|  |  |  |  |
| --- | --- | --- | --- |
| Assignment 1 | | Project Summary | |
| Course | | Practical Robotics and Smart Things - 2021 | |
| GitHub Repository | | -- URL to your repo here-- | |
|  | | | |
| Project author | | | |
| № | Name | | Faculty Number |
| 1 | Trayan Iliev | | 81255 |

|  |  |
| --- | --- |
| Project name | Paper printed keyboard for Raspberry Pi |

|  |
| --- |
| 1. Short project description (Business needs and system features) |
| A keyboard requires a great deal of resources and is restricted by its physical features. Additionally, discarded keyboards also inevitably contribute to environmental pollution. Consequently, the touch screen is designed to replace the physical keyboard and thus reduce these flaws. However, the internal digital keyboard on the touch screen takes up a substantial amount of space, which causes some content to be covered. Moreover, the touch screen can be dirtied by fingerprints and become worn over time by human fingernails through frequent use. Hence, it is necessary to develop a new type of environment-friendly virtual keyboard with fewer flaws. For that project virtual keyboard will be paper printed keyboard.  .  The hardware implementation is based on Raspberry Pi 3 single board computer, equipped with following sensors:   * *High resolution camera* - used to take pictures in order to determine which key is pressed * MPU-6050 Six-Axis (Gyro + Accelerometer) MEMS – used to measure rotation and   Movement of the finger in order to determine if key is pressed  Software algorithm consist of learning an algorithm for determine if key is pressed. Input data is generated by the accelerometer and gyroscope. For learning it will be used Keras and probably Recurrent Neural Network.  Second part of the algorithm will be recognizing which key is pressed by taking a photo from the raspberry camera and printing on the screen recognized key.  For the second phase raspberry pie will send pictures in json format via kafka and Apache spark will learn and predict by consume the pics from Kafka broker. For learning the algorithm  will be use "hand/pose\_iter\_102000.caffemodel" for feature extraction and data preprocessing and will be used various algorithms from Spark MLIB. |

|  |  |  |
| --- | --- | --- |
| 1. Main Use Cases / Scenarios | | |
| **Use case name** | **Brief Descriptions** | **Actors Involved** |
| * 1. **Type in a terminal** | User can using paper printed keyboard as a regular keyboard | All users |

|  |  |  |
| --- | --- | --- |
| 1. Main Views (Mobile Web App) | | |
| **View name** | **Brief Descriptions** | **URI** |

|  |  |  |
| --- | --- | --- |
| 1. API Resources (REST/SSE/WebSocket Backend) | | |
| **View name** | **Brief Descriptions** | **URI** |