

1. Beschreibung des Spiels (Projektidee)

Unser Projekt ist eine vereinfachte Version von „**Pictionary**“, bei dem eine gezeichnete Skizze von einem Neural Netz erkannt und klassifiziert wird.

Der Nutzer kann auf einer digitalen Zeichenfläche (ZeichenPanel) mit der Maus eine Zeichnung anfertigen (z. B. einen Apfel, eine Brille, eine Kerze). Nach dem Zeichnen kann per Knopfdruck die Skizze von einem **trainierten neuronalen Netz** klassifiziert werden. Die **fünf unterstützten Kategorien** sind:

- Apple
- Candle
- Eyeglasses
- Fork
- Star

Das System zeigt:

- Die geschätzte Wahrscheinlichkeit für jede Kategorie in einem Balkendiagramm
- Die am wahrscheinlichsten erkannte Kategorie als Text
- Eine Animation zur besseren Visualisierung

2. Teamrollen & Aufgabenverteilung

♦ *Dilshod Jabborov*

- Aufbau der Benutzeroberfläche mit **Swing**
- Implementierung des ZeichenPanel (Malfläche) und DiagramPanel (Balkendiagramm)
- Gestaltung der Main-GUI mit modernen Farben und Animationslogik
- Verbesserung der Interaktion über TEST und CLEAR Buttons

♦ *Mohammed Al-Shami*

- Implementierung der neuronalen Netzarchitektur (NeuralNetz, Trainer)
- Entwicklung des **Forward**- und **Backpropagation**-Verfahrens
- Aufbau und Durchführung des Trainingsprozesses mit echten QuickDraw-Daten
- Erweiterung der mathematischen Funktionen (MathFunctions) für das Netzwerk

◇ *Mohammed Elayan*

- Anbindung von GUI und Modell über GameController
- Mitentwicklung von Datenstrukturen (TrainingSample, MultiModelClassifier)
- Unterstützung bei Formatierung und Datenfluss zwischen GUI → NN → Rückgabe
- Dokumentation und Vorbereitung auf Tests und Präsentation

◇ *Taeb Al-Areqi*

- Aufbau und Design der Netzstruktur in Java
- Erstellen des Trainingsalgorithmus (inkl. Early Stopping, Weight Update)
- Pflege der Datenpfade und Speicherung/Einlesen von Gewichtsmatrizen
- Verbesserung der Backpropagation und Durchführung des Trainings

3. Startanleitung & Voraussetzungen

Systemvoraussetzungen

- **Java 17** oder höher
- Swing und Gson sind erforderlich.

So startet man das Programm:

1. Projekt mit einem Java-fähigen Editor öffnen
2. Startklasse: `javac -cp ".;model\Data\gson-2.8.9.jar" view*.java` danach
`java -cp ".;model\Data\gson-2.8.9.jar" view.Main` ausführen
3. Nach Start erscheint das Fenster mit Zeichenfläche
4. Mit Maus zeichnen → „Recognize“ drücken → Ergebnisse werden angezeigt
5. Mit „Clear“ das Bild löschen