

Санкт-Петербургский государственный университет

Программная инженерия

Дулетов Дмитрий Евгеньевич

Разработка архитектуры программно-аппаратной платформы

Отчёт о прохождении производственной практики

Научный руководитель:
д.ф.-м.н., профессор кафедры СП О.Н. Граничин

Рецензент:

Санкт-Петербург
2023

Оглавление

1. Введение	3
2. Постановка задачи	4
3. Обзор	5
4. Реализация	6
4.1. Архитектура	6
4.2. Внешний сервер	11
5. Заключение	12
Список литературы	13

1. Введение

В современном мире роботизация набирает обороты и роботы на дистанционном управлении уже не кажутся редкостью. Однако их широкому распространению как в жизни, так и в образовательных процессах сильно мешает сложность конструирования и программирования роботов. До сих пор в школах стандартом преподавания робототехники является конструирование из Lego [1].

Для образовательных учреждений и частных исследований доступность аренды роботов или возможности конструирования на заказ очень ограничены, с одной стороны сложностью процесса настройки, с другой стоимостью услуг. Более того, возможности в удалённых населённых пунктах сводятся практически к минимуму.

Возможности моделирования роботов на компьютере за последнее время сильно продвинулись, существует несколько программ для данных целей, таких как Gazebo или WeBots. Однако, данные программы ограничены исключительно виртуальными представлениями и не дают полного понимания, как поведёт себя робот в той или иной реальной ситуации.

Цель этой и нескольких смежных работ разработать программно-аппаратную платформу, которая позволит клиентам получить доступ к сборке робота из любой точки мира, а затем протестировать его на реальном полигоне, имея при этом возможность отслеживать все изменения через подробную симуляцию.

2. Постановка задачи

Целью работы является разработка архитектуры программно-аппаратной платформы и настройка внешнего и внутреннего серверов.

Для достижения поставленной цели были поставлены следующие задачи:

1. Проведение обзора научно-исследовательской литературы в рамках разработки программно-аппаратных платформ.
2. Разработка архитектуры платформы.
3. Создание и настройка внешнего и внутреннего сервера.
4. Создание базы данных, содержащей драйвера и программы.

3. Обзор

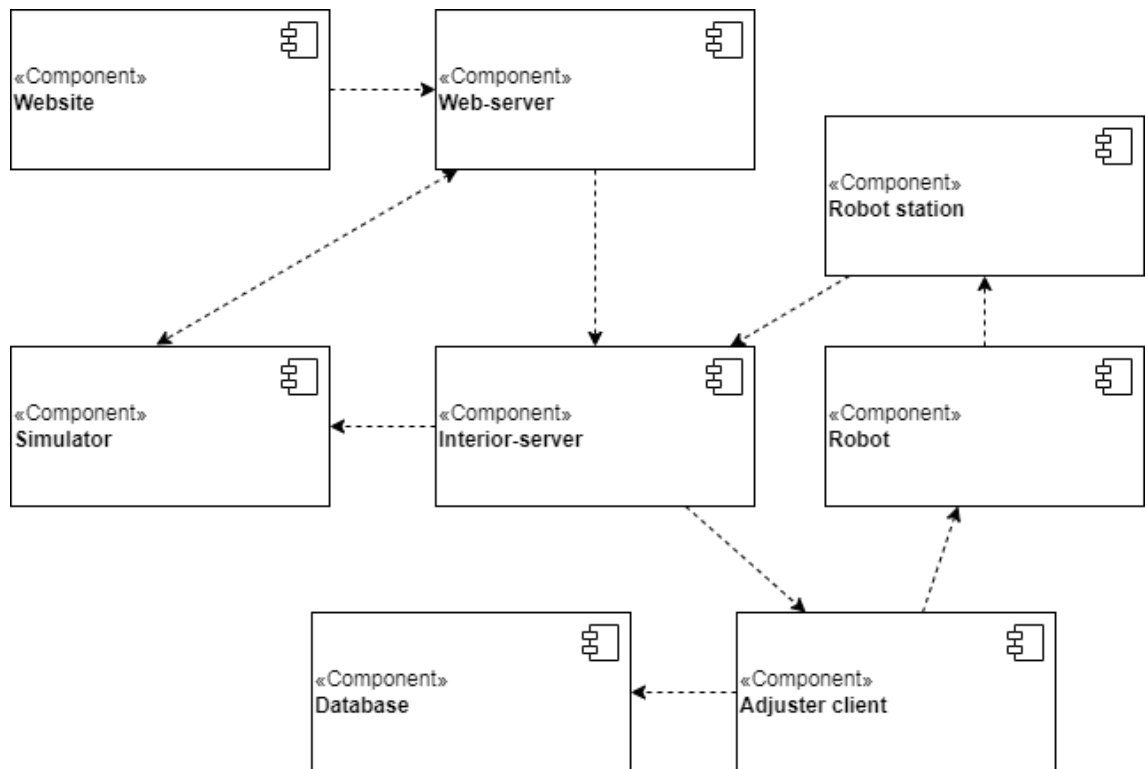
Разнообразие технических средств и операционных систем вынудили разработчиков систем ввести понятие платформы. Платформа определяет тип оборудования и программного обеспечения, на которых можно установить покупаемую информационную технологию. Она имеет сложную структуру. Главным компонентом платформы является тип компьютера, определяемый типом процессора. Следующим компонентом является операционная система, работающая на том или ином процессоре. Многие современные информационные технологии используют добавочное оборудование и специальные программные средства для его обслуживания. Например, сетевые информационные технологии зависят от сетевого оборудования: модемов, адаптеров, каналов связи и т.д. и средств, их обслуживающих. Новейшие информационные технологии представляют собой продукт интеграции различных информационных технологий. Поэтому платформа зависит от всех структурных частей: типа процессора и работающей на нем операционной системы, типа дополнительного оборудования и поддерживающих это оборудование программных средств.

В традиционном понимании платформа—это комплекс аппаратных и программных средств, на котором функционирует программное обеспечение пользователя ЭВМ. Программно-аппаратная платформа - комплекс, состоящий из аппаратуры(чипсеты,процессоры), а также конфигурации запускаемых на них программ [2]. На данный момент не существует не только программно-аппаратной платформы по запуску роботов, но и просто доступного онлайн симулятора по запуску роботов. Так один из самых популярных симуляторов роботов с открытым кодом - Gazebo имеет поддержку развёртывания на удалённом сервере при помощи ТСП

IP, однако доступных открытых серверов по использованию этой платформы онлайн не существует, а использование её на стационарном компьютере средней конфигурации осложнено системными требованиями [3].

4. Реализация

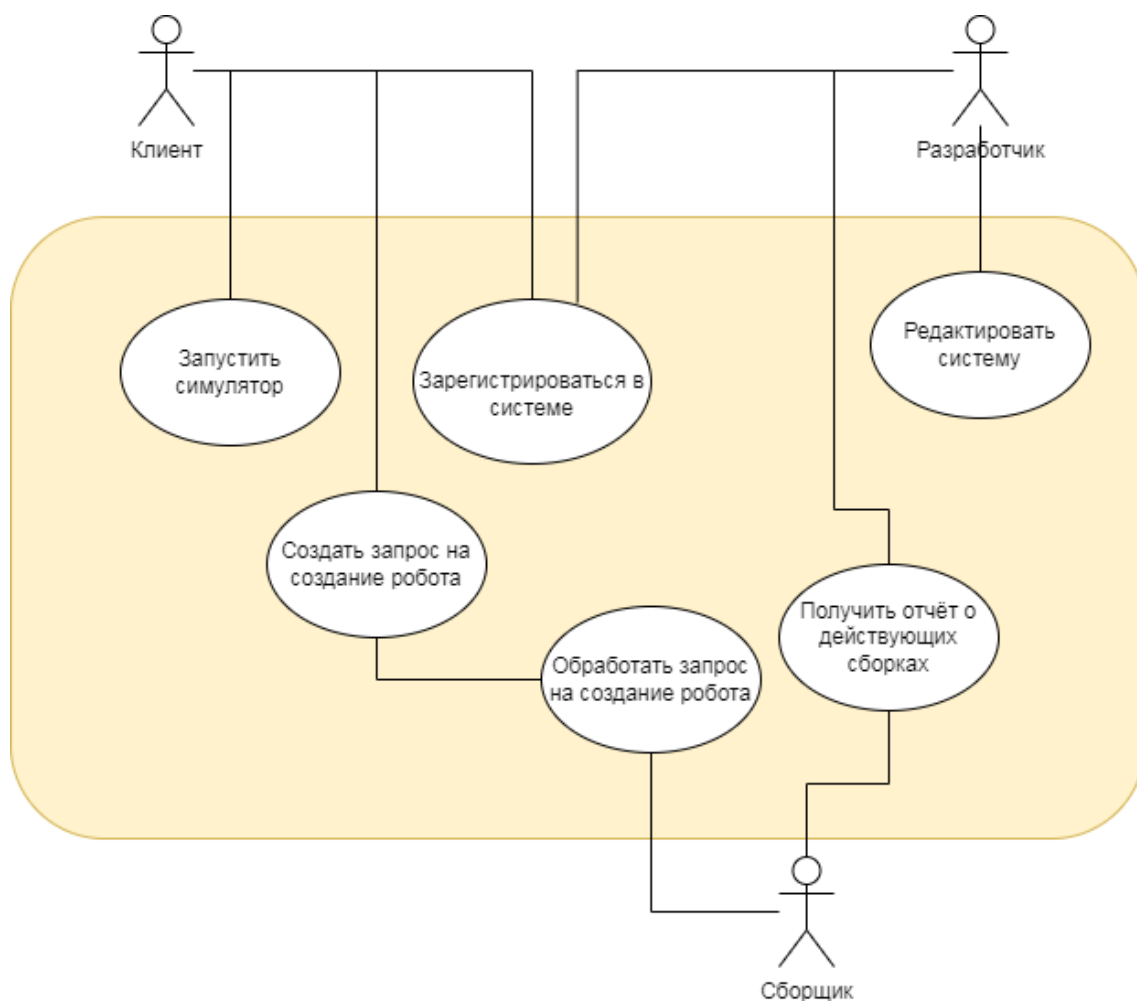
4.1. Архитектура



В связи с требованиями было решено конструировать приложение в виде веб-сервиса. Запросы пользователя, совершённые на сайте, обрабатываются внешним сервером, который пересылает запрос к внутреннему серверу, который в свою очередь выполняет запрос, возвращая результат работы через внешний сервер обратно пользователю.

Пользователь должен иметь следующие возможности, реализованные при помощи сайта:

1. Зарегистрироваться на сайте.
2. Узнать о возможностях системы.
3. Воспользоваться симулятором.
4. Создать запрос на сборку робота.
5. Протестировать робота.



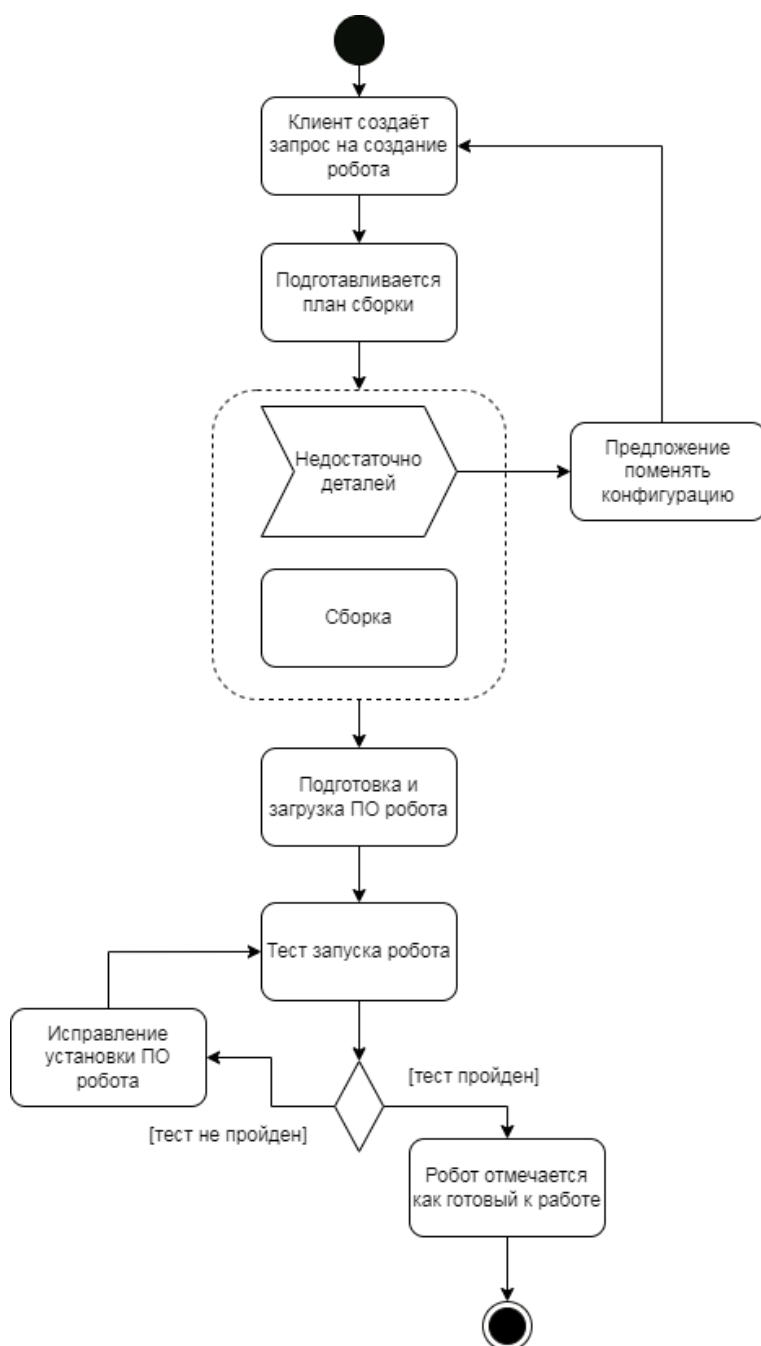
Сайт располагается на внешнем сервере, имеющем статический ip-адрес. Домен сервера пока не установлен. Запросы на регистрацию и ознакомления с возможностями системы выполняются непосредственно на внешнем сервере.

Помимо внешнего сервера система предполагает наличие внутреннего сервера, на котором располагается база данных, запущен симулятор и отслеживается состояние всех уже собранных роботов.

На внутреннем сервере одновременно запущены Mission planner и программа визуализации перемещения робота. Mission planner отвечает за обработку физической модели робота и передачи этих данных по протоколу MAVink. Эта программа отслеживает заряд, скорость, ускорение, положение в пространстве и другие показатели робота. Она может как генерировать эти показатели как для виртуального робота, так и считывать телеметрию со станции управления роботами. В качестве программы для визуализации рассматриваются AirSim, Gazebo и

Webots. Эта программа принимает данные от Mission planner в формате протокола MAVink и по этим данным генерирует картинку перемещения, которая захватывается при помощи ScreenCapture и через внешний сервер транслируется пользователю. Также при помощи Mission planner осуществляется управление роботом.

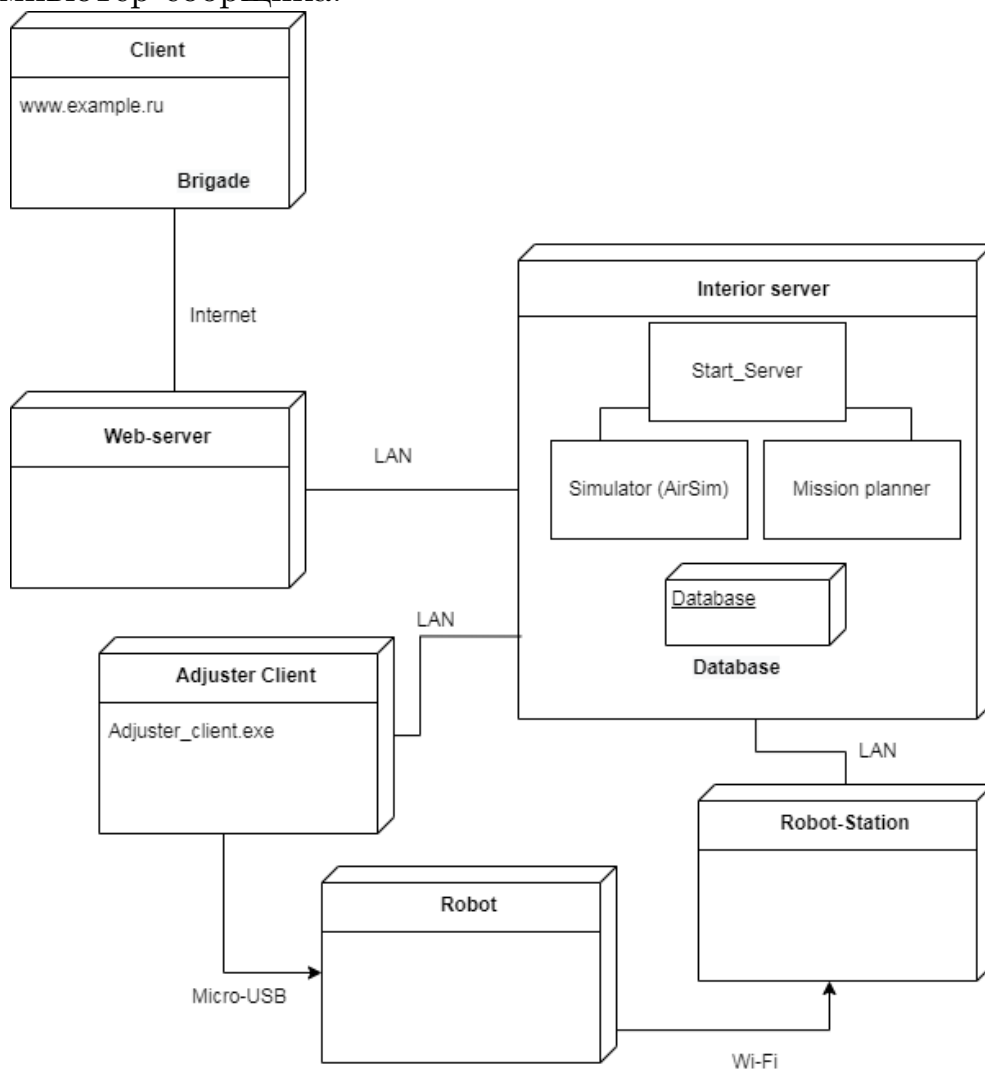
База данных внутреннего сервера хранит в себе драйвера и установочные пакеты для роботов. Когда в платформу поступает запрос на создание робота, из базы данных получаются все необходимые драйвера и программы и на клиенте сборщике компонуются в один пакет, который при развёртывании на плате робота автоматически устанавливает все необходимые драйвера, освобождая от этой работы самого сборщика.



Во время запуска робота, данные с него считывает станция управления роботами и передаёт его телеметрию в Mission planner, который формирует модель робота, для отладки или наглядной демонстрации.

Система предполагает физическое расположение двух или более серверов. Внешний сервер вмещает в себя веб-сервер для размещения сайта и прокси-сервер для защиты системы от внешнего воздействия. Внутренний сервер предполагает размещение симулятора, базы данных и программы мониторинга действующих роботов. Также на каждом полигоне возможна установка станции по отслеживанию роботов, которая

сообщает данные о них на внутренний сервер для отладки или визуализации. Программа клиента сборщика устанавливается непосредственно на компьютер сборщика.



4.2. Внешний сервер

Внешний сервер был развернут на следующем аппаратном обеспечении:

- Процессор Intel Core i3 10700
- 8 Гб ОЗУ
- Видеокарта Nvidia GeForce 940MX

В качестве операционной системы была взята Ubuntu 22.04.2 LTS как широко поддерживаемая система для серверов с множеством удобных утилит и широким выбором инструментов, будь то nginx или Apache.

5. Заключение

В рамках учебной практики была составлена архитектура программно-аппаратной платформы и установлены внешний и внутренний сервер. Дальнейшая разработка затруднена ожиданием других участников проекта.

Список литературы

- [1] Кузьмина, М. В. др. Робототехника в школе как ресурс подготовки инженерных кадров будущей России // сборник методических материалов для работников образования в условиях реализации Федеральных государственных образовательных стандартов (по итогам областных семинаров и курсов повышения квалификации по образовательной робототехнике): ИРО Кировской области, 2017. – 179 с.
- [2] Алексеевич Шашин Михаил. Аппаратно-программная платформа // Образовательный портал «Справочник». — URL: https://spravochnik.ru/bazy_dannyh/dopolnitelnye_voprosy_primeneniya_baz_dannyh/apparatno-programmnaya_platforma.
- [3] Yoonseok Pyo, Hancheol Cho, Leon Jung, Darby Lim: ROS Robot Programming (English): ROBOTIS, 2017. - 487 p.