

Gruppe E (Ely Schybol, Ruben Singer, Coffi Akplogan, Nour Hashem, Maximilian Schwarz)

## **SoftwareTechnik PVL Projekt – Evolutionäre Algorithmen**

### **Nutzergruppen**

Liste und kurze Erläuterung

### **Funktionalitäten**

Hinzufügen zu der Nutzergruppen Liste

### **Welche Nutzergruppen brauchen welche Funktionalitäten ?**

Oben in der Nutzergruppen Liste beschreiben

### **Wie sehen die Funktionalitäten genau aus ?**

mindestens 5 Funktionalitäten sollen per Use-Case-Formular beschrieben werden.

Sonstiges:

- Skizze einer wichtigen Ansicht der GUI als Papierprototyp (GUI-Mockup, Wireframe),
  - (ggf. Visualisierung von Abläufen per Aktivitätsdiagramm, falls dies dem Verständnis hilft.)
- 

### **Nutzergruppen**

Enduser oder anonyme Nutzer mit wenig Rechten

Premium Enduser die für volle User Funktionalität bezahlen

Admins zur Verwaltung und Pflegen der App

(→ Rollen für Studierende & Profesor\*innen)

### **Funktionalitäten**

- Individuen sind binäre Zeichenketten fester Länge (16bits)/ **Individuen erstellen**
- Mutationen sind durch Veränderung der festen Zeichenketten zulässig → Rekombinationsoperatoren sind durch Bitoperatoren implementiert ( $\&$ (And),  $\mid$ (Or),  $\wedge$ (Xor))
- Mutationen werden durch „Normale“ Zeichenketten abgegrenzt.
  - Festlegung der Mutationskriterien (!)
- Zugriff auf verschiedene Evolutionsalgorithmen (Premium Nutzer haben Zugriff auf alle Algorithmen)
- Konfiguration vom ausgewählten Algorithmus
- Grafische Darstellung der Algorithmen
- Grafik der Statistik von Ablauf der Optimierung von den Algorithmen (?) (Premium Nutzer)
- Herunterladen der Grafiken und Numerischen Darstellungen
- Hochladen und damit persistenten Speichern der Grafiken, Konfiguration von Abläufen und Numerischen Darstellung (für Premium Nutzer)
- Laden von Algorithmen und Konfigurationen
- Nutzung für GUI ausgelegt.
- Konfiguration der DatenspeicherMedien (Cloud, Formate) (Admin)
- Accountverwaltung (Admin)
- Erprobung ungewöhnlicher Abläufe (Experimentelle Stränge, Erhöhte Außeneinwirkungen → Extreme Parameter der Algorithmenfunktionen)

## **Functionality Use Cases**

---

### **Use Case 000**

---

Name:	Evolutions Algorithmen Simulation
Kurzbeschreibung:	Die Darstellung der Wirkungsweise von Evolutionären Algorithmen soll auf Individuen, welche als eine feste Bitfolge dargestellt werden, durchgeführt werden. Hierbei ist es möglich Entwicklungsstränge verschiedener Individuen zu beobachten, wobei gelegentlich Mutationen erfolgen können.
Akteure:	Forschende, Endnutzer
fachliche Auslöser:	Biologische Evolutionsforschung
normaler Ablauf:	<ul style="list-style-type: none"><li>- Evolutionsalgorithmus wird ausgewählt</li><li>- Mutationskonfiguration wird implementiert/definiert</li><li>- Auslesekongfiguration wird definiert</li><li>- Individuen werden erstellt</li><li>- Evolutionsalgorithmus startet und wirkt sich auf die Individuen aus</li><li>- Individuen vereinigen sich und bilden neue Stränge</li></ul>
alternativer Ablauf:	<ul style="list-style-type: none"><li>- Simulation wird beendet</li><li>- Mutationen treten auf und bilden einen neuen Evolutionsstrang oder</li><li>- Individuen sterben aus, sodass es keine Entwicklungsstränge mehr gibt</li></ul>
Vorbedingung:	<ul style="list-style-type: none"><li>- Ein vorheriger Evolutionärer Algorithmus ist abgeschlossen</li><li>- Individuen sind vorhanden</li><li>- Die Abgrenzungen zu Mutationen ist definiert</li><li>- Mutationsoperatoren sind definiert und deren Auftritt Wahrscheinlichkeit implementiert</li></ul>
Nachbedingungen:	<ul style="list-style-type: none"><li>- Ablauf und Konfiguration Daten sind erstellt und abrufbar</li></ul>

---

---

#### Use Case 001

---

Name:	Grafische Darstellung der Algorithmen
Kurzbeschreibung:	Der Verlauf von Evolutionären Algorithmen wird mittels Diagramme dargestellt,
Akteure:	Endnutzer
fachliche Auslöser:	Mathematik, Biologie (Evolutionäre Forschung)
normaler Ablauf:	<ul style="list-style-type: none"><li>- Ein Evolutionärer Algorithmus wird gestartet</li><li>- Die gewünschte Anzeigeform (Diagramtyp) wird ausgewählt</li><li>- Der Algorithmus wird abgebildet</li><li>- Das Programm wird gestoppt</li></ul>
alternativer Ablauf:	<ul style="list-style-type: none"><li>- Anzeigeform wird während eines Durchlaufs geändert</li></ul>
Vorbedingung:	<ul style="list-style-type: none"><li>- Evolutionärer Algorithmus muss ausgewählt werden</li></ul>
Nachbedingungen:	<ul style="list-style-type: none"><li>- Einstellen ob experimentelle Abläufe auf angewendeten Algorithmus ausgeführt werden sollen</li></ul>

---

---

#### Use Case 010

---

Name:	Speicherung der Daten in die Cloud
Kurzbeschreibung:	Die Numerischen Daten und Grafiken der evolutionärer Algorithmen werden in der Cloud gespeichert.
Akteure:	Premium Endnutzer
fachliche Auslöser:	Dezentrale Persistenz
normaler Ablauf:	<ul style="list-style-type: none"><li>- Daten werden erzeugt mittels eines Ablaufs</li><li>- Speicherfunktionalität wird ausgewählt</li><li>- Ablaufkonfiguration wird gesichert</li><li>- Daten und Grafiken, sowie Datei der Ablaufkonfiguration wird komprimiert</li><li>- Client logged sich in die Cloud ein</li><li>- Komprimierte Daten werden in die Cloud hochgeladen</li></ul>
alternativer Ablauf:	<ul style="list-style-type: none"><li>- Verbindungsprobleme sorgen für einen Abbruch des Hochladens, sodass nichts in der Cloud gespeichert werden kann</li><li>- Login in die Cloud schlägt fehl, sodass der Prozess abgebrochen werden muss</li></ul>
Vorbedingung:	<ul style="list-style-type: none"><li>- User Account muss vorhanden sein</li><li>- User Account muss Premium sein</li><li>- Internet Verbindung muss vorhanden sein</li></ul>
Nachbedingungen:	<ul style="list-style-type: none"><li>- User lädt Daten herunter</li></ul>

---

---

Use Case 011

---

Name:	Laden von Konfigurationen
Kurzbeschreibung:	Konfigurationen können aus der Cloud oder von der lokalen Festplatte aus geladen werden um damit Daten und Grafiken wiederherzustellen
Akteure:	Endnutzer, Admins
fachliche Auslöser:	Konsistenz
normaler Ablauf:	<ul style="list-style-type: none"><li>- Lade Funktionalität wird ausgewählt</li><li>- Konfigurationsdatei wird ausgewählt</li><li>- Konfigurationsdatei Integrität wird überprüft</li><li>- Daten werden aus der Konfigurationsdatei extrahiert</li><li>- Daten werden im Programm dargestellt</li></ul>
alternativer Ablauf:	<ul style="list-style-type: none"><li>- Konfigurationsdatei ist beschädigt</li></ul>
Vorbedingung:	<ul style="list-style-type: none"><li>- Konfigurationsdatei ist vorhanden</li><li>- Internetzugang ist vorhanden</li></ul>
Nachbedingungen:	<ul style="list-style-type: none"><li>- Konfig Dateien werden nachträglich umkonfiguriert</li></ul>

---

---

Use Case 100

---

Name:	Herunterladen der Daten
Kurzbeschreibung:	Erzeugte Daten und Grafiken werden auf das Gerät des Endnutzers heruntergeladen und damit persistent gespeichert
Akteure:	Endnutzer
fachliche Auslöser:	Lokale Persistenz
normaler Ablauf:	<ul style="list-style-type: none"><li>- Evolutionärer Algorithmus wurde durchlaufen</li><li>- Die Speicherungs Funktionalität wurde ausgewählt</li><li>- Ein Kontext Fenster öffnet sich</li><li>- Der Speicherort wurde durch das Kontext Fenster ausgewählt</li><li>- Daten werden auf die Festplatte heruntergeladen</li></ul>
alternativer Ablauf:	<ul style="list-style-type: none"><li>- Konfigurationen werden aus der Cloud geladen</li><li>- Grafik wurde erzeugt</li></ul>
Vorbedingung:	<ul style="list-style-type: none"><li>- Genügend Platz auf dem Speichermedium</li><li>- Daten zum Speichern sind vorhanden</li></ul>

-Daten können bearbeiten werden (dargestellt / umkonfiguriert )