



# Hochschule Macromedia für angewandte Wissenschaften University of Applied Sciences

Kursbezeichnung: (Aufbaumodul Programmierung)

Prüfer: (Prof. Berger und Prof. Brunner )

**Vom Studierenden auszufüllen:**

300834

Matrikelnummer

F-UBf DT UID 6d 22w

Kohorte

Hertfelder

Nachname

Victoria

Vorname

Die Arbeit wird eingereicht als:

(Tragen Sie bitte in die zutreffende Box den Buchstaben X ein)

☒

Einzelarbeit

☐

Gruppenarbeit

**Trifft nur auf Gruppenarbeiten zu:** (Nur bei Gruppenarbeiten auszufüllen)

Falls Sie eine Gruppenarbeit einreichen, dann müssen bitte die Vor- und Nachnamen aller Gruppenmitglieder aufgeführt werden. Die Namen sind von den jeweiligen Gruppenmitgliedern selbst elektronisch einzutragen. Durch Eintrag des Namens wird bestätigt, dass der/die Studierende mit der Abgabe der Arbeit in der vorliegenden Form einverstanden ist. Ferner wird mit Eintrag des Namens erklärt, die Projektarbeit (bei Gruppenarbeit: den von dem/der jeweiligen Studierenden erstellten und entsprechend gekennzeichneten Teil der Arbeit selbstständig und ohne fremde Hilfe angefertigt zu haben. Dabei hat der/die Studierende sich keiner anderen Hilfsmittel bedient als derjenigen, die im beigefügten Quellenverzeichnis genannt sind. Alle Stellen, die wörtlich oder sinngemäß aus Veröffentlichungen entnommen wurden, sind von dem/der Studierenden als solche kenntlich gemacht. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen. Mit der Abgabe stimmen die Gruppenmitglieder zu, dass alle Bewertungen und Hinweise der Prüfer:innen in der hochgeladenen Arbeit hinterlegt werden. Das Gruppenmitglied, das den Upload vorgenommen hat, muss die Korrekturhinweise, den anderen Gruppenmitgliedern zugänglich machen.

1)

2)

3)

4)

5)

6)

7)

8)

**Bewertung der Gruppenarbeiten:**

(Tragen Sie bitte in die zutreffende Box den Buchstaben X ein)

☐

Ich beantrage bei meiner Gruppenarbeit eine Individualbewertung

☐

Ich beantrage bei meiner Gruppenarbeit eine Gruppenbewertung

Eine Individualbewertung bedeutet, dass jedes Gruppenmitglied eine individuelle Note bekommt und eine Gruppenbewertung bedeutet, dass jedes Gruppenmitglied eine identische Note bekommt.

Freiburg, 22.06.23

Ort/Datum

Hertfelder, Victoria

Vollständiger Vor- und Nachname

**Bewertung (gemäß Notenschlüssel), Ergebnis Erstprüfung: Gesamtpunkte:** \_\_\_\_\_**Datum:****Name, Vorname Erstprüfender (elektronisch auszufüllen)****Vom Prüfer auszufüllen:** (Freitext für die Zweitkorrektur)

## Inhalt

1.	Projektkonzept.....	3
1.1.	Wieso ich dieses Projekt gewählt habe? .....	3
2.	Vorgehensweise.....	4
3.	Fazit.....	6
4.	Link zu GitHub.....	6

## 1. Projektkonzept

### 1.1. Wieso ich dieses Projekt gewählt habe?

Die Idee im ersten Semester bestand darin, einen Passwortgenerator zu entwickeln, der auf Funktionen basiert. Dieses Programm generiert Passwörter beliebiger Länge, indem es Buchstaben, Zahlen und Sonderzeichen kombiniert. Um den individuellen Anforderungen gerecht zu werden, kann der Benutzer den Zeichenvorrat selbst festlegen. Die generierten Passwörter wurden mit den entsprechenden Benutzern und Anwendungen verknüpft und als Datensatz in einer CSV-Datei gespeichert.

Im zweiten Semester liegt der Schwerpunkt der Abschlussarbeit hauptsächlich auf der Implementierung der bestehenden Funktionen in geeignete Klassen. Das neue Konzept basiert auf der objektorientierten Programmierung (OOP) und zielt darauf ab, den Code besser zu strukturieren und die Wiederverwendbarkeit zu verbessern. Zusätzlich soll das alte SCV-Format durch JSON ersetzt werden.

## 2. Vorgehensweise

Um meine Aufgabe effizient zu bewältigen, bin ich dem folgenden Plan gefolgt:

**Schritt 1:** Einen Überblick über die alten Funktionen verschaffen und eine Liste der Funktionen erstellen:

- Die Funktion "**Zeichenvorrat**" wurde implementiert, um den Zeichenvorrat basierend auf den vom Benutzer ausgewählten Optionen (Großbuchstaben, Kleinbuchstaben, Zahlen, Sonderzeichen) zusammenzustellen.
- Die Funktion "**psw\_erzeugen**" generiert ein Passwort mit einer bestimmten Anzahl von Zeichen aus dem Zeichenvorrat.
- Die Funktion "**psw\_speichern**" speichert den generierten Datensatz, bestehend aus einem Titel, Benutzer und Passwort, in einer Passwortliste.
- Die Funktion "**psw\_liste\_anzeigen**" zeigt die gespeicherten Passwortlisten an.
- Die Funktionen "**get\_psw**" und "**view\_psw**" wurde implementiert, um einen bestimmten Datensatz aus der Passwortliste zu holen und benutzerfreundlich darzustellen.

Nachdem ich die Funktionen analysiert habe, konnte ich feststellen, dass einige von denen zusammengefasst werden können, um sie in entsprechende Klassen zu implementieren.

**Schritt 2:** Funktionen den entsprechenden Klassen zuordnen. Basierend auf der Analyse wurden drei Hauptklassen erstellt: "JsonFile", "CharacterSet", und "PasswordManagement". "CharacterSet" erbt von "JsonFile", während "PasswordManagement" sowohl von "JsonFile" als auch von "CharacterSet" erbt. Diese Strukturierung verbessert die Code-Organisation und erhöht die Wiederverwendbarkeit.

In der Klasse "**JsonFile**" wird die Datenverwaltung mit JSON erleichtert. Sie ermöglicht das Einlesen des Zeichenvorrats und das Speichern sowie Laden der Passwortlisten als JSON-Dateien aus den untergeordneten Kindklassen.

Die Klasse "**CharacterSet**" erbt von "JsonFile" und ist verantwortlich für die Auswahl und Zusammenstellung des benutzerdefinierten Zeichenvorrats. Sie übergibt den erstellten Zeichenvorrat an die nächste Klasse.

Die Klasse "**PasswordManagement**" erbt sowohl von "JsonFile" als auch von "CharacterSet" und generiert aus dem Zeichenvorrat Passwörter. Diese Passwörter werden zusammen mit Benutzer, Passwort und

Anwendung als Datensatz gespeichert und in Form einer JSON-Datei abgelegt. Zusätzlich kann die Klasse die Liste aller Passwortsätze darstellen.

Mit diesem Klassenkonzept konnte ich erfolgreich zu dem nächsten Schritt übergehen.

Schritt 3: Das Projekt wurde als Paket verpackt und auf GitHub veröffentlicht, um anderen Nutzern den Zugriff und die Nutzung zu ermöglichen. Das Ziel besteht darin, das Paket auf GitHub zu sichern, um es für zukünftige Verwendung und Weiterentwicklung bereitzuhalten.

Schritt 4: Zuletzt habe ich mich mit virtuellen Umgebungen auseinandergesetzt und mein Projekt erfolgreich von der globalen Umgebung in eine lokale virtuelle Umgebung verschoben. Dabei habe ich eine requirements.txt-Datei erstellt und mein Projekt auf GitHub aktualisiert, um alle erforderlichen Abhängigkeiten für das Projekt zu dokumentieren.

### 3. Fazit

Durch die Umsetzung dieses Plans ist der Passwortgenerator zu einer gut strukturierten und wiederverwendbaren Anwendung geworden. Darüber hinaus bietet das Projekt Potenzial für eine Weiterentwicklung und Erweiterung in Richtung Verschlüsselung.

### 4. Link zu GitHub

Link zu GitHub: [https://github.com/MacroVictoria/password\\_package.git](https://github.com/MacroVictoria/password_package.git)