User Manual

Tim Visée & Nathan Bakhuijzen

October 2018

1 Goal

The purpose of this manual is twofold:

- To inform research how to use the platform to execute experiments and conduct research
- To inform future developers how to continue developing the platform

2 Problem Definition

What physical motions are natural, effortless and easy, in order to control a computer or other digital device?

Questions of this nature can be answered using the *Can't Touch This* platform. *Can't Touch This* aims to be a platform for researchers that would like to conduct research in the field of touchless computer systems. We believe that our platform allows researchers to build a strong foundation for the future of touchless control. Giving researchers the opportunity to conduct research improves the chance for touchless control of computers only seen in futuristic movies and tv shows.

2.1 Motivation

At the start of the KB-80 minor, students were given a choice in the subject of the research. Mister Hani introduced us to a series of subjects, of which the LeapMotion project was the most interesting to us. The idea of the LeapMotion was to create or extend existing software to enable people to control a computer without touching any peripherals, like keyboards and mice.

2.2 Background information

Research in the field of touchless computer systems is motivated by the desire for these systems in sterile environments. For example, surgeons often make use of computer systems to aid them during their surgeries by providing crucial information such as CT, MRI and X-ray scans. This is where touchless computer systems come in. These systems allows surgeons on control a computer without the need for physical peripherals.

3 Installation Guide

3.1 Requirements

- A computer with the Windows (7+), OSX (10.7+, Lion+) or Linux (kernel 2.6.18+) operating system
- An installation of the LeapMotion SDK
- An installation of the Rust programming language
- The physical LeapMotion device itself

3.2 Software Dependencies

The *Can't Touch This* platform is written using the Rust programming language. This means that the operating system that the platform will run on must support the Rust language. Fortunately, Rust runs on all popular operating systems today, shown above in the list of requirements. An up-to-date list of all supported versions can be found on the Rust website. Additonally, the *Can't Touch This* platform requires the LeapMotion SDK to provide all necessary sensor data. Just like the Rust programming language, the LeapMotion SDK can be installed on all platforms.
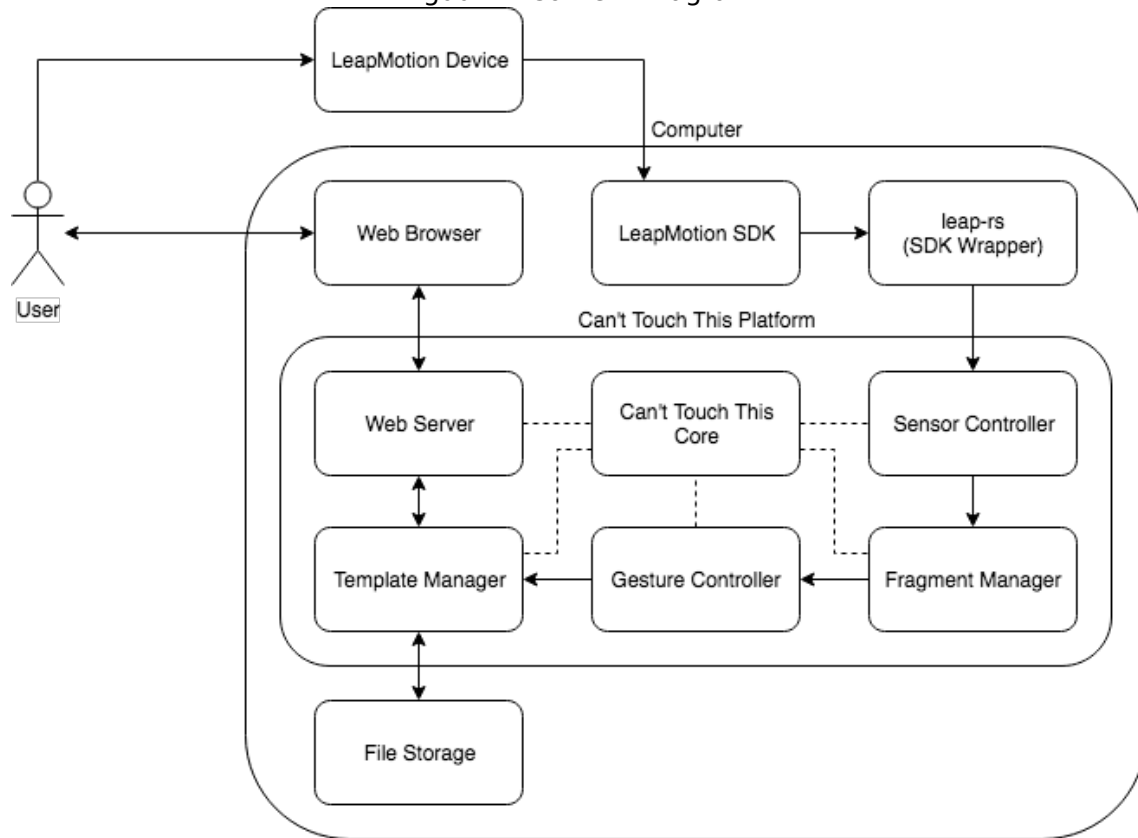
3.3 External resources
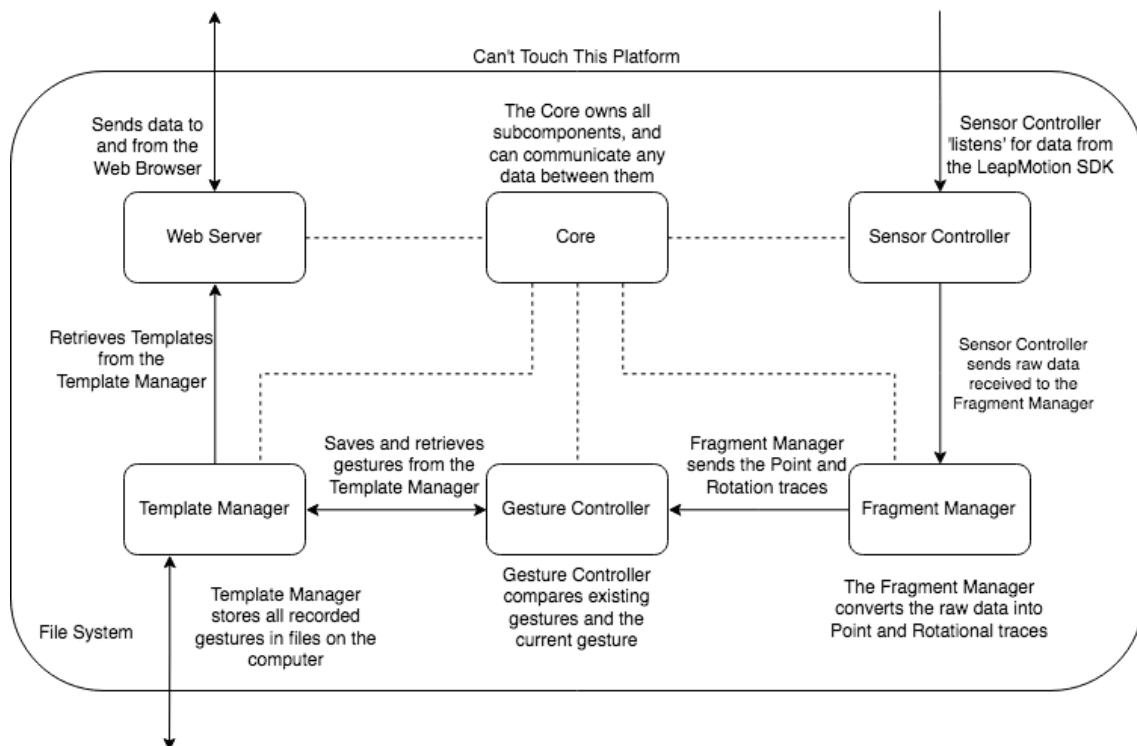
To be written.

3.4 External development tools

To be written.

4   Architecture Diagrams

Figuur 1: Context Diagram

Figuur 2: Context Diagram

Can't Touch This Platform

Sends data to and from the Web Browser

The Core owns all subcomponents, and can communicate any data between them

Sensor Controller 'listens' for data from the LeapMotion SDK

Web Server --- **Core** --- **Sensor Controller**

Retrieves Templates from the Template Manager

Sensor Controller sends raw data received to the Fragment Manager

Saves and retrieves gestures from the Template Manager

Fragment Manager sends the Point and Rotation traces

Template Manager ⟷ **Gesture Controller** ⟷ **Fragment Manager**

Template Manager stores all recorded gestures in files on the computer

Gesture Controller compares existing gestures and the current gesture

The Fragment Manager converts the raw data into Point and Rotational traces

File System

5 Known Issues

- *Can't Touch This* may crash upon running the release version of the exectable
- On macOS, the LeapMotion device may never give data to begin with
- On macOS, the LeapMotion device may stop recording data randomly
- On macOS, the application may not run well when minimalizing the backend application