Inhalt

[1 Aufgabenstellung 2](#_Toc479235395)

[1.1 Zielbestimmung 2](#_Toc479235396)

[1.1.1 Java 2](#_Toc479235397)

[2 Produktumgebung 3](#_Toc479235398)

[2.1 Anwendungsbereich 3](#_Toc479235399)

[2.1.1 Java 3](#_Toc479235400)

[2.2 Anwendergruppen 3](#_Toc479235401)

[2.2.1 Java 3](#_Toc479235402)

[2.3 Basismaschine 3](#_Toc479235403)

[2.4 Betriebsbedingungen (Mengengerüst) 3](#_Toc479235404)

[3 Produktmodell / Produktfunktionen 3](#_Toc479235405)

[3.1 Java 4](#_Toc479235406)

[3.1.1 Anwendungsfälle der Software 4](#_Toc479235407)

[/F1010/ Auswahl des Active-Directory-Profils 4](#_Toc479235408)

[/F1020/ Hinzufügen von Benutzern 4](#_Toc479235409)

[/F1030/ Entfernen von Benutzern 4](#_Toc479235410)

[/F1040/ Editieren der Gruppenmitgliedschaften 5](#_Toc479235411)

[/F1050/ CSV-Export 5](#_Toc479235412)

[3.1.2 Ablaufanalyse 5](#_Toc479235413)

[3.1.3 Strukturelle Datenanalyse 5](#_Toc479235414)

[4 Qualtätsanforderungen 7](#_Toc479235415)

[4.1 Java 7](#_Toc479235416)

[4.1.1 Funktionalität 7](#_Toc479235417)

[4.1.2 Skalierbarkeit 7](#_Toc479235418)

[4.1.3 Zuverlässigkeit 7](#_Toc479235419)

[4.1.4 Performanz 7](#_Toc479235420)

[4.1.5 Verfügbarkeit 7](#_Toc479235421)

[5 Benutzerschnittstelle 8](#_Toc479235422)

[5.1 jAD-Editor 8](#_Toc479235423)

[6 Entwicklungsumgebung 11](#_Toc479235424)

[6.1 Software 11](#_Toc479235425)

[6.1.1 Integrierte Entwicklungsumgebung 11](#_Toc479235426)

[6.1.2 Grafische Benutzeroberfläche 11](#_Toc479235427)

[6.1.3 UML-Editor 11](#_Toc479235428)

[7 Abkürzungsverzeichnis 11](#_Toc479235429)

[8 Literaturverzeichnis 11](#_Toc479235430)

# Aufgabenstellung

Erstellt von Wenderlich und Schmidt.

## Zielbestimmung

### Java

#### Muss-Kriterien

Bis zum 21.04.2017 soll eine Java-Anwendung mit Benutzeroberfläche erstellt werden, die das Anlegen neuer Active-Directory Benutzer und deren Attribute ermöglicht.

Die Bearbeitung erfolgt dabei in einer tabellarischen Darstellung, die resultierende Tabelle kann zur Weiterbearbeitung in einer CSV-Datei gespeichert werden.

Die Software liegt dem Kunden am 21.04.2017 zur Ansicht und Einarbeitung inkl. Dokumentation vor und wird von den Entwicklern am Dienstag, dem 25.04.2017, den Kunden vorgestellt.

#### Kann-Kriterien

Das Zwischenspeichern der aktuellen Sitzungsdaten in einem festen Intervall soll ermöglicht werden.

Ebenso die Sortierung der einzelnen Tabellenspalten im Editorfenster.

Copy-und-Paste sowohl einzelner Werte als auch der ganzen Tabelle kann eingefügt werden.

#### Abgrenzung

Das Programm ist nicht dazu geeignet, eine direkte Bearbeitung des Active-Directorys vorzunehmen.

Ein Einlesen einer Tabelle zur Weiterverarbeitung ist nicht vorgesehen.

Es erfolgt keine Ausgabe in Tabellenkalkulationsformate oder in Datenbankformate.

# Produktumgebung

## Anwendungsbereich

### Java

Mittels der Programmiersprache Java soll eine Software (Arbeitstitel: jAD-Editor V.0.1) erstellt werden, die das bequeme Anlegen neuer Active-Directory Benutzer und bequeme Bearbeiten der dazugehörigen Active- Directory Attributen ermöglicht.

Die Attribute liegen dabei in Konfigurationsdateien dem Programm bei und bilden dabei den Rahmen für die einzugebenden Benutzer.   
Die resultierende Tabelle kann im CSV-Format gespeichert und in das Active-Directory mittels csvde.exe importiert werden.

## Anwendergruppen

### Java

Anwendergruppe des jAD-Editors sind in der Administration des Active-Directorys unerfahrene Anwender, die über eine gewohnte, dem Industriestandard entsprechende, Benutzeroberfläche arbeiten können.   
Die Zielplattform sind aktuelle Versionen der Windows Betriebssysteme

## Basismaschine

Java 8 JRE ist Voraussetzung ( Anwendung von Lambda-Ausdrücken …. ).   
Soll vor allem auf Windows-BS zum Einsatz kommen, da die Zielgruppe damit arbeitet.

Tabelle 1: Mindestanforderungen / empfohlene Konfiguration

|  |  |
| --- | --- |
| Softwarevoraussetzungen. | Java 8 |
| Betriebssystem | Windows, Linux, MacOS |
| Hardware | X86 - Prozessoren  (Pentium oder höher; AMD kann auch funktionieren, wird aber nicht getestet)  TODO |
| RAM | 512 MB oder höher |

## Betriebsbedingungen (Mengengerüst)

MB ein, das Programm inkl. aller benötigten Bibliotheken und Konfigurationsdateien wird 30 MB nicht überschreiten.

Das JRE, das Voraussetzung für den Betrieb von Java-Programmen ist, benötigt noch einmal TODO

# Produktmodell / Produktfunktionen

## Java

### Anwendungsfälle der Software

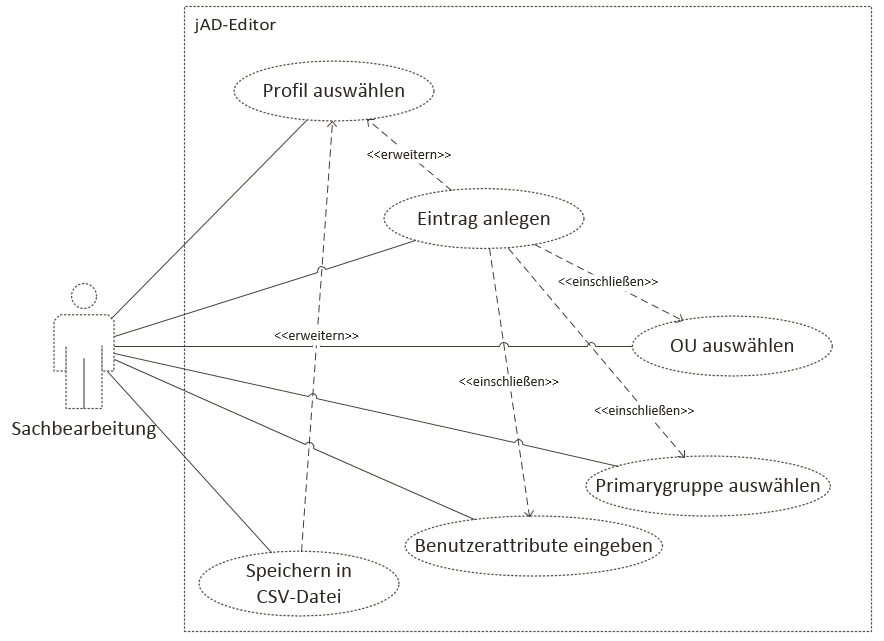


Abbildung 1: USE-Case Diagramm für die Anwendung jAD-Editor, V. 0.1

### /F1010/ Auswahl des Active-Directory-Profils

Es werden verschiedene Profile zur Verfügung gestellt, die die zur Verfügung stehenden Attribute definieren und die verschiedene Voreinstellungen beinhalten, wie den FQDN (Fully Qualified Domain Name).  
Ein beschreibendes Feld wird für jedes Profil angezeigt.

### /F1020/ Hinzufügen von Benutzern

Das Hinzufügen von Benutzern erlaubt das Editieren aller festgelegten Attribute des zu erzeugenden Nutzers.

### /F1030/ Entfernen von Benutzern

Die Funktion entfernt den jeweils ausgewählten Benutzer.

### /F1040/ Editieren der Gruppenmitgliedschaften

Den einzelnen Nutzern lassen sich ihre primären Active-Directory Gruppen-Mitgliedschaften zuordnen.

### /F1050/ CSV-Export

Die erstellte Nutzertabelle kann in eine CSV-Datei exportiert werden.   
Die erstellte Datei ist einlesbar mit csvde.exe. enthalt also alle Attribute die zum Hinzufügen von Active-Directory Objekten notwendig sind.

### Ablaufanalyse

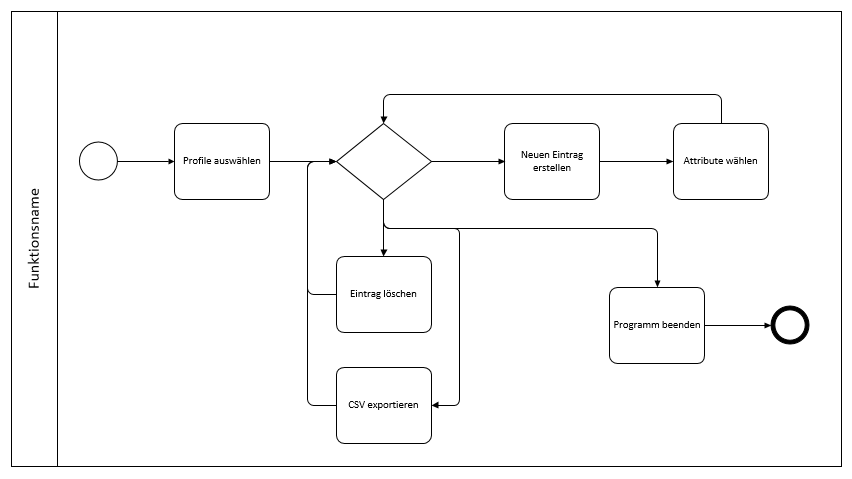


Abbildung 2: Grundlegender Programmablauf

Das Programm startet mit einem Profilauswahldialog.

Ausgehend von der Profilauswahl öffnet sich das Hauptfenster auf denen die Eingaben vorgenommen werden können (siehe 5.1). Hier wählt der Nutzer zwischen dem Hinzufügen und dem Löschen der eingegebenen Einträge aus, außerdem erstellt man an dieser Stelle die Ausgabedatei.

### Strukturelle Datenanalyse

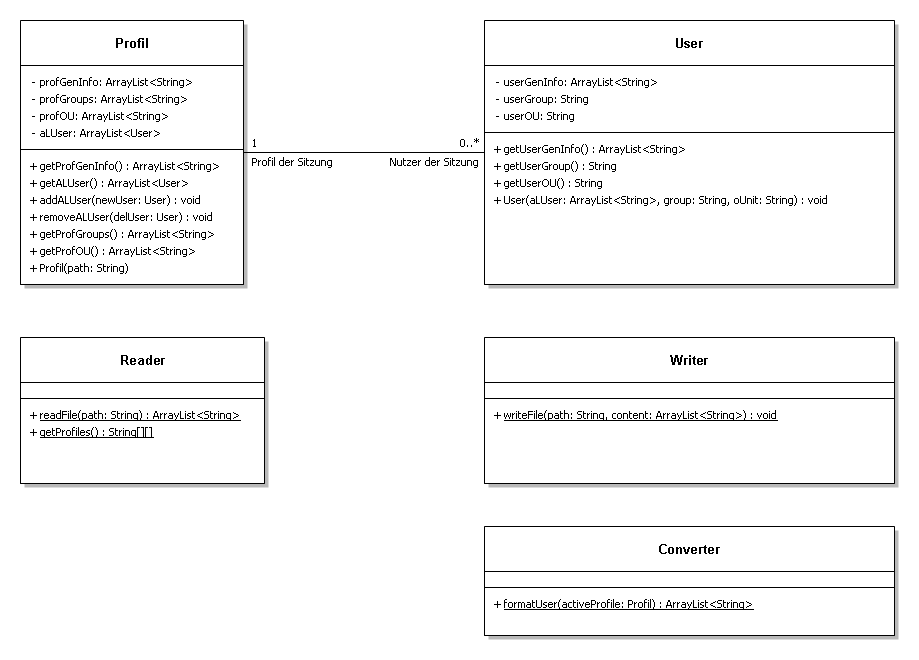


Abbildung 3 zugrunde liegende Klassenmodell

Abbildung 3 verweist zeigt das grundlegende Klassenmodell.

Die Klasse Profil erstellt ein Objekt, in dem die gelesenen Daten des gewählten Profils enthalten sind.  
Die Daten bestehen aus den verfügbaren Benutzerattributen, möglichen Gruppen und OUs. Zudem werden hier die bisher angelegten Nutzer der aktuellen Sitzung zwischengespeichert.

Die Nutzer werden durch Objekte der Klasse User erzeugt.

Sie enthalten die einzelnen Attribute, seine Primäre Gruppe und seine OU.

Die Klassen Converter und Writer sind für die Formatierung und Ausgabe der CSV-Daten zuständig, die Klasse Reader liest die Informationen der Profilkonfigurationsdateien aus.

# Qualtätsanforderungen

## Java

Da jAD-Editor als benutzerfreundliche Software zum Anlegen neuer Active-Directory Benutzer gedacht ist, sollte auf Anwenderfreundlichkeit und einen hohen Wiederkennungswert besonderer Wert gelegt werden.

Andere Eigenschaften, wie eine besondere Performanz, treten dabei in den Hintergrund.

Eine Skalierbarkeit, im Sinne einer einfachen Erweiterbarkeit, wird während der Entwicklung berücksichtigt, ist aber kein Hauptaugenmerk.

Tabelle 2: Qualitätsanforderungen, Stand 4.4.17

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Sehr wichtig | Wichtig | Weniger wichtig | Unwichtig |
| Benutzerfreundlichkeit | X |  |  |  |
| Skalierbarkeit |  | X |  |  |
| Funktionalität |  | X |  |  |
| Zuverlässigkeit |  | X |  |  |
| Performanz |  |  | X |  |
| Verfügbarkeit |  |  | X |  |

### Funktionalität

Alle im Lastenheft aufgeführten Funktionen werden vollständig und angemessen umgesetzt.

Die Software wird ein eigenständiges System und kann lokal auf den gewünschten Rechnern gespeichert werden, unter Berücksichtigung der unter 2.4 geschilderten Voraussetzungen.

### Skalierbarkeit

Der Aufbau des Programms ist so gehalten, dass es leicht um weitere Funktionen ergänzt werden kann.

### Zuverlässigkeit

Um die geforderte fehlerfreie Funktion des Programms zu gewährleisten wird vom Entwickler eine geeignete Prüfung (logisch oder praktisch) gewählt, durchgeführt und dokumentiert

### Performanz

Das Programm wird von einem Benutzer bedient. Eine Skalierbarkeit kann durch Speicherung auf mehreren Rechnern und dementsprechende Aufteilung der zu anlegenden Benutzer erreicht werden.

Dabei werden ebenso viele Dateien, wie genutzte Rechner erzeugt.

### Verfügbarkeit

Da die Software auf einem Einzelplatzrechner lauffähig ist und keine Daten geladen werden müssen, hat sie eine hohe Verfügbarkeit.

# Benutzerschnittstelle

## jAD-Editor

Essentiell für das Programm ist das Benutzen von Profilen, welche für jeweilige Domänen und Einstellungen Voraussetzung sind.

Dies wird beim Starten über die Profilauswahl realisiert.

In dem Fenster „Profil auswählen“ (Abbildung 1) werden die vom Administrator angelegten Profile angezeigt. Zusätzlich gibt es die Möglichkeit eine Kurzbeschreibung zu dem Profil ausgeben zu lassen um Verwirrung vorzubeugen.

Ist das passende Profil gewählt, wird mit „OK“ fortgefahren.

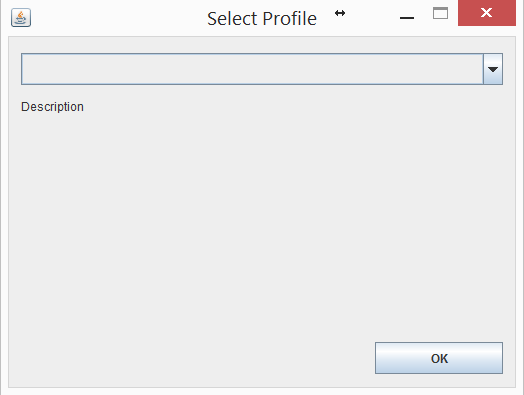


Abbildung 4: Das Profilselektionsfenter

Mit den Profilinformationen wird das Hauptfenster von jAD-Editor (Abbildung 2) erzeugt.

Hauptaugenmerk liegt hier auf der Tabelle basierend auf dem ausgewählten Profil. Im Initialisierungszustand wird die Tabelle leer sein.

Es stehen vier Buttons zur Verfügung:

* Mit dem „Eintrag hinzufügen“ Button kann ein neuer Eintrag in die Tabelle eingefügt werden. Ein Aktivieren führt zum Fenster „Eintrag hinzufügen“ (Abbildung 3).
* Durch das klicken auf „Eintrag entfernen“ werden die selektierten Einträge in der Tabelle gelöscht. Wird nur durch ein Bestätigen ausführbar sein.
* About zeigt Informationen über das Programm an ( TODO: Link Abbildung 5).
* Der Button „Export“ erstellt aus den Informationen der Tabelle eine weiter verwertbare CSV Datei.

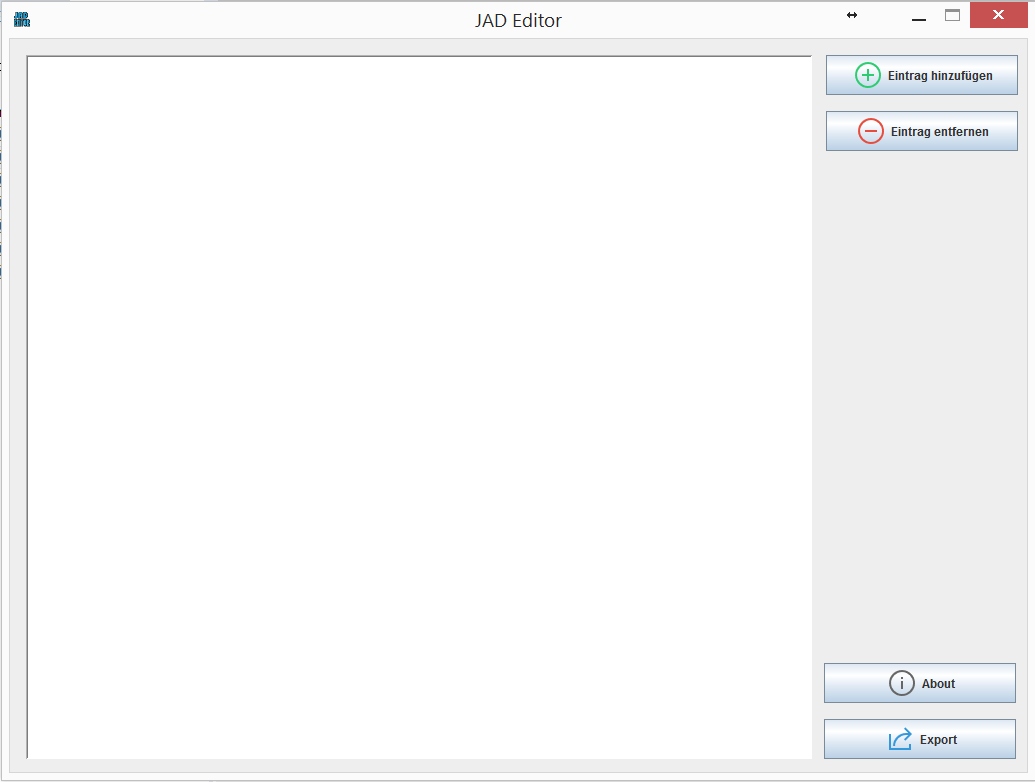


Abbildung 5: Das Hauptfenster

Das „Eintrag hinzufügen“ Fenster ist das Grundelement für einen neuen Eintrag.

Es wird dynamisch erzeugt auf Grundlage des Profils. Die Auswahlbox enthält die möglichen OUs in der der Benutzer später angelegt werden soll. Der Button „Gruppen…“ erzeugt ein neues Fenster (Abbildung 4).

Das Bestätigen mit „OK“ erzeugt aufgrund der eingetragenen Informationen einen neuen Eintrag.

Abbildung 6: Eintrag hinzufügen

Das „Gruppen verwalten“-Fenster (Abbildung 4) macht es möglich, für einen späteren Benutzer die Zugehörigkeit zu Gruppen festzulegen. Multiple Auswahl ist möglich.

Sind die gewollten Gruppen ausgewählt, können die Einstellungen mit „OK“ übernommen werden.

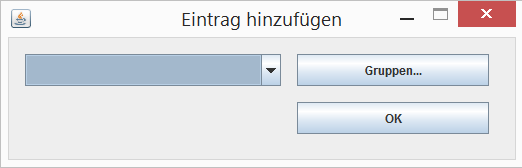
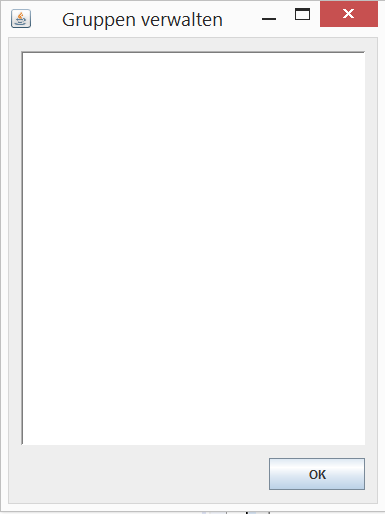


Abbildung 7: Gruppen verwalten

Mit dem klicken auf „About“ im Hauptfenster (Abbildung 2) wird das dazugehörige Fenster (Abbildung 5) erzeugt.

Hier stehen simple Informationen über das Programm.

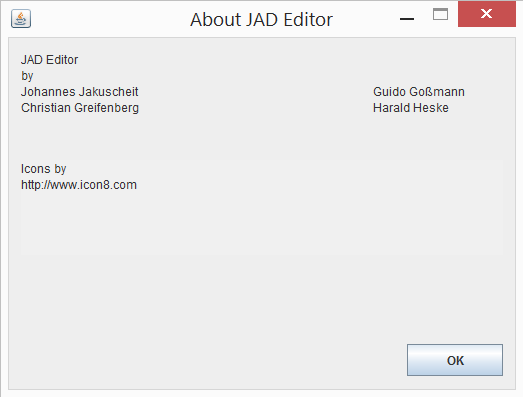


Abbildung 8: Fenster "About"

# Entwicklungsumgebung

## Software

### Integrierte Entwicklungsumgebung

Eclipse, Version: Neon.3 Release (4.6.3, Build id: 20170314-1500)

### Grafische Benutzeroberfläche

Pencil, Version 3.01, Skizzierung des GUI-Prototyps

WindowBuilder, Version 4.6, Eclipse Neon.2 Addon

### UML-Editor

NClass, Version 2.04

### Versionsverwaltungssystem

GitHub, sowie

GitHub Desktop in der Version 3.3.4.0 (“chocolate covered yaks”)

# Abkürzungsverzeichnis

AD Active Directory

FQDN Fully Qualified Domain Name

GUI Graphical User Interface

OU Organisationseinheit (Organisation Unit)

# Literaturverzeichnis

Was kommt hier herein?  
(Literatur, klar, aber welcher Art soll die sein? Sicher keine wissenschaftliche Fachartikel)

(DRI) Doku-Richtlinie für IT-Projekte, Version 2.4, bib international college

(P15) Albernes Pflichtenheft der Vorgängerklasse, Schatz, Schleimer, Schluffi et al.