**Bericht zur Anwendung zum erkennen von Dubletten**

Für die Umsetzung der Anwendung wurde das Slim Framework in der Version 4 verwendet.

Um eine schnelle Einrichtung, inklusive Error Handling und Logging, zu erhalten wurde eine Skeleton Applikation aufgesetzt und angepasst. Außerdem war die Ornerstruktur hier bereits auf Benutzerverwaltung ausgelegt. Ein entscheidender Punkt für die Auswahl liegt auch in der einfachen Erweiterbarkeit, falls benötigt.

Details zum Starten und Verwendenden der Anwendung sind in der Datei README beschrieben. Vorrausgesetzt wird PHP in der Version 8. Zum besseren Testen der Funktionalität wird die MySQL-anbindung unabhängig Konfiguriert.

Zur Authentifizierung muss in der URL vor dem Aufruf der eigentlichen Aktion Benutzername und Passwort angegeben werden.

Zur Umsetzung der Anforderungen wurde die ListUserDoubletsAction hinzugefügt. Diese greift auf das neu Angelegte Repository für die Benutzerdaten zu. Hier werden die Daten aus der Datenbank geladen, verarbeitet und zurückgegeben. Die selektierten Daten werden dann als XML ausgegeben.

Der Algorithmus sucht nach einer Übereinstimmung von Vornamen, Nachnamen und Geburtsdatum. Hierfür werden in einer SQL-Abfrage nur diese 3 Felder selektiert. Im Fall von mindestens einer gefundenen Doublette, werden alle Daten der gefundenen Personen Selektiert. Bei der Ausgabe werden die Daten der einzelnen Personen jeweils unter den gemeinsamen Vornamen, Nachnamen und Geburtsdatum angezeigt.

Zur Individualisierung könnten optional weitere Spalten zur Feststellung von Doubletten als Parameter bei der Abfrage mit angegeben werden. Außerdem ist ein Abgleich des Geburtsnamens, falls vom Nachnahmen abweichend, denkbar. Für den weiteren Umgang mit den gefundenen Dubletten können auch Daten zur Aktualität bzw. letzten Aktivität ausgewertet werden, um schneller zu erkennen, welches Duplikat ggf. eine Datenleiche ist. Für diese wenig Komplexe Anforderung an das Datenbankmanagement habe ich mich für eine einfach PDO-MySQL Anbindung entschieden. Für eine Umfangreichere Anwendung würde ich ein ORM wie Doctrine einsetzen. Die Überprüfung der Benutzer und Ihrer Rechte kann(und sollte) noch verbessert werden.

Nach Durchsicht der Möglichkeiten für Benutzerrechte und Anmeldung bin ich davon ausgegangen, dass dies den geforderten Zeitraum überschreitet. Deshalb habe ich mich für eine einfache Angabe von Benutzernamen und Passwort in der URL entschieden. Mir ist bewusst, dass dies keine Lösung für einen Live-Betrieb ist.

Da ich durch das fehlende Rechtemanagement meine eigenen Zielvorgaben nicht erfüllen konnte, kann ich bei dieser Aufgabe nicht von Stolz sprechen. Im allgemeinen bin ich auf meine Arbeit nur Stolz, wenn Anforderungen eines Auftraggebers übertroffen werden.