Software Architecture Document (SAD)

PC parts web shop

***Made by Hristo Hristov***

**Fontys University of Applied Sciences**

**Eindhoven – Netherland**

**2020-2021**

Table of Contents

[1. Introduction 3](#_Toc84626375)

[1.1 Purpose of this document 3](#_Toc84626376)

[2. System Context (C1) 3](#_Toc84626377)

[2.1 Context 3](#_Toc84626378)

[3. Containers (C2) 4](#_Toc84626379)

[3.1 Context 4](#_Toc84626380)

[4. Components(C3) 5](#_Toc84626381)

[4.1 Context 6](#_Toc84626382)

[5.Code(C4) 6](#_Toc84626383)

[5.1 Context 7](#_Toc84626384)

Version Table

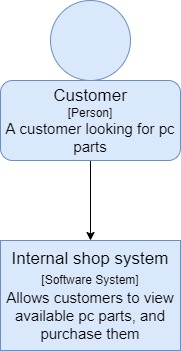
|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Changes** |
| 0.0 | Friday, October 8, 2021 | Work in progress |

# Introduction

## Purpose of this document

This document provides an architectural overview of the system, using the C4 model. It is intended to convey the architectural decisions which have been made on the system.

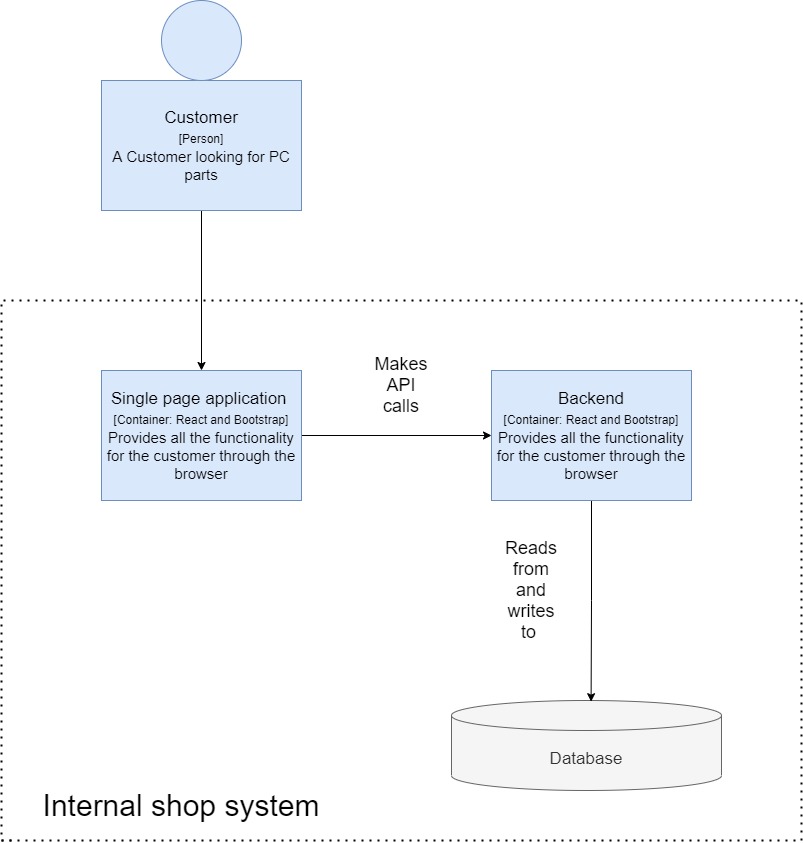
# System (C1)



## 2.1 Context

The first level represents the big picture of the software system. There is a user, and he interacts with the shop system. This diagram could be shown to non-technical people.

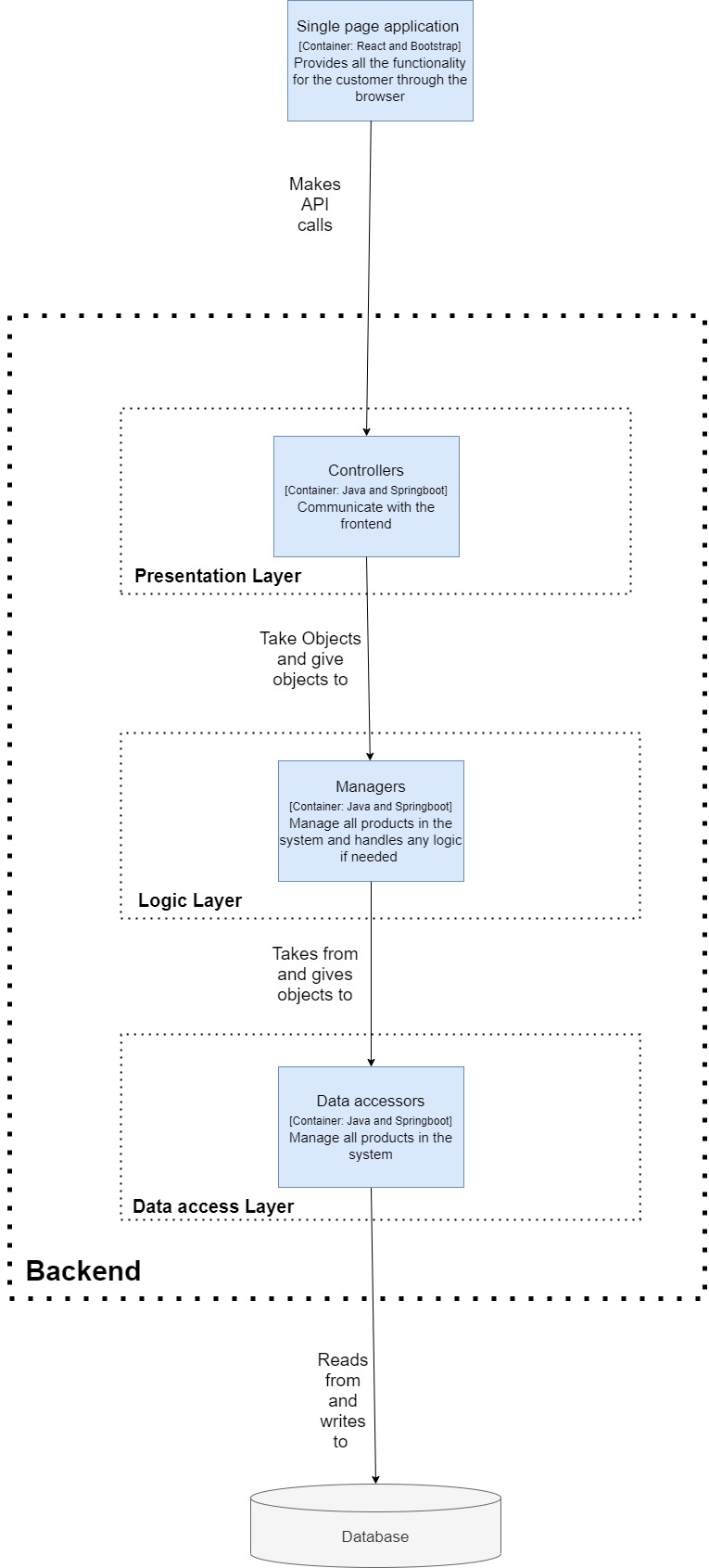
# Containers (C2)



## 3.1 Context

In the second level we zoom into the software system to see what containers it is has. Each container is a separately runnable. This shop system is made out of a frontend, backend, an API that connects the two and a database to store all information.

# Components(C3)



## 4.1 Context

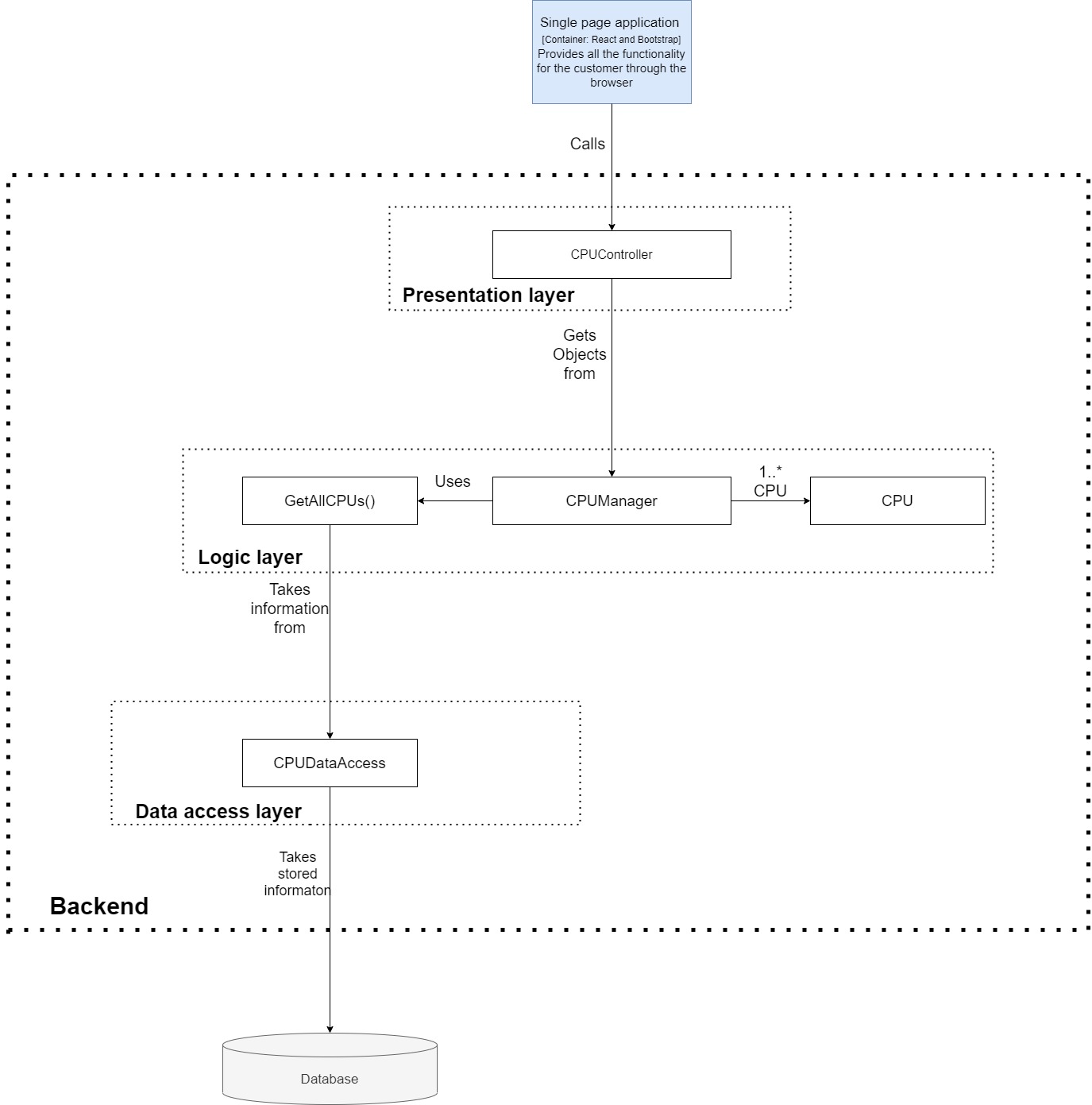
In the third level we zoom into one of the containers. In this case we take a look at the Backend and what components it is made out of.

The presentation layer deals with receiving API calls from the frontend and then in turn sending responses back.

The logic layer deals with taking information for the data access layer, manipulating it if needed and passing it to the frontend.

The data access layer deals with storing or taking information from the database.

# 5.Code(C4)



## 5.1 Context

In the fourth level we see in code what actual classes are used when the frontend makes a call to get all the CPUs from the backend. Making a request call for getting other parts (getting all motherboards for example) functions in the same way but makes use of different classes (MotherboardController instead of CPUController).