

```

\documentclass[parskip=full]{scrartcl}
\usepackage[utf8]{inputenc} % use utf8 file encoding for TeX sources
\usepackage[T1]{fontenc} % avoid garbled Unicode text in pdf
\usepackage[german]{babel} % german hyphenation, quotes, etc
\usepackage{hyperref} % detailed hyperlink/pdf configuration
\usepackage{graphicx} % provides commands for including figures
\usepackage{csquotes} % provides \enquote{} macro for "quotes"
\usepackage[nonumberlist]{glossaries} % provides glossary commands
\usepackage{enumitem}

\makenoidxglossaries

\newglossaryentry{nn}
{
  name={neural network},
  description={a network or a circuit of neuron used for information processing
inspired by the way biological neural systems process data},
  plural=neural networks
}

\newglossaryentry{img}
{
  name={image},
  description={a two dimensional matrix of red,green,glue (RGB) values that can be
visualized as each cell represents a single pixel on the monitor. (ex.: a
photo)},
  plural=images
}

\title{Specifications}
\author{}
\begin{document}

\maketitle

\section{Preface}

\section{Goal}
The goal is a software which performs image classification and is able to switch
between deploy platforms and working modes.
It also should have a GUI to control the software and to show the results.

\section{Product use}
Image classification

\section{Acceptance criteria}
\subsection{Must}
\begin{itemize}[nosep]
\item [MAC010]: Image classification
\item [MAC020]: Running NN on heterogenous platforms, CPU and FPGA
\item [MAC030]: Different operating modes
\item [MAC040]: GUI for interacting with software
\item [MAC050]: Performance/power prediction
\end{itemize}

\subsection{Can}
\begin{itemize}[nosep]
\item [KAC060]: Training a nn for classification
\item [KAC070]: Illustration of a topology of the nn
\item [KAC080]: Object detection
\item [KAC090]: Choosing between different models
\item [KAC100]: Creating new models

```

```

\item [KAC110]: Voting of multiple nn
\item [KAC120]: Using video for classification
\item [KAC130]: Using camera for classification input
\item [KAC140]: Running NN on GPU
\end{itemize}

```

\section{Functional Requirements Must}

```
\begin{itemize}[nosep]
```

```

\item [MFR025]: Dispatching the calculation process defined from the mode
\item [MFR030]: Support CPU for calculation
\item [MFR031]: Support FPGA for calculation
\item [MFR040]: Communication between Host-PC and platform
\item [MFR041]: Send image for classification
\item [MFR042]: Receive result
\item [MFR050]: GUI
\item [MFR060]: Showing results
\end{itemize}

```

```
\begin{tabular}{p{2cm}p{12cm}}
```

```

\textbf{MFR010} & \textbf{Use \gls{nn} for \gls{img} classification}\\
& A \gls{nn} should be used in order to classify \glspl{img} based on what is shown
on them. For each \gls{img} a list of possible classes it could belong to along with
degree of confidence should be given as output.\\
& \\
\textbf{MFR011} & \textbf{Deploy pre-trained \gls{nn} with the corresponding layers}\\
& \\
& A pre-trained \gls{nn} should be deployed to with all the corresponding layers in
order to fulfill MFR010.\\
& \\
\textbf{MFR020} & \textbf{Have high performance operating mode}\\
& An option to perform calculations fast with low regard for power consumption.\\
\textbf{MFR021} & \textbf{Have low power consumption operating mode}\\
& An option to perform calculations with low power consumption and low regard for
speed.\\
& \\
\textbf{MFR022} & \textbf{Have high energy efficiency operating mode}\\
& An option to perform calculations at an adequate balance between speed and power
consumption.\\
& \\
\textbf{MFR023} & \textbf{Calculator for power consumption}\\
& Calculations for the possible power consumption running the \gls{img}
classifications would result in based on the \gls{nn}, platform and operating mode
used.\\
& \\
\textbf{MFR024} & \textbf{Calculator for performance}\\
& Calculations for the possible performance running the \gls{img} classifications
would result in based on the \gls{nn}, platform and operating mode used.\\
& \\
\textbf{FR070} & \textbf{Choosing image for classification}\\
& Testet with: Implements: \\
& The GUI has a button with an on click event which opens a file explorer. The
explorer filters the files so that only files of the format .jpg, .png, .bmp are
listed. That also are the only valid formats.\\
\textbf{FR080} & \textbf{Choosing platform/hardware}\\
& Testet with: Implements: \\
& The GUI has a dropdown which lists the devices on which the classification can be
done. The devices which can be theoretically be accessed but aren't connected to the
host pc or the communication with them doesn't work are grayed out. \\
\textbf{FR090} & \textbf{Choosing mode}\\
& Testet with: Implements: \\
& The GUI has dropdown which lists the modes (high performance mode, low power
consumption mode and best energy efficiency mode). The power consumption in Watts and

```

& The GUI has dropdown which lists the modes (high performance mode, low power consumption mode and best energy efficiency mode). The power consumption in Watts and performance in FLOPs are also stated behind the mode names.

\end{tabular}

\section{Functional Requirements Can}

\begin{tabular}{p{2cm}p{12cm}}

\textbf{CFR012} & \textbf{Reading and parsing \gls{nn} configuration/weight file}\\& Having the ability to import/export different (external) \glspl{nn} to use for the \gls{img} classification.\\

& \\

\textbf{FR100} & \textbf{Choosing between different models}\\

& Testet with: Implements: \\

& The GUI has a button which opens the file explorer which filters for .txt files, there you choose the config file of the neural network with which you want to use. The program loads this config and parses it so it can be deployed. Possible models are GoogLeNet or AlexNet.\\

\textbf{FR110} & \textbf{Train nn for classification of imageset (with transfer learning)}\\

& Testet with: Implements: \\

& The user chooses a pretrained neural network and a new imageset and then can train the neural network on this new imageset with transfer learning.\\

%\newpage

\end{tabular}

\begin{itemize}[nosep]

%\item [KFR111]: Saving new trained nn (config an weights)

%\item [KFR112]: Choosing/Reading data set

\item [KFR032]: Support GPU for calculation

\item [KFR113]: Backpropagation

\item [KFR114]: Choosing parameters like learning rate

\item [KFR120]: Illustrating nn topology

\item [KFR130]: Object detection algorithm

\item [KFR131]: Showing detected object

\item [KFR132]: Choosing between detection and classification mode

%\item [KFR140]: Creating new topology

%\item [KFR150]: Choosing between training and inference mode

%\item [KFR160]: Choosing video in format .avi

%\item [KFR161]: Apply classification for a certain amount of frames

%\item [KFR170]: Connect with camera

%\item [KFR171]: Receive video stream from camera

%\item [KFR180]: Detecting object

%\item [KFR181]: Drawing bounding box

\end{itemize}

\section{Productdata}

\begin{tabular}{p{2cm}p{12cm}}

\textbf{PD010} & \textbf{Images for classification}\\

& The user can choose images of the format .jpg, .png, .bmp. The images are chosen by the user with the file explorer.\\

\textbf{PD020} & \textbf{Config/weight file of pretrained model}\\

& It is a .cfg file. In the beginning are hyperparameters described with the format \textit{name = value}. Then the layers are described in their order with the following format\\

& \textit{\lbrack kind of layer\rbrack}\\

& list of parameters in the format \textit{name = value}\\

\textbf{PD030} & \textbf{Labeled image set for classification training}\\

& The dataset is chosen by the user. The dataset is a directory with images and the

& The dataset is chosen by the user. The dataset is a directory with images and the name of the image is the label.\\
 $\textbf{\textit{PD040}}$ & $\textbf{\textit{Labeled set of images for object detection training}}$ \\
 & It is a .txt file and a directory with images. The images are labeled with their name. The bounding box for each image are described in the .txt file, in the format $\textbf{\textit{imagename, x,y,width,height}}$. (X,Y) are the coordinates in pixel of the left bottom corner, the width and height are in pixel.
 $\textbf{\textit{end{tabular}}}$

$\textbf{\textit{section{Demarcation}}}$
 $\textbf{\textit{begin{itemize}[nosep]}}$
 $\textbf{\textit{item}}$ [D010]: No real time / no performance optimization
 $\textbf{\textit{item}}$ [D020]: No mobile support
 $\textbf{\textit{item}}$ [D030]: No neural network size optimization
 $\textbf{\textit{item}}$ [D040]: No low-level (Assembler) optimization
 $\textbf{\textit{end{itemize}}}$

$\textbf{\textit{section{Non-functional requirements}}}$
 $\textbf{\textit{begin{itemize}[nosep]}}$
 $\textbf{\textit{item}}$ [NF10]
 $\textbf{\textit{end{itemize}}}$

$\textbf{\textit{section{Test cases}}}$
 $\textbf{\textit{begin{tabular}}}\{p\{2cm\}p\{12cm\}\}$
 $\textbf{\textit{textbf{T010}}}$ & $\textbf{\textit{Use \textit{gls}\{nn\} for \textit{gls}\{img\} classification}}$ \\
 & $\textbf{\textit{State:}}$ A $\textbf{\textit{gls}\{img\}}$ is given as an input.\\
 & $\textbf{\textit{Action:}}$ Calculations are performed on hand of the $\textbf{\textit{gls}\{img\}}$ and a $\textbf{\textit{gls}\{nn\}}$.\\
 & $\textbf{\textit{Reaction:}}$ A list of possible classes the given $\textbf{\textit{gls}\{img\}}$ could belong to along with degree of confidence for each class are given as output.\\
 & \\\

$\textbf{\textit{textbf{T011}}}$ & $\textbf{\textit{Deploy pre-trained \textit{gls}\{nn\} with the corresponding layers}}$ \\
 & $\textbf{\textit{State:}}$ There is a $\textbf{\textit{gls}\{nn\}}$ (already trained).\\
 & $\textbf{\textit{Action:}}$ Calculations are performed based on a given $\textbf{\textit{gls}\{img\}}$ and the given $\textbf{\textit{gls}\{nn\}}$.\\
 & $\textbf{\textit{Reaction:}}$ A list of possible classes the given $\textbf{\textit{gls}\{img\}}$ could belong to along with degree of confidence for each class are given as output.\\
 & \\\

$\textbf{\textit{textbf{T012}}}$ & $\textbf{\textit{Reading and parsing \textit{gls}\{nn\} configuration/weight file}}$ \\
 $\textbf{\textit{T012.1}}$ & $\textbf{\textit{State:}}$ The user is on the page to select a $\textbf{\textit{gls}\{nn\}}$ to use for the $\textbf{\textit{gls}\{img\}}$ classification.\\
 & $\textbf{\textit{Action:}}$ The user selects the option to import a $\textbf{\textit{gls}\{nn\}}$.\\
 & $\textbf{\textit{Reaction:}}$ The file explorer opens.\\
 $\textbf{\textit{T012.2}}$ & $\textbf{\textit{State:}}$ The file explorer is open\\
 & $\textbf{\textit{Action:}}$ The user selects an $\textbf{\textit{gls}\{nn\}}$ to import\\
 & $\textbf{\textit{Reaction:}}$ The file explorer closes and $\textbf{\textit{gls}\{nn\}}$ is imported and selected for the classification calculations.\\
 & \\\

$\textbf{\textit{textbf{T020}}}$ & $\textbf{\textit{Have high performance operating mode}}$ \\
 & $\textbf{\textit{State:}}$ The user is ready to start the calculations.\\
 & $\textbf{\textit{Action:}}$ The user chooses to perform the calculations in high performance operating mode.\\
 & $\textbf{\textit{Reaction:}}$ The calculations run considerably faster than in the other possible modes with the same conditions.\\
 & \\\

$\textbf{\textit{textbf{T021}}}$ & $\textbf{\textit{Have low power consumption operating mode}}$ \\
 & $\textbf{\textit{State:}}$ The user is ready to start the calculations.\\
 & $\textbf{\textit{Action:}}$ The user chooses to perform the calculations in low power

```

& \textbf{Action:} The user chooses to perform the calculations in low power
consumption operating mode.\\
& \textbf{Reaction:} The calculations run with considerably lower power consumption
than with the other possible modes in the same conditions.\\
& \\

\textbf{T022} & \textbf{Have high energy efficiency operating mode}\\
& \textbf{State:} The user is ready to start the calculations.\\
& \textbf{Action:} The user chooses to perform the calculations in high energy
efficiency operating mode.\\
& \textbf{Reaction:} The calculations run with regard to balance between power
consumption and speed.\\
& \\

\textbf{T070} & \textbf{Choosing image for classification}\\
T070.1 & \textbf{State:} The user is on the page for image classification. \\
& \textbf{Action:} The user clicks on the button \glqq Choose image\grqq .\\
& \textbf{Reaction:} The file explorer opens with the filter for .png, .jpg, .bmp\\
T070.2 & \textbf{State:} The file explorer is open\\
& \textbf{Action:} The user selects an image with a valid format\\
& \textbf{Reaction:} The file explorer closes and image is as preview shown\\
%Manu quick fix:
\end{tabular}
\begin{tabular}{p{2cm}p{12cm}}
\newpage
\textbf{T080} & \textbf{Choosing platform/hardware}\\
T080.1 & \textbf{State:} The user is on the page for image classification\\
& \textbf{Action:} The user chooses with the dropdown the desired platform\\
& \textbf{Reaction:} An internal flag is set to the desired platform and the dropdown
shows the chosen platform.\\
& \\
\textbf{T090} & \textbf{Choosing mode}\\
T090.1 & \textbf{State:} The user is on the page for image classification\\
& \textbf{Action:} The user chooses with the dropdown the desired mode\\
& \textbf{Reaction:} An internal flag is set to the desired mode and the dropdown
shows the chosen mode\\
& \\
\textbf{T100} & \textbf{Choosing between different models}\\
T100.1 & \textbf{State:} The user is on the page for image classification\\
& \textbf{Action:} The user clicks on the button \glqq Choose neural network\grqq\\
& \textbf{Reaction:} The file explorer opens\\
T100.2 & \textbf{State:} The file explorer is open\\
& \textbf{Action:} The user selects an config/weight file\\
& \textbf{Reaction:} The file explorer closes and the software loads the input and
parses it. If it is loaded there is success message\\
& \\
\textbf{T110} & \textbf{Train neural network for classification of imageset}\\
T110.1 & \textbf{State:} The user is on the page for training, has selected a neural
network, a dataset for training, the kind of training, the learning rate and the
desired precision.\\
& \textbf{Action:} The user clicks on the button \glqq Train\grqq\\
& \textbf{Reaction:} The software starts to train the selected network with the
selected configuration and shows the progress in line graph.\\
& \\
%Manuel
\textbf{T111} & \textbf{Saving a NN after training}\\
T111.1 & \textbf{State:} The user is on the page for training, has selected a neural
network, a dataset for training, the kind of training, the learning rate and the
desired precision.\\
& The training finishes.\\
& \textbf{Action:} No action required\\
& \textbf{Reaction:} The software stores the trained network weights in a predefined

```



```
& \textbf{Reaction:} The software stores the trained network weights in a predefined
format and with a usefull name in a fixed location.\\
& \\
\textbf{T112} & \textbf{Choosing/Reading data set}\\
T112.1 & \textbf{State:} A folder containing images is provided by the user.\\
& \textbf{Action:} No action required\\
& The software is able to read all images matching the allowed formats and allows
training and inferencing on these images.\\
```

```
& \\
\textbf{T140} & \textbf{Creating new topology}\\
T140.1 & \textbf{State:} \\
& \textbf{Action:} No action required\\
& \textbf{Reaction:} \\
& A folder containing images is provided by the user.\\
& The software is able to read all images matching the allowed formats and allows
training and inferencing on these images.\\
```

%Manu quick fix:

```
\end{tabular}
\begin{tabular}{p{2cm}p{12cm}}
& \\
\textbf{T150} & \textbf{Choosing between training and interference mode}\\
T150.1 & \textbf{State:} \\
& \textbf{Action:} No action required\\
& \textbf{Reaction:} \\
& A folder containing images is provided by the user.\\
& The software is able to read all images matching the allowed formats and allows
training and inferencing on these images.\\
```

```
& \\
\textbf{T160} & \textbf{Choosing video in format .avi}\\
T160.1 & \textbf{State:} The software is running. \\
& \textbf{Action:} The user selects a .avi video file.\\
& \textbf{Reaction:} The system stores the path to the selected video and is \\
& available to process single images from this video.\\
```

```
& \\
\textbf{T161} & \textbf{Apply classification for a certain amount of frames}\\
T161.1 & \textbf{State:} The software is running. A video source was choosen by the
user. All network details were provided by the user. Classification was choosen by
the user.\\
& \textbf{Action:} The user clicks on the button ``start classification''\\
& \textbf{Reaction:} The system processes the video file imagewise\\
```

```
& \\
\textbf{T170} & \textbf{Connect with camera}\\
T170.1 & \textbf{State:} The software is running\\
& \textbf{Action:} The user connects a usb camera to the pc\\
& \textbf{Reaction:} The system dynamically detects the camera and allows the user to
select the camera as an image source\\
```

```
& \\
\textbf{T171} & \textbf{Receive video stream from camera}\\
T171.1 & \textbf{State:} The software is running, a camera is connected to the host
pc.\\
& \textbf{Action:} The user chooses the camera as image source.\\
& \textbf{Reaction:} The first camera image is provided as a preview, the continuous
image stream is available for further processing.\\
```

```
& \\
\textbf{T180} & \textbf{Detecting object}\\
```

```
T180.1 & \textbf{State:} The software is running, a network including all parameters
and weights was provided. An image was provided. Detection mode was picked by the
user\\
& \textbf{Action:} The user clicks on the button ``start detection''\\
& \textbf{Reaction:} The network is run for inferencing and the network output is
shown to the user.\\

& \\
\textbf{T181} & \textbf{Drawing bounding box}\\
T181.1 & \textbf{State:} Inferencing was executed on an image given by the user, the
chosen neural network predicted bounding boxes.\\
& \textbf{Action:} No action required\\
& \textbf{Reaction:} The original image, given by the user, is overlayed with the
boxes predicted by the network, the updated image is presented to the user.\\

\end{tabular}
\newpage
\section{System models}

\subsection{Scenarios}
\subsection{Usecases}
\subsubsection{Seminarorganisation}

\printnoidxglossaries

\end{document}
```