

NameSplitter

Softwarequalität
HOR-TINF2021

Erstellt von:
Schmidt Julian, Blattau Markus, Janocik Luka

Repository:
<https://github.com/ProgrammierenHorb/name-splitter>

Abgabetermin: 10.05.2024

Inhaltsverzeichnis

User Stories.....	3
Erste User Story	3
Zweite User Story.....	3
Dritte User Story	3
Vierte User Story	3
Fünfte User Story	3
Design	4
Testkonzept	5
Ziele:	5
Testkriterien:.....	5
Abdeckung:	5
Äquivalenzklassen:.....	5
Definition of Done (DoD)	6
Design:	6
Design (UI):.....	6
Code:	6
Release Instructions and Limitations (RiLI):	6
Unit Tests (UT):.....	6
Releasenotes.....	7
Benutzung.....	7
Backend ist nicht erreichbar:.....	7
Hauptanzeige.....	7
Rückgabe des Parsers	9
Titel löschen.....	9
Titel hinzufügen	10
Einschränkungen.....	10

User Stories

Erste User Story

Als Entwickler wünsche ich mir, dass eine eingegebene Anrede in ihre Bestandteile zerlegt werden kann. Dies ermöglicht eine effiziente Speicherung dieser innerhalb der anhängenden Datenbank. Ich weiß, dass das Ziel erreicht ist, wenn sie die Eingabe in die Bestandteile Titel, Anrede, standardisierte Briefanrede, Vorname, Nachname und Geschlecht unterteilt.

Story Points: 5

Priorität: 1

Zweite User Story

Als Anwender möchte ich eine Möglichkeit zur manuellen Eingabe, falls eine automatische Zerteilung nicht möglich ist. Dadurch möchte ich sicherstellen, dass meine Angaben valide sind. Ich weiß, dass das Ziel erreicht ist, wenn ich fehlende Informationen manuell ergänzen kann.

Story Points: 3

Priorität: 2

Dritte User Story

Als Anwender möchte ich eine einfache funktionale Benutzeroberfläche. Dies soll dabei helfen, eine Anrede einfach und schnell eingeben und speichern zu können. Ich weiß, dass das Ziel erreicht ist, wenn die Benutzeroberfläche intuitiv, vollständig und performant ist.

Story Points: 4

Priorität: 1

Vierte User Story

Als Anwender möchte ich einen neuen Titel innerhalb der Datenbank abspeichern können, um diesen bei der nächsten Anwendung erneut verwenden zu können, wodurch die Effizienz gesteigert werden würde. Ich weiß, dass das Ziel erreicht ist, wenn das System einen neu hinzugefügten Titel bei der nächsten Anwendung erkennt.

Story Points: 3

Priorität: 2

Fünfte User Story

Als Anwender möchte ich Anreden für den DACH-Markt abspeichern können, um das System auch in angrenzenden deutschsprachigen Ländern verwenden zu können. Ich weiß, dass das Ziel erreicht ist, wenn die typischen Anreden des DACH-Marktes erkannt werden.

Story Points: 3

Priorität: 3

Design

Testkonzept

Ziele:

Das Hauptziel der Tests ist, die Funktionalität und Nutzbarkeit des Kontakt-Splitters zu gewährleisten. Es soll sichergestellt werden, dass die Applikation Anreden korrekt in ihre Bestandteile zerlegt kann oder Benutzer korrekt informiert.

Testkriterien:

1. Funktionalität: Die Applikation soll alle definierten Anforderungen erfüllen.
2. Zuverlässigkeit: Die Ergebnisse sind konsistent und reproduzierbar. Zudem ist die Applikation fehlertolerant.
3. Benutzbarkeit: Die Applikation soll intuitiv und benutzerfreundlich sein.
4. Leistung: Die Applikation sollte innerhalb einer kurzen Zeit auf die Benutzereingabe reagieren (für den Fall, dass das Backend lokal gehostet wird in unter 0,5 Sekunden).
5. Wartbarkeit: Der Code ist wartbar (nach State of the Art). Die internen Module müssen gut modifizierbar sein.
6. Übertragbarkeit: Das System ist auf Windows 10 und 11 vollständig funktionsfähig.
7. Sicherheit: Falscheingaben werden nicht versehentlich abgespeichert (für die Datensicherheit ist das CRM-System verantwortlich).
8. Kompatibilität: Front- und Backend sind jeweils getrennt voneinander austauschbar und erweiterbar.

Abdeckung:

Es werden sowohl vollständig zerlegbare als auch nicht vollständig zerlegbare Anreden getestet, um sicherzustellen, dass die Applikation unter allen möglichen Umständen korrekt funktioniert.

Äquivalenzklassen:

1. Gängige Anreden: Es werden die gewöhnlichen Anreden wie "Herr", "Frau" und "Dr." überprüft.
2. Falsche Anreden: Es werden offensichtlich falsche und nicht passende Eingaben getestet.
3. Keine Eingabe: Es wird versucht ein leerer String oder „null“ zu parsen.
4. Ungewöhnliche Namen: Namen mit mehreren Bestandteilen wie Doppelnamen oder Namen mit Präfixen.
5. Kein Vorname: Der Vorname fehlt komplett, Nachname ist allerdings vorhanden.
6. Kein Vorname und Nachname: Vorname und Nachname fehlen in der Eingabe. Somit ist die Eingabe nicht valide und kann nicht geparkt werden.

Definition of Done (DoD)

Design:

- Die im Design enthaltenen Komponenten sind in der Lage alle definierten Anforderungen zu erfüllen
- Das Design ist einfach und lesbar gestaltet.
- Das Design ermöglicht eine Übersicht über das Gesamtsystem.
- Das Design enthält alle Relevanten Komponenten und Klassen.
- Das Backend und Frontend sind voneinander unabhängig ersetzbar.
-

Design (UI):

- Die UI ist intuitiv und Benutzerfreundlich gestaltet.
- Alle in den Anforderungen definierten Funktionen sind vorhanden.

Code:

- Der Code ist funktional und erfüllt die definierten UserStories.
- Der Code enthält keine Fehler, welche zum Abstürzen der Applikation oder zur Beschädigung bereits gespeicherter Daten führt.
- Der Code folgt den gängigen Codierungsstandards und Konventionen.
- Der Code ist verständlich und gut dokumentiert.
- Alle definierten Anforderungen sind erfüllt und korrekt umgesetzt.
- Fehler werden sinnvoll behandelt oder der Benutzer darauf hingewiesen.
- Das Backend und Frontend sind voneinander unabhängig ersetzbar.

Release Instructions and Limitations (RiLI):

- Die RiLI enthalten alle notwendigen Informationen zur Inbetriebnahme der Applikation.
- Die RiLI erklären klare und korrekte Annahmen und Limitierungen der Applikation.
- Die RiLI sind verständlich formuliert, so dass auch ein Benutzer mit wenig Erfahrung die Applikation Starten und bedienen kann.

Unit Tests (UT):

- Die Tests erfüllen das First Prinzip (Fast, Independent, Repeatable, Self-validating, Timely).
- Die Tests sind verständlich und gut dokumentiert.
- Die im Testkonzept definierten Tests bestätigen die erwarteten Funktionalitäten.
- Die Testabdeckung ist wie im Testkonzept definiert erreicht.
- Die Tests zeigen eine hohe Abdeckung von Funktionen und Logik.

Releasenotes

Benutzung

Zu dieser Dokumentation haben Sie ebenfalls eine verpackte Datei namens „NameSplitter_Release.zip“ erhalten. Wird diese entpackt, so erhalten Sie einen Ordner, welcher die Dateien „NameSplitter_Release.exe“, „NameSplitter_Release.jar“ sowie „NameSplitter_Release.bat“ enthält. Die Datei „NameSplitter_Release.bat“ können Sie nun durch einen doppelten Linksklick öffnen. Dadurch wird das Backend („NameSplitter_Release.jar“), sowie das Frontend („NameSplitter_Release.exe“) in jeweils einzelnen Prozessen gestartet. Nun öffnet sich die grafische Benutzeroberfläche, mit welcher Sie nun Ihre Namen parsen und zur Laufzeit speichern können.

Im Folgenden bekommen Sie einen Einblick in die Benutzung der grafischen Benutzeroberfläche:

Backend ist nicht erreichbar:

Falls das Backend nicht erreichbar ist, bekommen Sie eine Fehlermeldung. Um die Erreichbarkeit manuell zu testen, können Sie in einem beliebigen Browser die URL <http://localhost:8080/api/status> aufrufen. Erhalten Sie nicht die Rückgabe „Online“, so starten sie die Datei „NameSplitter_Release.bat“ erneut.

Titel	Vorname	Nachname	Briefanrede	Geschlecht
-------	---------	----------	-------------	------------

Hauptanzeige

Dies ist das initial geöffnete Fenster, nachdem die Applikation gestartet wurde. Hier haben Sie verschiedene Möglichkeiten, um den NameSplitter zu verwenden. Anhand der in rot dargestellten Zahlen werden die Funktionalitäten kurz erläutert:

1: Innerhalb dieser Textbox können Sie Ihre Eingabe tätigen. Durch einen Klick auf den Button „Parsen“ können Sie anschließend die Eingabe formatieren, wodurch sich ein weiteres Fenster öffnet. Eine manuelle Eingabe können Sie durch das Klicken auf den Button „Manuelle Eingabe“ machen.

2: Wird eine Eingabe geparsed und anschließend gespeichert, so werden alle zur Laufzeit gespeicherten Eingaben innerhalb dieser Tabelle ersichtlich sein. Diese Einträge können Sie durch das Klicken des Buttons „Zurücksetzen“ löschen, wodurch alle Einträge innerhalb dieser Tabelle

verschwinden werden. Möchten Sie eine gespeicherte Eingabe noch einmal bearbeiten, so können Sie auf den gewünschten Eintrag einen Doppelklick ausführen. Dadurch erscheint ein weiteres Fenster, innerhalb welchem Sie das ausgewählte Element anpassen können.

3: Hier erhalten Sie einen Überblick über alle bisher bekannten Titel. Um einen weiteren hinzuzufügen, klicken Sie auf den Button „Hinzufügen“, wodurch sich ein weiteres Fenster öffnet. Das Löschen eines Titels erfolgt ebenfalls hier durch einen Doppelklick auf den Titel. Anschließend erscheint ein Bestätigungsfenster.

NameSplitter

1 Eingabe:

Manuell eintragen

Durch einen Doppelklick auf einen Eintrag innerhalb der Tabelle, können Sie Ihre Einträge anpassen.

Titel	Vorname	Nachname	Briefanrede	Geschlecht
Dr.	Max	Mustermann	Sehr geehrter Herr Dr. IV	Männlich

2

Durch einen Doppelklick auf ein Titel können Sie diesen löschen


Verfügbare Titel

- Dr.rer.nat.
- Dr.med.
- Dr.phil.
- Dr.rer.pol.
- Dr.oec.
- Dr.h.c.mult.
- Dr.h.c.
- Dr.-Ing.
- Prof.
- Prof.
- Dr.

3

Rückgabe des Parsers

Dieses Fenster erscheint, nachdem innerhalb der Hauptanzeige ein gespeichertes Element bearbeitet, eine Eingabe geparsed oder eine manuelle Eingabe erfolgen soll. Der Titel „Das Parsen war erfolgreich“ signalisiert den Status, nachdem eine Eingabe erfolgt ist. Darunter stehen die Fehler, falls welche aufgetreten sind. Je nach Fehler wird die darunter stehende Eingabe farblich markiert, um dem Benutzer zu signalisieren, wo der Parser fehlgeschlagen ist. Ansonsten können Sie hier einen Vor- und Nachnamen, das Geschlecht, sowie Titel eingeben. Innerhalb des Dropdowns für die Titel ist das Element „-Keine Auswahl-“, wodurch Sie signalisieren, dass dieses einzelne Dropdown nicht mehr beachtet werden soll. Nach dem Speichern wird nicht nur die Tabelle innerhalb der Hauptanzeige, sondern auch die standardisierte Briefanrede generiert.

 Rückgabe des Parsers — □ ×


Das Parsen war erfolgreich

Im Folgenden stehen alle gefundenen Elemente, welche Sie nun noch vor dem Speichern anpassen können.


Eingabe:

Dr. Max Mustermann


Vorname




Nachname:




Geschlecht:

 ☒ Männlich ☐ Weiblich ☐ Divers

 ▼

Titel löschen

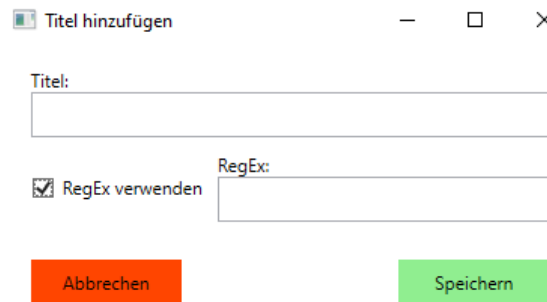
Möchten Sie einen vorhandenen Titel zur Laufzeit löschen, so können Sie dies durch einen Doppelklick auf diesen innerhalb des Hauptfensters machen. Anschließend öffnet sich ein Bestätigungsfenster, innerhalb welchem Sie den ausgewählten Titel final löschen können.

 Titel löschen — □ ×

Sind Sie sicher, dass sie "Dr.rer.pol." löschen möchten?

Titel hinzufügen

Innerhalb des Hauptfensters können Sie ebenfalls Titel hinzufügen. Dadurch öffnet sich ein weiteres Fenster. Innerhalb diesem können Sie den gewünschten Titel eingeben. Durch die Checkbox „RegEx verwenden“ erscheint eine weitere Textbox. Innerhalb dieser können Sie einen Regex definieren, mit welchem der NameSplitter beim Parsen nach dem Titel suchen soll. Standardmäßig sucht dieser nur den Titel, wenn er alleinsteht. Wird die Eingabe gespeichert, so erscheint Ihr Titel innerhalb der Tabelle „Verfügbare Titel“ in dem, Hauptfenster.



The screenshot shows a small window titled "Titel hinzufügen" with standard Windows window controls (minimize, maximize, close). Inside the window, there is a label "Titel:" followed by a text input field. Below this, there is a checked checkbox labeled "RegEx verwenden" and a label "RegEx:" followed by another text input field. At the bottom of the window, there are two buttons: "Abbrechen" (red) and "Speichern" (green).

Folgende Funktionalitäten werden durch das Backend unterstützt:

Einschränkungen