden. Abgesehen von der ID wird für den Nutzer der Zeitpunkt gespeichert, an der sich der Nutzer das erste mal angemeldet hat. Dadurch kann beim Abruf der Nutzer für die Rangliste bereits in der Datenbank gefiltert werden. Weiterhin werden noch Stammdaten wie Name oder Email-Adresse gespeichert. Zuletzt wird für jeden Nutzer noch eine Liste an Zeichenketten gespeichert. Diese Liste beinhaltet alle IDs der Achievements, die der Nutzer bereits erledigt hat. Über diese IDs kann effizient die Liste an erledigten Achievements geladen werden, um Punkte oder die prozentuale Anzahl erledigter Achievements zu berechnen.

6 Testphase

Um die korrekte Arbeitsweise der Anwendung zu garantieren, werden während der Entwicklung regelmäßig Tests durchgeführt.

Mithilfe von Unit-Tests wurde während der Entwicklungsphase sichergestellt, dass Methoden korrekt funktionieren. Unit-Tests werden während der Implementierung der Software automatisiert ausgeführt, um Fehler schnell erkennen und beheben zu können.

In regelmäßigen Abständen führte der Prüfungsteilnehmer explorative Tests durch. Hierbei wird die GUI durch falsche Eingaben auf Robustheit getestet. Damit wird sichergestellt, dass bei falscher Bedienung der Software keine ungewollten Fehler auftreten, die den Nutzer von seiner Arbeit abhalten.

Da bei der Software die Usability im Vordergrund steht, wurden zusätzlich User-Tests durchgeführt. Bei diesem Testverfahren wurden Nutzern, die die Software noch nicht kannten, Aufgaben gestellt. Diese Aufgaben sollten erledigt und Gedankengänge erläutert werden. Dadurch wurde überprüft, ob die GUI intuitiv bedienbar ist und sich Aufgaben ohne Komplikationen erledigen lassen. Falls sich Differenzen zwischen dem Design und den Erwartungen des Nutzers feststellen lassen, kann das Design im Anschluss angepasst und verbessert werden. Eine Differenz, die hierbei festgestellt wurde ist, dass der Nutzer beim Anklicken des Namens der Anwendung in der Navigationsleiste erwartet, auf die Home-Seite der Anwendung zu kommen, was in diesem Fall die

Achievement-View ist. Dieses Feature wurde im Folgenden implementiert. Ansonsten konnten alle Aufgaben ohne Probleme bearbeitet werden.

Bei den Testverfahren wurden außerdem die Dienste eines Test-Consultants in Anspruch genommen, um die Anwendung noch einmal ohne Einfluss der Projektleitung oder des Programmierers zu testen. Hierbei wurden ein paar kleinere Bugs gefunden. Diese wurden im Anschluss vom Prüfungsteilnehmer bearbeitet und korrigiert.

Außerdem gab es Code Reviews von erfahreneren Entwicklern, um die Codequalität sicherzustellen und um effizientere Wege gezeigt zu bekommen, um diese im Anschluss anwenden zu können.

7 Abnahme

Zum Abschluss des Projektes fand ein Abnahmegespräch mit der Projektbetreuung statt. Bei diesem Gespräch wurde in einem Abnahmetest überprüft, ob alle Projektziele erreicht wurden. Das Ergebnis des Tests war sehr gut. Es ist geplant, die Software in naher Zukunft bei BREDEX im produktiven Betrieb zu nutzen.

8 Fazit

8.1 Soll/Ist-Vergleich

Während der Durchführung der Projektarbeit kam es zu kleinen Differenzen zu der initialen Zeitplanung.

Da ein paar Anforderungen während der Implementierungsphase hinzu kamen, musste in die Designphase zurückgewechselt werden, da das Design erst überarbeitet werden musste, um die neuen Anforderungen zu berücksichtigen. Dieser Aufwand kostete den Prüfungsteilnehmer in etwa eine Stunde.

In der Implementierungsphase kam es durch zusätzlich erforderliche Einarbeitung in die Webentwicklung und das Vue-Framework zu einigen Verzögerungen. Diese Verzögerungen konnten jedoch durch gut gewählte Bibliotheken und Frameworks wieder aufgeholt werden. Insgesamt konnte sogar eine Stunde zusätzlich gewonnen werden, da in der Designphase gut genug gearbeitet wurde, um schneller Implementieren zu können als ursprünglich geplant.

Die restlichen Phasen wurden gut eingeschätzt, wodurch keine weiteren Abweichungen in der Zeitplanung während der Projektdurchführung auftraten.

Einen graphischen Überblick über die Änderungen der Zeitplanung bietet Tabelle 4.

Da im Projektverlauf keine ungeplanten Ressourcen benötigt wurden, gibt es keine Differenzen in der Kosten- oder Ressourcenplanung.

8.2 Lessons learned

Während des Projektes lernte der Prüfungsteilnehmer sehr viel über Webentwicklung und das Planen und Umsetzen von Projekten. Vor allem die Wichtigkeit und Auswirkung einer guten Design- und Analysephase in einem Projekt haben sich gezeigt. Aber auch die Wahl von Technologie und genutzten Bibliotheken können einen großen Einfluss auf den flüssigen und schnellen Ablauf der Softwareentwicklung haben.

Glossar

Achievement Aufgaben, die neue Mitarbeiter bei der BREDEX GmbH zusammen mit ihren BREDEX-Buddys erledigen können, um sich ins Unternehmen einzuarbeiten. 4–9

API (Application Programming Interface / Programmierschnittstelle) Teil eines Softwaresystems, die dem Programmierer auf Quelltextebene den Anschluss an ein anderes System erlaubt. 7

Backend-As-A-Service Dienst, der eine Entwicklungsumgebung im Browser zur Verfügung stellt mit Anbindung an eine Cloud, um die Entwicklung von Backends für Webanwendungen zu vereinfachen. 3, 8, 19

Bootstrap CSS-Framework, das Gestaltungsvorlagen für Benutzeroberflächen bereitstellt. 7

BREDEX-Buddy Ansprechpartner für neue Mitarbeiter bei der BREDEX GmbH. 1, 4, 5

explorative Tests Die Benutzeroberfläche wird von dem Testenden auf fehlerhafte oder falsche Eingaben überprüft, um Robustheit zu garantieren. 9

Firebase Entwicklungsplattform für mobile und Webentwicklung, die Tools und Infrastruktur zur Verfügung stellt. 1, 3, 6, 8, 19

Firestore NoSQL-Datenbank, die in Firebase integriert ist. 1, 3, 6, 8, 19

Gamification Übertragung von spieltypischen Elementen und Vorgängen in spielfremde Zusammenhänge. 1, 4, 5

git Software zur Versionsverwaltung von Dateien. 3, 19

modaldialog Dialogfenster im Vordergrund einer Webanwendung. 6

NoSQL Nicht-relationale Datenbankstruktur ohne festgelegte Tabellenstruktur. 6–9

ref Ein Objekt in Vue, dessen Wert in der Benutzeroberfläche in Echtzeit aktualisiert werden kann. 8

Single-Page Application Eine Webanwendung, die aus einer einzigen HTML-Datei besteht, deren Inhalte dynamisch nachgeladen werden. 7

Unit-Tests Automatisierte Tests, die Methoden auf korrekte Funktionsweise testen. 9

User-Test Einen Nutzer werden Aufgaben gestellt, die dieser mit der Anwendung erledigen soll. Mit diesen Tests wird die intuitive Bedienbarkeit der Anwendung getestet. 9

UUID (Universal Unique Identifier) 128-bit Zahl, die zur Identifikation von Informationen in Computersystemen genutzt wird. 8, 9

view Anderer Name für eine Komponente in Vue. Views sind in den meisten Fällen Komponenten, die eine Unterseiten der Webseite darstellen. 6–8, 10

VisualStudio Code Code-Editor für die Programmierung von verschiedenen Programmiersprachen. 3

vue HTML-Framework zum Erstellen von Webanwendungen. 1, 7, 8, 10

Vue I18n Vue Plugin, um leicht die Anwendung einer Sprache wechseln zu können. 8

Tabellenverzeichnis

1	Zeitplan	3
2	Kostenplan	15
3	Nutzwertanalyse	15
4	Zeitplanung; Differenzen	15

Abbildungsverzeichnis

1	Ranking	16
2	Achievements - Manage-View	16
3	ModalDialog - Add/Edit	16
4	Achievements	17

Listings

1	Lade Daten Beispiel	18
2	OnMounted-Lifecyclehook Beispiel	18
3	Achievement Datenmodell	18

Akteur	Stundensatz (in EUR)	Dauer (in Stunden)	Kosten (in EUR)
Prüfungsteilnehmer	9	80	720
Projektbetreuer	45	6	270
Gesamtkosten			990

Tabelle 2: Kostenplan

Kriterium	Gewichtung	Alte Struktur		Neue Struktur	
		Erfüllung	Nutzwert	Erfüllung	Nutzwert
Usability	50%	2	1	4	2
Einheitlichkeit	30%	2	0,6	4	1,2
Motivation	10%	1	0,1	4	0,4
Sicherheit	10%	2	0,2	4	0,4
SUMME	100%	7	1,9	15	4,0

Bewertungsmaßstab: 5 = sehr hoch, 4 = hoch, 3 = mittel, 2 = gering, 1 = sehr gering

Tabelle 3: Nutzwertanalyse

Phase	geplante Dauer (in Stunden)	tatsächliche Dauer (in Stunden)	Differenz (in Stunden)
Anforderungsgespräch	3	3	0
Analysephase	6	6	0
Planungsphase	6	6	0
Designphase	6	7	+1
Umsetzung	40	39	-1
Test und Abnahme	7	7	0
Dokumentation	12	12	0
Gesamt	80	80	+0

Tabelle 4: Zeitplanung; Differenzen

Place	Name	Mail	Points
1	Julian Thiele	juliant@dev.bredex.de	35
2			24
2			24
4	Christian Rucinski	christianr@dev.bredex.de	0

Abbildung 1: Ranking

Meinen Buddy kennengelernt

Punkte: 2

Bearbeiten

Löschen

Abbildung 2: Achievements - Manage-View

Achievement hinzufügen

Englischer Text

Deutscher Text

Erster Tag

1 Punkt

Abbrechen

Hinzufügen

Abbildung 3: Modaldialog - Add/Edit

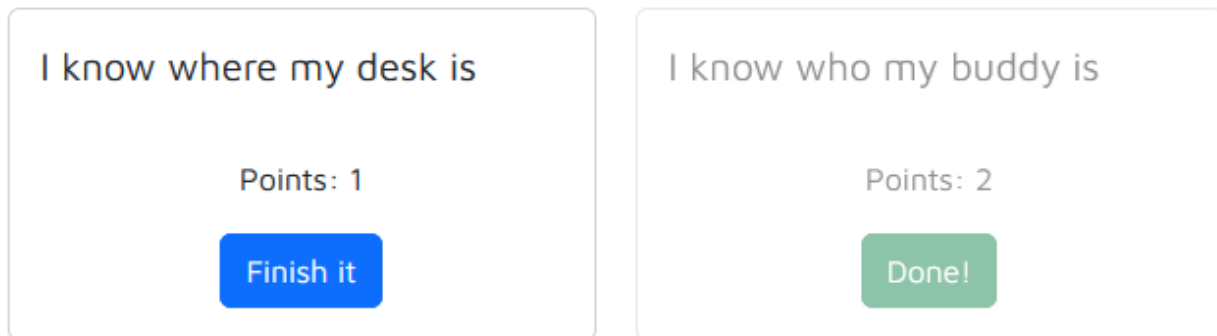


Abbildung 4: Achievements

```

1 fetchData() {
2     firebase.firestore().collection("/achievements")
3         .onSnapshot(snapshot => {
4             this.achievements = []
5             snapshot.forEach(doc => {
6                 this.achievements.push({
7                     id: doc.id,
8                     name_de: doc.data().name_de,
9                     name_en: doc.data().name_en,
10                    timestamp: doc.data().unlock_time,
11                    points: doc.data().points
12                })
13            })
14        })
15 }

```

Listing 1: Lade Daten Beispiel

```

1 OnMounted(async () => {
2     doneTasks.value = UserService.getDoneAchievements();
3 })

```

Listing 2: OnMounted-Lifecyclehook Beispiel

```

1 {
2     id: string,
3     data: {
4         name_de: string,
5         name_en: string,
6         points: int,
7         unlock_time: int,
8     }
9 }

```

Listing 3: Achievement Datenmodell

A genutzte Ressourcen

Personal

- Projektbetreuer
- Prüfungsteilnehmer

Software

- Windows 10 Professional
- VisualStudio Code
- Firebase als Backend-As-A-Service
- Firestore als Datenbank
- Git
- GitLab
- GitHub

Hardware

- ThinkPad
- 2 Monitore
- Tastatur
- Maus

Digitalisierung des internen Onboarding Prozesses neuer Mitarbeiter



BREDEX

Benutzerdokumentation

Inhaltsverzeichnis

1	Startseite	1
2	Allgemeiner Aufbau	1
2.1	Navigation	1
2.2	Hinweise	2
3	Achievement Liste	2
3.1	Überblick	2
3.2	Achievements erledigen	3
3.3	Fortschrittsleiste	3
4	Rangliste	3
5	Management	4
5.1	Navigierung zur Management Seite	4
5.2	Achievements anlegen	4
5.3	Achievement bearbeiten	6
5.4	Achievement löschen	6

1 Startseite

Beim Aufrufen der Webseite gelangt der Nutzer auf die Login-Seite der Webanwendung. Über den Login-Button kann der Nutzer sich mit seinem Microsoft-Account anmelden. Ist dieser auf dem Gerät bereits mit einem Microsoft-Account angemeldet, wird die Anmeldung beim Betätigen des Buttons automatisch ausgeführt.



Abbildung 1: Login-Seite

2 Allgemeiner Aufbau

2.1 Navigation

Am oberen Rand der Anwendung befindet sich eine Navigationsleiste. Darüber können die verschiedenen Seiten der Anwendung erreicht werden.

- **Achievements:** Die Hauptseite der Anwendung. Hier können Achievements erledigt werden.
- **Rangliste:** Auf dieser Seite wird die Rangliste dargestellt, bei der man sich mit anderen Nutzern vergleichen kann.
- **Bearbeiten:** Hier können Achievement bearbeitet werden. Nur Personen mit Bearbeitungsberechtigung haben Zugriff auf diese Seite.

Der Bearbeiten-Tab ist nur für Nutzer sichtbar, die Zugriff auf diese Seite besitzen.

Auf der rechten Seite der Navigationsleiste kann die Sprache der Anwendung zwischen Deutsch und Englisch umgestellt werden. Außerdem kann sich der Nutzer dort über den Ausloggen-Button ausloggen.

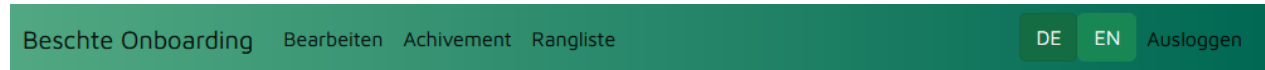


Abbildung 2: Navigationsleiste

2.2 Hinweise

Direkt unter der Navigationsleiste jeder Seite befindet sich eine kurze Beschreibung, was der Nutzer auf der jeweiligen Seite tun kann.

3 Achievement Liste

3.1 Überblick

Nach der Anmeldung wird der Nutzer auf die Achievement-Ansicht weitergeleitet. Achievements werden in 4 Zeitpunkte unterteilt:

Erster Tag — Erste Woche — Erster Monat — Erstes halbes Jahr

Je nach dem, wie lange der Nutzer bereits bei der BREDEX GmbH arbeitet, werden Achievements freigeschaltet. Ein Nutzer, der bereits seit einem Jahr bei BREDEX arbeitet, hat alle Sektionen freigeschaltet, während einem Nutzer, der erst seit 2 Wochen bei BREDEX arbeitet, nur die ersten beiden Abschnitte zur Verfügung stehen.

Erster Tag	▼
Erste Woche	▼
Erster Monat	▼
Erstes halbes Jahr	▼

Abbildung 3: Sektionen - Achievementseite

3.2 Achievements erledigen

Klappt der Nutzer eine Sektion aus, werden die Achievements dieses Zeitabschnittes angezeigt. Jedes Achievement beinhaltet einen Text, der die Aufgabe des Achievements angibt. Darunter werden die Punkte angezeigt, die das jeweilige Achievement gibt. Durch einen Klick auf den Button in dem Achievement kann dieses als erledigt markiert werden. Über denselben Button können bereits erledigte Achievement wieder als nicht-erledigt markiert werden. Erledigte Achievements werden in der Darstellung ausgegraut, um sie von nicht-erledigten unterscheiden zu können und nicht-erledigte Achievements hervorzuheben.

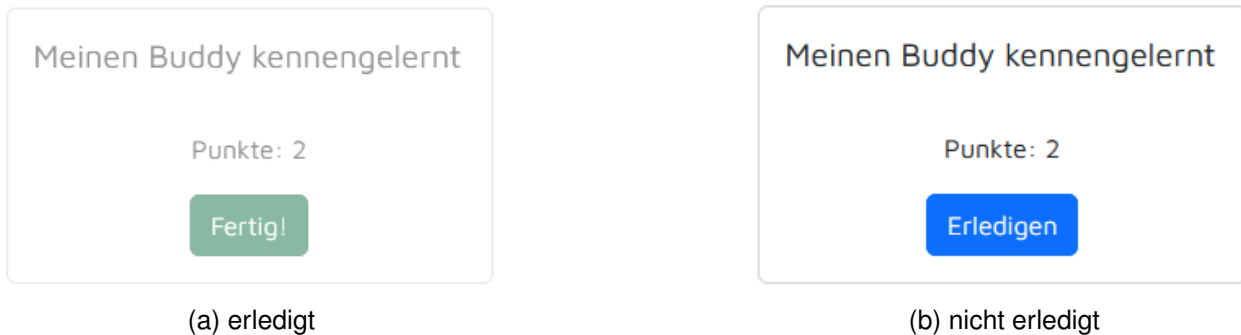


Abbildung 4: Achievement

3.3 Fortschrittsleiste

Eine Leiste im oberen Teil der Ansicht zeigt dem Nutzer an, wie viele Achievements er bereits erledigt hat. Ist die Leiste voll, sind alle Achievements erledigt, die dem Nutzer bereits verfügbar sind.



Abbildung 5: Fortschrittsleiste

4 Rangliste

Über die Navigation gelangt der Nutzer auf die Ansicht der Rangliste. Welche Nutzer in der Rangliste angezeigt werden, hängt von dem aktuell angemeldeten Nutzer ab. Damit alle Nutzer in der Rangliste ungefähr ähnliche Voraussetzungen haben, werden nur andere Nutzer angezeigt, die höchstens einen Monat vor oder nach dem aktuell angemeldeten Nutzer bei der BREDEX GmbH angefangen haben.

Place	Name	Mail	Points
1	Julian Thiele	juliant@dev.bredex.de	35
2			24
2			24
4	Christian Rucinski	christianr@dev.bredex.de	0

Abbildung 6: Rangliste

Die Nutzer werden in der Rangliste nach ihrer Platzierung absteigend sortiert. Je mehr Punkte ein Nutzer hat, desto besser ist seine Platzierung. Die Punkte, die ein Achievement dem Nutzer einbringt, richtet sich nach der Schwierigkeit des Achievements und kann zwischen 0 und 5 Punkten variieren. Nutzer mit gleicher Punktzahl erhalten in der Rangliste dieselbe Platzierung.

5 Management

5.1 Navigierung zur Management Seite

Die Management Seite kann von Nutzern erreicht werden, denen es erlaubt ist Achievements zu bearbeiten. Der Link in der Navigationsleiste kann nur von jeweiligen Nutzern gesehen werden.

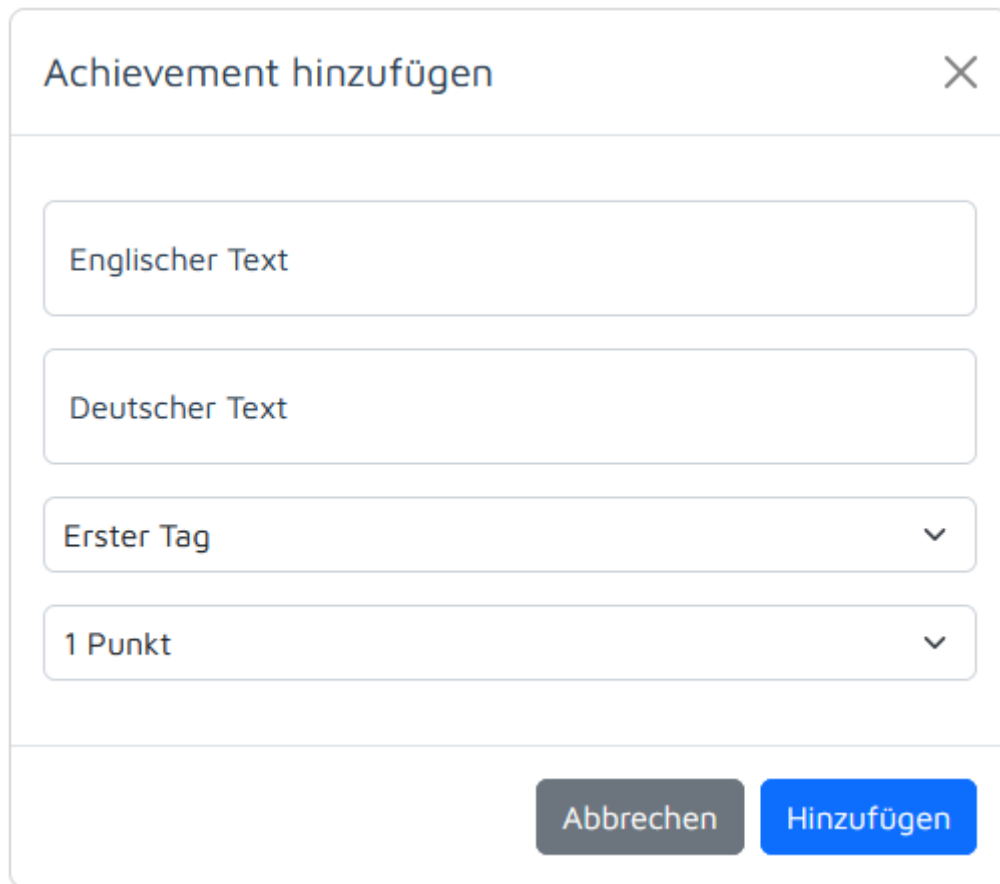
5.2 Achievements anlegen

Auf der Management Seite sind die Achievements, ähnlich der Achievements Seite, in ihre Zeitabschnitte unterteilt. Auf der rechten Seite eines jeden Zeitabschnittes kann auf einen Button mit einem "+" geklickt werden. Im Anschluss dazu öffnet sich ein Modaldialog. Über diesen Dialog können Achievements angelegt werden.

Bei den ersten beiden Textfeldern muss der Nutzer die Aufgabe des Achievements eintragen. Das erste der beiden Felder ist für den englischen, das Zweite für den deutschen Text. Fehlt die Angabe in einem der beiden Textfelder, kann der Dialog nicht bestätigt werden. Das Feld mit fehlender Angabe, wird dem Nutzer durch eine Eingabeaufforderung an dem Textfeld angezeigt.

Erster Tag	▼	+
Erste Woche	▼	+
Erster Monat	▼	+
Erstes halbes Jahr	▼	+

Abbildung 7: Sektionen bei Bearbeitung



Achievement hinzufügen

Englischer Text

Deutscher Text

Erster Tag

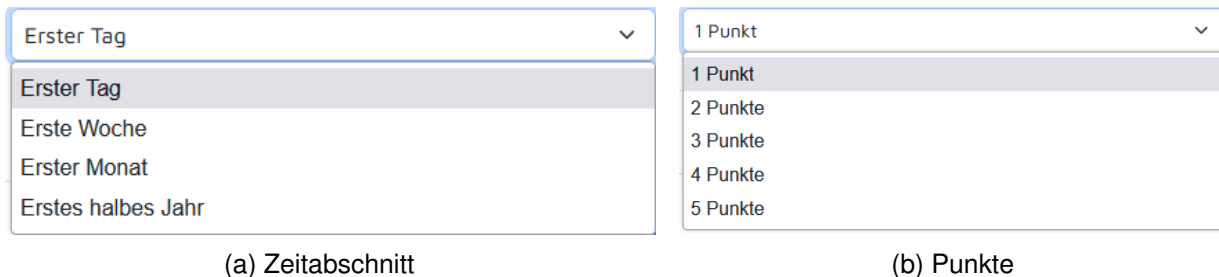
1 Punkt

Abbrechen Hinzufügen

Abbildung 8: Modaldialog

Im Anschluss kann der Zeitpunkt für die Freischaltung des Achievements ausgewählt werden. Je nach dem, neben welchem Abschnitt der Nutzer auf den Button gedrückt hat, ist dieses Feld bereits vorausgefüllt, kann aber über das Dropdown-Menü im Nachhinein geändert werden.

Final kann noch die Punktzahl des Achievements ausgewählt werden. Diese ist standardmäßig auf 1 gesetzt, kann aber über das Dropdown-Menü in einen Wert von 1 bis 5 geändert werden.



(a) Zeitabschnitt

(b) Punkte

Abbildung 9: Dropdown Menüs

5.3 Achievement bearbeiten

Jedes Achievement kann über einen Button bearbeitet werden. Dieser Button befindet sich auf der Karte des Achievements. Beim Drücken den Bearbeiten-Buttons, öffnet sich der selbe Dialog, der sich beim Erstellen eines neuen Achievements öffnet. Der Dialog ist beim Bearbeiten mit den Daten des Achievements vorausgefüllt, das der Nutzer gerade bearbeitet.

5.4 Achievement löschen

Neben dem Bearbeiten-Button existiert für jedes Achievement noch ein Button, um dieses zu löschen. Versucht Nutzer ein Achievement zu löschen, öffnet sich ein Bestätigungsdialog, damit Achievements nicht aus versehen gelöscht werden können.





Winterprüfung 2023

Ausbildungsberuf

Fachinformatiker/Fachinformatikerin (VO 2020) Fachrichtung:
Anwendungsentwicklung

Prüfungsbezirk

1201 FIA 4 (AP T2V1)

Julian Rafael Thiele

Identnummer: 734117

E-Mail: julian.thiele@bredex.de, Telefon: +49 1515 6324179

Ausbildungsbetrieb: Bredex GmbH

Projektbetreuer: Christian Rucinski

E-Mail: christian.rucinski@bredex.de, Telefon: +49 531 243300

Thema der Projektarbeit

Digitalisierung des internen Onboarding Prozesses neuer Mitarbeiter*innen

1 Thema der Projektarbeit

Digitalisierung des internen Onboarding Prozesses neuer Mitarbeiter*innen

2 Geplanter Bearbeitungszeitraum

Beginn: 22.09.2023

Ende: 22.11.2023

3 Ausgangssituation

In der IT-Branche herrscht sehr viel Wachstum und es existiert ein großer Markt an Jobangeboten. Da die BREDEX GmbH ein Teil dieser Branche ist, erfährt sie auch ein entsprechendes Wachstum. Regelmäßig werden neue Talente eingestellt, um mit der wachsenden Anfrage des IT-Marktes mitzuhalten.

Um den neuen Mitarbeiter*innen den Einstieg in das Unternehmen zu erleichtern, werden Ihnen zwei Ansprechpersonen zu Verfügung gestellt. Einer davon ist der sogenannte BREDEX-Buddy, der neuen Mitarbeiter*innen bei allgemeinen Fragen und dem Kennenlernen des Unternehmens zur Seite stehen soll. Die Aufgaben eines Buddys bei der BREDEX GmbH sind sehr vielseitig. Vom Helfen bei der Einrichtung des Arbeitslaptops, Einweisen in die intern genutzte Software bis hin zum Näherbringen des Freizeitangebots und der Firmenevents. Was genau die Aufgaben des Buddys sind, ist aber noch nicht einheitlich in den Kommunikationskanälen und -medien der BREDEX GmbH dargestellt. Die Folge davon ist, dass verschiedene Buddys ihre Aufgaben auf unterschiedliche Art und Weise umsetzen, je nach dem, an welcher dieser Quellen sie sich orientieren.

Um in Zukunft die Aufgaben des BREDEX-Buddys zentralisiert einsehbar zu machen, soll online ein Onboarding-Tool zur Verfügung gestellt werden. Hier soll es sogenannte Achievements geben, kleine Aufgaben, die neue Mitarbeiter*innen erledigen können. Diese Achievements spiegeln verschiedene Aspekte des Unternehmens wider. Beim Erledigen der Aufgaben lernen sie das Unternehmen besser kennen und unternimmt etwas mit anderen Mitarbeiter*innen. Außerdem informieren die Aufgaben über weitere Freizeitangebote wie beispielsweise Sportaktivitäten, die die BREDEX GmbH zur Verfügung stellt.

In der Anwendung soll es eine Rangliste geben, in denen einem diejenigen Mitarbeiter*innen angezeigt werden, die im selben Monat wie der User selber angefangen haben. Dadurch kann verglichen werden, wer seitdem mehr Punkte gesammelt hat. Hierdurch sollen die Mitarbeiter*innen ermutigt werden, Aufgaben zu erledigen und somit das Unternehmen kennenzulernen. Achievements sollen je nach Schwierigkeit und Aufwand unterschiedlich viele Punkte geben.

Jede*r Mitarbeiter*in erhält entsprechend der eigenen Position im Unternehmen eine Rolle zugewiesen, die es ihm erlaubt, diese zu bearbeiten, zu löschen oder Neue zu erstellen. Personalverantwortliche und Verantwortliche für Buddys haben das Recht zu bearbeiten und zu löschen, während alle anderen Mitarbeiter*innen Achievements lediglich als „Erledigt“ markieren und die Rangliste einsehen können. Die verschiedenen Funktionen werden über sogenannte Views zur Verfügung gestellt, auf die jeweils nur berechtigte Personen Zugriff



haben. Views sind verschiedene Unterseiten der Webanwendung mit einem eigenen Link. Auf die Seiten, die unter den Links er Auch die Mehrsprachigkeit im Unternehmen soll in der Anwendung beachtet werden, sodass bei der dargestellten Sprache zwischen Deutsch und Englisch gewählt werden kann.

reichbar sind, können also nur von Personen mit der jeweiligen Berechtigung geöffnet werden. Achievements sowie User werden in der NoSQL Datenbank von Firebase gespeichert. Firebase wird in Form von einem Backend as a Service genutzt. Für das Frontend wird das Framework Vue mit der Skriptspache TypeScript verwendet.

4 Projektziel

Mithilfe der Anwendung soll es für buddys eine zentrale, interaktive Liste geben, an der sie sich orientieren können, um neuen Mitarbeiter*innen das Unternehmen und dessen Kultur näher zu bringen. Dies verhindert in Zukunft die Differenzen in der Art und Weise, wie Buddys ihren Aufgaben nachgehen.

5 Zeitplanung

- Anforderungsgespräch: 3h
- Analysephase: 6h
- Planungsphase: 6h
- Designphase: 6h
- Umsetzung: 40h
- Test und Abnahme: 7h
- Dokumentation: 12h

Gesamt: 80h

6 Anlagen

keine

7 Präsentationsmittel

- Notebook
- Projektor
- Karteikarten
- Als Rüstzeit sind 7 Minuten eingeplant

8 Hinweis!

Ich bestätige, dass der Projektantrag dem Ausbildungsbetrieb vorgelegt und vom Ausbildenden genehmigt wurde. Der Projektantrag enthält keine Betriebsgeheimnisse. Soweit diese für die Antragstellung notwendig sind, wurden nach Rücksprache mit dem Ausbildenden die entsprechenden Stellen unkenntlich gemacht.

Mit dem Absenden des Projektantrages bestätige ich weiterhin, dass der Antrag eigenständig von mir angefertigt wurde. Ferner sichere ich zu, dass im Projektantrag personenbezogene Daten (d. h. Daten über die eine Person identifizierbar oder bestimmbar ist) nur verwendet werden, wenn die betroffene Person hierin eingewilligt hat.

Bei meiner ersten Anmeldung im Online-Portal wurde ich darauf hingewiesen, dass meine Arbeit bei Täuschungshandlungen bzw. Ordnungsverstößen mit „null“ Punkten bewertet werden kann. Ich bin weiter darüber aufgeklärt worden, dass dies auch dann gilt, wenn festgestellt wird, dass meine Arbeit im Ganzen oder zu Teilen mit der eines anderen Prüfungsteilnehmers übereinstimmt. Es ist mir bewusst, dass Kontrollen durchgeführt werden.

genehmigt



Bestätigung über die durchgeführte Projektarbeit

(Diese Bestätigung ist mit der Projektdokumentation einzureichen!)

Berufsbezeichnung/Fachrichtung/Einsatzgebiet/Fachbereich:

Fachinformatiker Anwendungsentwicklung

Prüfungsteilnehmer/-in: Julian Thiele

Ausbildungsbetrieb oder Umschulungsträger: BREDEX GmbH

Azubi-Identnummer: 734117

Abschlussprüfung ~~Sommer~~ Winter 2023

Projektbezeichnung: Digitalisierung des internen Onboarding Prozesses neuer Mitarbeiter*innen

Projektbeginn: 22.08.2023 Projektfertigstellung: 22.11.2023 Zeitaufwand in Stunden: 80h

Bestätigung der Ausbildungsfirma:

Wir bestätigen, dass der/die Auszubildende das oben bezeichnete Projekt einschließlich der Dokumentationselbständig ausgeführt hat.



BREDEX

Software Entwicklungs- und Beratungs- GmbH

Lindentwete 1

38100 Braunschweig

21.11.2023

Datum

i.A. Ronja Kugler

Unterschrift

Die Dokumentation darf zu Unterrichtszwecken an der Berufsschule verwendet werden: JA

NEIN ☒

Eidesstattlich Erklärung:

Ich versichere, dass ich das Projekt und die dazugehörige Dokumentation selbständig erstellt habe,

21.11.2023, 38100 Braunschweig

Ort und Datum

Julian Thiele

Unterschrift des Prüflings