**Real-time rock, paper, scissors recognition project and winning move suggestions**

**Team member:** Makhan Azat IT-2206

**Project idea:** The main goal of the project is to create a machine learning system that analyzes human gestures from an image in real time and, taking into account winning combinations, recommends a winning move to the user.

**Background information of the idea:** Rock-paper-scissors is not a transitional game, which is usually played by two people, in which each player simultaneously forms one of three figures with an outstretched hand. These are "stone" (clenched fist), "paper" (flat palm) and "scissors" (fist with extended index and middle fingers forming the letter V).  
A simultaneous zero-sum game, it has three possible outcomes: a draw, a win or a loss. A player who decides to play stone will beat another player who chooses scissors ("stone crushes scissors" or "breaks scissors", and sometimes "blunts scissors"), but will lose to the one who played paper ("paper covers stone"); the game of paper will lose to the game of scissors ("scissors they cut the paper"). If both players choose the same piece, the game ends in a draw and a substitution is usually made to break the tie.

The main mission of the project is to interact with the machine using a game (rock, scissors, paper) as a way of interaction. However, it is impossible for machines to understand human images or gestures, so we use Machine Learning and Deep Learning as an interaction tool to create a neural network model architecture that will reproduce human interactions.

**Solutions:** As a similar solution to this project, the tenso.rs website was found, which has a similar way of working. However, there the game is played against the computer, where any combination can occur, for example, a win, loss, or draw. Moreover, the demo version is based on the Tensor Force library with GPU acceleration for deep learning in the browser. It is fast enough to classify the video from the webcam on the client side in real-time.

Reference: <https://tenso.rs/demos/rock-paper-scissors/>  
  
Also, after another study, several sites were discovered that implemented the principle of this project on the same TensorFlow, and the information that is published on these sites will be useful for further development of the project and creating a model for the project.

Reference: <https://medium.com/@sdwiulfah/rock-paper-scissors-image-classification-with-keras-tensorflow-1c29ba0fe14d>

<https://medium.com/geekculture/rock-paper-scissors-image-classification-using-cnn-eefe4569b415>

**Used Dataset:** The official TensorFlow datasets were selected as the data set for the project. There is a dataset in the catalog called rock\_paper\_scissors, which is a convenient dataset because you can directly connect the dataset to the Google colab project. It also contains about 3 thousand photos that will give an accurate model after training. **Link:** <https://www.tensorflow.org/datasets/catalog/rock_paper_scissors?hl=en>

**My solution:** My solution involves a web application that analyzes the gesture in real time and then recommends a winning move. The implementation is carried out by analyzing data using machine learning. Creating a gesture recognition model using multiple layers and a basic model. The model will go through a dataset that is divided into two sets of train and test, which will eventually give an accurate model.

**Tech stack that will be used:**

* **Python:** the main language which will be used for data processing
* **TensorFlow and Keras:** a deep learning framework which will be used to build and train a neural network model.
* **TensorFlow Datasets:** will be used to load and process the "rock\_paper\_scissors" dataset.
* **TensorFlow.js :** JavaScript code that will be used to load a pre-trained model, make predictions based on captured frames, and display the results on a web page.
* **JavaScript:** handles the integration of the trained model into a web application**.**
* **HTML/CSS:** visualization and design.