Project Name (TBD)
Requirements Specification

authors here (TBD)

December 2, 2020

Contents

1 Product Overview

Project Name (TBD) is a web-tool enabling its users to create, edit and manage machine learning models on different sensor data. Project Name (TBD) consists of two primary components:

1. A server application that handles model training and storage, and makes these models available through a REST API. Supports multiple models and multiple users.

2. Desktop and mobile client applications to manage and interact with the model, e.g. adding new data points, identifying given data points and setting parameters for model training. A model can be utilized on different clients at the same time. Clients have different capabilites depending on the platform, mobile clients allow input of new data points in form of raw sensor data, while desktop clients allow management features.

Both components together allow for a streamlined and easy-to-use machine learning experience for both the unexperienced and the expert.

2 Purpose

2.1 Core criteria

- The web pages are the graphical user interfaces (GUI) for users.

- Users register a new account or login with their existing accounts to use the tool. The users are presented with their own set of workspaces after logging in or a prompt to create a new workspace if they do not have one.

- When creating a new workspace, the user chooses a name for their new workspace and the sensors that they wish to use when recording samples and classifying actions.

- The desktop client allows creating new labels and renaming the existing ones for the actions to be recorded in the workspace. It also shows the amount of recorded samples with the label.

- The desktop client presents a QR code specific to the selected workspace. The code connects the mobile client to the same workspace when scanned.

- The mobile web client allows recording of data samples in form of raw sensor data. The user chooses the label for which they wish to record. The recording parameters like sampling rate and countdown or recording duration can be configured.

- The mobile web client gives a real time feedback of sensor data to the user during the recording period.

- Recorded data samples are stored in a database and displayed chronologically on the desktop web client. The user can view the metadata of the sample by clicking on it, e.g. recording date and time or sample size in memory. The user can remove samples.

- The desktop web client presents the user options/features to process the data set in the workspace. The results are stored in turn and are available to the user.

- Trained and available models are listed on the desktop web client. The user can view the metadata of each model, e.g. parameters and data samples used for the training or performance metrics by clicking on the model.

- Each model is assigned a link which the user can use on a mobile device to classify action with the respective model. This identification happens in real time. The

user can view the link as well as the QR code with the same link embedded to ease usage.

2.2 Optional criteria

- The desktop web client serves a "Stay Signed In" functionality.

- Other data capturing devices are supported, e.g. Arduino.

- The desktop web client displays a status sign if a data collecting device is currently connected to the workspace, e.g. a green sign if connected and a red sign otherwise.

- The mobile web client can define triggers if something is detected, e.g. play a sound.

- Workspaces can be deleted together with the related data samples.

- Data samples can be transferred between compatible workspaces.

2.3 Exclusion criteria

- The mobile web client does not have a QR scanner.

- The user can not change the sensors used in a workspace.

- Each web page is designed for either a desktop device or a mobile device. Trying to display a web page in an unsupported device has undefined behavior.

3 Usage

3.1 Area of Application

The application is for collecting and labeling sensor data, training a model from the collected data and serving the stored resulting model to the user to classify actions in real time.

3.2 Target Groups

The target group of this application is people that wants to create a machine learning model from mobile sensor Data. That includes:

- People new to machine learning who wants an easy tool to discover machine learning and create models that they can use.

- **ADD PERSONA TBD**

- Experienced engineers that have knowledge of machine learning but need a fast tool to collect data and create a model.

We assume a rather technical audience.

3.3 Operating Conditions

The tool is a web application that can be run on any modern browser. A web browser must be installed on the user device to view the webpage. The application can be used from anywhere with a decent network connection.

Service Duration: 24 hours a day

4 Operating Environment

Database and the server appliance both run on a server, whereas different web browsers may be used for the clients. A Docker image may be provided for the server appliance.

4.1 Software

4.1.1 Server Appliance

- MongoDB support

- Recent versions of Node.js and Python

4.1.2 Client Devices

- A modern web browser (Chrome 87+, Firefox 83+) supporting:

  - Sensor APIs

  - Web Storage API

4.2 Hardware

4.2.1 Server Appliance

The server has to be fast enough to support all clients. This depends on the expected number of clients. Most computations will be done on the server. **TOO ARBITRARY?**

4.2.2 Client Devices

- Video Device

5 Functional Requirements

5.1 Main Functions

These functions must be implemented in order to fulfill the core criteria.

5.1.1 Web Client

The web client supports both desktop and mobile modes. The functionality that will be displayed is determined by the device information of the browser.

Desktop Web Client

/**F010**/ Show a welcome page with the registration and login panel

/**F020**/ List workspaces of the logged in user

/**F030**/ Allow creating workspaces given the name and sensors to be used

/**F040**/ List the available sensors for the recordings

/**F050**/ Allow renaming workspaces

/**F060**/ Show a workspace panel when a workspace is selected

/**F070**/ Allow creating labels for the actions to be recorded on the workspace

/**F080**/ List labels with their sample count

/**F090**/ Allow renaming labels

/**F100**/ Allow deleting labels which in turn deletes the data samples with the selected label

/**F110**/ Display a link and the QR code with the same link embedded to be used for recording data

/**F120**/ Display the collected data samples chronologically on the workspace panel

/**F130**/ Display the date and time of the recording and data size of the selected sample

/**F140**/ Allow deleting samples

/**F150**/ List the trained models

/**F160**/ Display the used parameters and data samples of the selected model

/**F170**/ Display the performance metrics of the selected model

/**F180**/ Allow selecting the possible options on the data on the management panel

/**F190**/ Request the processing of the data according to the selected options

Mobile Web Client

/**F200**/ Show a configuration page with available labels

/**F210**/ Allow configuring the countdown duration until the recording starts

/**F220**/ Allow configuring the recording duration

/**F230**/ Allow configuring the sampling rate of the sensors

/**F240**/ Show a button to initiate the recording

/**F250**/ Show a countdown page

/**F260**/ Display the current configuration on the countdown page

/**F270**/ Show a recording page

/**F280**/ Display the sensor data in real-time as curve graphs

/**F290**/ Send the sensor data to the server

/**F300**/ Show a recording completed page

/**F310**/ Allow another recording with the same configurations

/**F320**/ Allow editing configurations for the next recording

5.1.2 Server API

/**F330**/ Serve authentication services

/**F340**/ Serve workspace information for a user

/**F350**/ Serve data set and recording link for a workspace

/**F360**/ Create, rename and delete labels on a workspace

/**F370**/ Serve label information to the mobile client

/**F380**/ Accept data from the mobile client

/**F390**/ Initiate the configured model training

/**F400**/ Serve the model information on a workspace

/**F410**/ Rename and delete models on a workspace

5.1.3 Data Processing

/**F420**/ **TBD, after workshop**

5.2 Extending Functions

/**F430**/ Allow listing samples by label on the workspace panel

6 Data

TBD

6.1 System Data

TBD

6.2 User Data

TBD

6.3 Database

TBD (Data-Warehouse data ?)

7 Nonfunctional Requirements

7.1 Usability

- /N010/ User does not need to register to use the web-tool.

- /N020/ User should be able to record a new sample in less then 3 interactions.

- /N030/ User should be able to train and deploy the machine learning model (/FXXX/) with a single click.

7.2 Swiftness

- **lazy load components?**

- /N040/ Sth about page load speed - might be too restrictive, cannot assess yet.

7.3 Maintainability

- /N050/ Codebase is well documented to ease bug fixes.

7.4 Portability

- /N060/ Desktop web client /FXXX/ should be functioning on browsers newer than Chrome version XX, Firefox version XX, ...

- /N061/ Mobile web client /FXXX/ should be functioning on browsers newer than Chrome version XX, Firefox version XX. ...

7.5 Security

- /N070/ A workspace is inaccessible to other users without the QR code or the link to the workspace.

7.6 Scalability

- /N090/ A maximum of 10.000 ?? workspaces with a maximum of 10 ?? users at the same time is supported.

8 System Models

TBD

8.1 Scenarios

TBD

8.2 Use Cases

TBD

9 User Interface

TBD

9.1 Desktop

TBD

9.2 Mobile

TBD

10 ??Qualitatszielbestimmung??

11 Test Cases and Test Scenarios

TBD

11.1 Test Cases

11.1.1 Main Test Cases

These test cases are necessary for the web-tool to work properly and succeed in serving basic requests.

- /T010/ Access web page (/F010/)

    - /T011/ via Google Chrome,

    - /T012/ via Mozilla Firefox.

- /T020/ Show a QR code.

    - /T021/ Scan the QR code to create a new workspace.

    - /T022/ Scan the QR code to access an already existing workspace.

- /T030/ Access an already existing workspace with a link.

- /T050/

- /T040/ Select which sensors to use for recording.

- /T050/ Set the duration for the countdown.

- /T060/

11.1.2 Extending Test Cases

11.2 Test Scenarios

TBD

12 Development Environment

12.1 Operating Systems

- Windows 10 with WSL1 / WSL2

- Ubuntu 20.04.1 LTS

12.2 IDEs, Editors

- VS Code

12.3 Versioning

- git

- Github for hosting

12.4 Integration

- **(we'll need a CI server, TECO has one?) FIXME**

12.5 Miscallaneous

- LaTeX for documents

- 3D Paint for mockups

- VS Code Live Share for communication

- Discord for communication

And many more to be added later...