1   System Requirements and Specifications

In this Chapter, the technical requirements for the system are described. These requirements provide the necessary standards to ensure the systems functionality. Additionally, the execution of the code is described.

1.1   Technical Requirements

Regarding the hardware requirements, the system should be capable of running Node.js, which is necessary for executing the frontend development tool Yarn. Additionally, the system should be able to execute the python backend and Melt framework within a Docker container. While implementing, it was observed that the system should have a minimum of 16 GB RAM to ensure smooth execution of the development environment and Docker containers.

The software requirements include information about the operating system, applications and frameworks. While testing, the code encountered difficulties running on Windows. While it should be possible that with debugging and configuration adjustments it could run, it was decided to proceed with this thesis project on a Linux system (Ubuntu 22.04). This decision was influenced by factors such as the compatibility of the code base with Linux and its command-line interface. Regarding the documentation in the README files, the Mac operating system should be able to run the code after making adjustments, but this was not tested in the scope of this thesis.

In addition to this, the system relies on several software components. Docker is needed to use Docker containers for encapsulating the system's backend. This allows consistent and efficient results across different devices. Node.js serves as the runtime environment for executing JavaScript code for web applications and Yarn is the package manager for Node.js applications, that is used in this project. Additionally, Git is essential for version control. While it is not strictly required for the system's execution, Git is used to pull Git repositories containing the code and necessary dependencies and frameworks, such as the Melt framework. An installation of the Melt framework is not needed for the execution or development of this system, but can be useful for testing or exploring the functionalities of Melt. Furthermore, the system adheres to standard security practices.

1.2 Execution of the Code

The codebase for this project consists of two directories – `accurate-player-3-core` and `jit-webrtc`. The system's architecture is described in Section **??**. Additionally, the READMEs of the two main components of the system can be seen in Appendix **??**.

The files for the frontend code are in a directory that is called `accurate-player-3-core`. To execute the frontend, it needs to be navigated to the directory `demo` in `packages` and then `yarn start` needs to be executed with the parameter for the backend in the CLI:

```
~/accurate-player-3-core/packages/demo$ JIT_BACKEND=http://localhost:8080 yarn
    start
```

For updating dependecies during development, the following command should be executed and running in addition:

```
~/accurate-player-3-core/packages/jit$ yarn start
```

To run the backend, the folder `jit-webrtc` needs to be accessed in the CLI to start the script `main.sh`, that starts the backend in a docker container and pulls the needed dependencies:

```
~/jit-webrtc$ docker/main/main.sh -v --threads 16 --port 8080  https://s3.eu-
    central-1.amazonaws.com/accurate-player-demo-assets/timecode/sintel-2048-
    timecode-stereo.mp4
```

The optional parameter `-v` (verbose) provides additional information or detailed output when executing the code. The threads and the port are set with the parameters `--threads` and `--port` and then the link to the test video file is added. This is the file that will then be played in the Accurate Player.

The test video file that is being used for this thesis project, is the animated short film *Sintel*, which was during its development known as *The Durian Open Movie Project*. It was created by the Blender Foundation using Blender, which is an open-source 3D animation software and released in 2010. It tells the story of a young girl, who is searching for her lost dragon companion.

The film is allowed to be used for free distribution, sharing, and adaptation, as long as the following credit is given to the original creators. This licensing approach promotes collaborations and progress of the Free and Open-Source Software (FOSS) community.

©copyright Blender Foundation | durian.blender.org

It is noted that for the sake of clarity, this copyright attribution is applied to the entire use of frames of the film in this thesis. It will not be added to each extracted frame.

The *Sintel* video file that is being used in this thesis project is retrieved from the following source:

https://s3.eu-central-1.amazonaws.com/accurate-player-demo-assets/timecode/sintel-2048-timecode-stereo.mp4