Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

ОТЧЁТ

по производственной (технологической) практике

по специальности 1-40 02 01 «Вычислительные машины, системы и сети»

БГУИР 1–40 02 01 06 ПЗ

Студент: К.В. Горбачевский

Руководитель практики:

от предприятия:

ООО «Финмаркет Софт» И.А. Петрушенко

от кафедры ЭВМ:

УО БГУИР, кафедра ЭВМ Д.В. Куприянова

МИНСК 2024

Дата защиты:​​​​ \_\_\_\_\_\_\_\_\_\_\_\_\_

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_

Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет: ФКСиС. Кафедра: ЭВМ.

Специальность: 40 02 01 «Вычислительные машины, системы и сети».

Специализация: 40 02 01-01 «Проектирование и применение локальных компьютерных сетей».

УТВЕРЖДАЮ

Заведующий кафедрой ЭВМ

\_\_\_\_\_\_\_\_\_ Б.В.Никульшин

«\_\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2024 г.

ЗАДАНИЕ

по производственной (технологической) практике студента

Горбачевского Кирилла Витальевича

**1** Место прохождения практики – утверждено приказом по университету № 1049-с, от 31 мая 2024 г.

**2** Срок сдачи студентом производственной (технологической) практики: 1-13 сентября 2024 г.

**3** Задание на производственную (технологическую) практику:

**3.1** Получить индивидуальное задание на практику

**3.2** Прибыть в установленные сроки на место прохождения практики и пройти необходимые на предприятии инструктажи по технике безопасности и охране труда

**3.3** Ознакомиться со структурой предприятия, организацией управления

**3.4** Ознакомиться с системами планирования, организации труда

**3.5** Изучить основные информационные потоки и документооборот

**3.6** Изучить основные экономические и иные показатели работы

**3.7** Изучить действующие на предприятии стандарты

**3.8** Изучить правила и методы разработки и оформления документации

**3.9** Изучить и проанализировать используемые на предприятии системы

**3.10** Ознакомиться с основами проектирования

**3.11** Получить навыки практической эксплуатации ПО

**3.12** Изучить распространяемое организацией оборудование

**3.13** Изучить соответствующее программное обеспечение

**3.14** Изучить методику развёртывания программного обеспечения

**3.15** Провести развёртывание программного обеспечения на оборудовании, проводить сопроводительную работу

**3.16** Подготовка отчета по практике

**4** Содержание пояснительной записки (перечень подлежащих разработке

вопросов):

Введение 1. Описание предприятия. 2. Развертывание программного обеспечения. 3. Разработка html-страниц на основе дизайна. 4. Развертывание Большой Языковой Модели Llama3. Заключение. Список использованных источников.

КАЛЕНДАРНЫЙ ПЛАН

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование этапов  дипломного проекта | Объем  этапа,  % | Срок выполнения этапа | Примечания |
| Изучение структуры, стандартов, систем, используемых на предприятии | 20 | 08.06 – 14.06 |  |
| Изучение теоретической базы для выполнения задания, выданное предприятием | 10 | 15.06 – 19.06 |  |
| Выполнение задания, выданное предприятием | 60 | 20.06 – 30.06 |  |
| Оформление отчета | 10 | 01.07 – 05.07 |  |

Дата выдачи задания: 07.06.2024

Руководитель от предприятия \_\_\_\_\_\_\_\_ И.А. Петрушенко

Руководитель кафедры ЭВМ \_\_\_\_\_\_\_\_ Д.В. Куприянова

ЗАДАНИЕ ПРИНЯЛ К ИСПОЛНЕНИЮ \_\_\_\_\_\_\_\_ К.В. Горбачевский

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 5](#_Toc171588964)

[1 ОПИСАНИЕ ПРЕДПРИЯТИЯ 6](#_Toc171588965)

[1.1. Краткая характеристика организации 6](#_Toc171588966)

[1.2. Организационная структура компании 6](#_Toc171588967)

[1.3. Анализ бизнес-процессов 8](#_Toc171588968)

[2 РАЗВЁРТЫВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ 9](#_Toc171588969)

[2.1 Описание программного обеспечения 9](#_Toc171588970)

[2.2 Установка инструментов 9](#_Toc171588971)

[2.3 Исходный код 9](#_Toc171588972)

[3 РАЗРАБОТКА HTML-СТРАНИЦ НА ОСНОВЕ ДИЗАЙНА 10](#_Toc171588973)

[3.1 Дизайн 10](#_Toc171588974)

[3.2 Разработка 10](#_Toc171588975)

[3.3 Ревью проделанной работы 12](#_Toc171588976)

[4 РАЗВЕРТЫВАНИЕ БОЛЬШОЙ ЯЗЫКОВОЙ МОДЕЛИ LLAMA3 13](#_Toc171588977)

[4.1 Запуска модели 13](#_Toc171588978)

[4.2 Веб-интерфейс 14](#_Toc171588979)

[4.3 Сравнение популярных Больших Языковых Моделей 16](#_Toc171588980)

[ЗАКЛЮЧЕНИЕ 17](#_Toc171588981)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 18](#_Toc171588982)

# ВВЕДЕНИЕ

Производственная практика для студентов – важная составляющего учебного процесса, позволяющая сориентироваться на рынке труда и найти себя в будущей профессии. Работодатели же за время «третьего семестра» получают возможность присмотреться к потенциальным сотрудникам, привлечь на предприятие молодые перспективные кадры. Поэтому, чтобы извлечь из практики максимальную пользу, учащимся вузов важно серьезно к ней отнестись

Целью производственной практики является:

- ознакомление с процессом работы на примере конкретной организации;

- получение практических и теоретических навыков;

- развитие коммуникативных качеств;

- выполнение индивидуальных заданий, полученных до прибытия к месту проведения практики;

- ознакомление со структурой предприятия в целом, со структурой подразделения, где проходила практика;

- сбор материалов, необходимых для написания отчета по практике.

При прохождении практики были поставлены следующие задачи:

- ознакомиться со структурой предприятия, организацией управления;

- изучить основные информационные потоки и документооборот на предприятии;

- выполнить выданное руководителем практики на предприятии индивидуальное задание.

В данном отчете подводится итог прохождения производственной практики на базе ООО «Финмаркет Софт».

# 1 ОПИСАНИЕ ПРЕДПРИЯТИЯ

## 1.1. Краткая характеристика организации

Основной вид деятельности организации – разработка и поддержка программного обеспечения, работа с клиентами. Организация вправе самостоятельно определять конкретные направления своей деятельности в зависимости от конъюнктуры рынка.

Согласно учредительным документам основным видом деятельности организации является деятельность в области компьютерного программирования, а именно:

- разработка сайтов для клиентов компании

- разработка приложения для торговли активами;

- разработка и внедрение сопутсвующих инструментов, по заказу клиента;

- разработка мобильных, десктопных и web-приложений;

- внедрение инструментов торговой аналитики;

## 1.2. Организационная структура компании

Структура компании представлена в виде схемы, изображенной на рисунке 1.1.

Во главе компании находится директор. Он организует всю работу и несет полную ответственность за результаты деятельности компании. В структуру компании входит три отдела:

- бухгалтерский отдел;

- отдел юриспруденции

- отдел найма сотрудников

- отдел разработки.

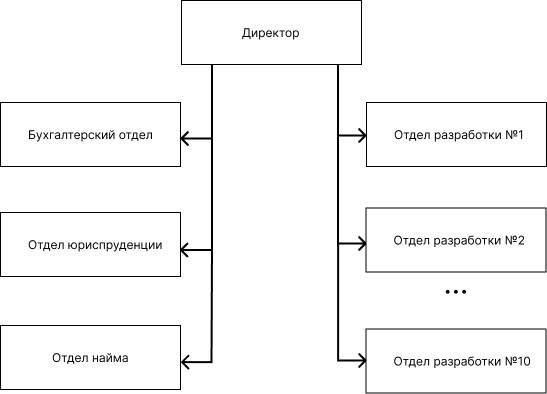


Рисунок 1.1 – Организационная структура компании

Бухгалтерский отдел определяет за собой следующие обязанности:

- Осуществление контроля за своевременным и правильным оформлением первичных документов и законностью совершаемых операций;

- Текущий контроль за целевым и экономным расходованием средств бюджета. Осуществление контроля денежных средств и материальных ценностей по счетам бюджетного учета;

- Начисление и выплата в установленные сроки заработной платы и иных выплат работникам;

- Своевременное проведение расчетов;

- Ведение учета доходов;

Отдел найма сотрудников отвечает за:

- поиск новых сотрудников;

- организацию и участие в собеседованиях;

- общение с сотрудниками;

Отдел разработки отвечает за:

- своевременное выполнение поставленных бизнесом задач;

- исследование системы на наличие недостатков и уязвимостей;

- разработка нового функционала;

## 1.3. Анализ бизнес-процессов

Анализ бизнес-процессов предприятия позволяет более полно понять принцип и логику основных процессов, существующих в рамках предприятия, их взаимосвязь, а также непосредственно устройство организации.

Перечень основных операций и процессов, выполняемых ООО «Финмаркет Софт»:

1) Рассмотрение подробностей заказа от клиента. Уточнение деталей заказа. Расчёт затрачиваемых средств.

2) Работа с клиентом по выбору дизайна системы, основный компонентов и инструментов для реализации.

3) Выбор команды разработки, формирование плана, постановка задач

4) Поэтапная реализация продукта и внедрение нового функционала в систему.

5) Тестовый период, рефакторинг и аудит проделанной работы.

6) Сдача готового проекта заказчику.

При выполнении заказа возможен иной процесс разработки, когда необходимая команда набирается заново через отдел найма.

В ходе разработки клиент имеет полный доступ к ресурсам разработки, видит результат в реальном времени, может запрашивать отчеты о выполненной работе и вносить коррективы в изначальные требования.

На рисунке 1.3.1 показан процесс разработки ПО в компании.

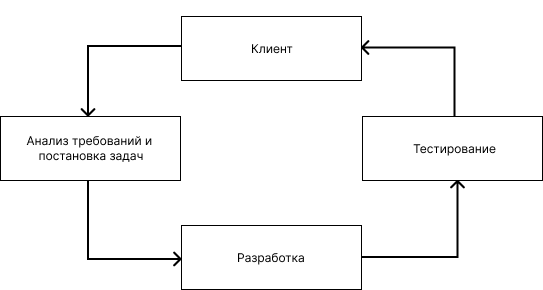


Рисунок 1.3.1 – Бизнес-процесс

# 2 РАЗВЁРТЫВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Развёртывание программного обеспечения (от английского словосочетания «Software deployment») – все действия, которые делают программную систему готовой к использованию. Данный процесс является частью жизненного цикла программного обеспечения.

Так как задачи, поставленные в рамках данной работы являются задачами разработки клиентской стороны. Необходимо установить необходимые инструменты для продуктивной работы.

## 2.1 Описание программного обеспечения

Большинство сайтов компании построены на CMS системе Ghost, являющейся open-source решением для построения сайтов. Разработка ведется с использованием системы контроля версий GitLab. В рамках данной системы автоматизированы процессы деплоя приложений и сохранения изменений.

При деплое активно применяется контейниризация. Так каждый сайт является копией Ghost CMS с собственной темой и базой данный внутри Docker контейнера.

Всего организация поддерживает более 20 сайтов, каждый из которых независим и его работоспособность не сказывается на остальный сайтах.

## 2.2 Установка инструментов

Для работы понадобится установить на компьютер Node.js, набор инструментов и пакетов для работы с JavsScript приложениями, и запустить локальную копию Ghost CMS на своем ПК для разработки темы.

В качестве среды разработки был выбран классический вариант - Visual Studio Code и некоторые расширения для удобства работы с HTML, CSS и JavaScript

## 2.3 Исходный код

Чтобы начать работу необходимо запустить копию сайта на своем локальном компьютере. Необходимые файлы находятся в git репозитории. Используя навыки полученный на лабораторных работах, нужно склонировать репозиторий, скачать нужные зависимости и запустить проект.   
 В файле Readme в корне репозитория приведена четка инструкция по работе с проектом.

# 3 РАЗРАБОТКА HTML-СТРАНИЦ НА ОСНОВЕ ДИЗАЙНА

Типичная задача frontend разработчика – создание стилизованной html страницы на основе дизайна. В компании были разработаны приложения, доступные для скачивания в магазинах на Android и Apple, а также десктопное и веб приложение.

## 3.1 Дизайн

Дизайн представлен в онлайн-сервисе для разработки интерфейсов и прототипирования Figma. На рисунке 3.1.1 можно увидеть часть дизайна.



Рисунок 3.1.1 – Дизайн страницы Technologies

## 3.2 Разработка

После согласования дизайна, разработчик приступает к работе. Была поставлена задача в системе ведения задач Jira сроком на неделю. За это время страница должна быть полность сделана, адаптирована под разными разрешения экрана (телефон, планшет, ноутбук).

Страница проверяется дизайнером и другими членами команды на соотвествие требованиям, после чего осуществляется релиз.

Разработка велась в Visual Studio Code. Основные технологии разработки:

1) SCSS – библиотека, расширяющая функционал обычного CSS, добавляя возможно выносить часто используемые блоки в щаблоны, создавать переменные, использовать наследование стилей.

2) HBS - движок, расширяющий функционал обычного HTML. С помощью данного инструмента можно писать вставки JS кода прямо в html использовать переменные и циклы. Эта библиотека является обязательной при работе с Ghost CMS на котором построены все сайты организации.

3) JavaScript – скриптовый язык программирования. С помощью него делаются анимации появления разных элементов на странице по мере прокрутки.

4) Webpack – сборщик проекта. Webpack берет исходный js файл как параметр и строит из этой точки дерево зависимостей: все необходимые скрипты, стили, шрифты. После этого Webpack генерирует один большой файл, который подключается к html-странице.

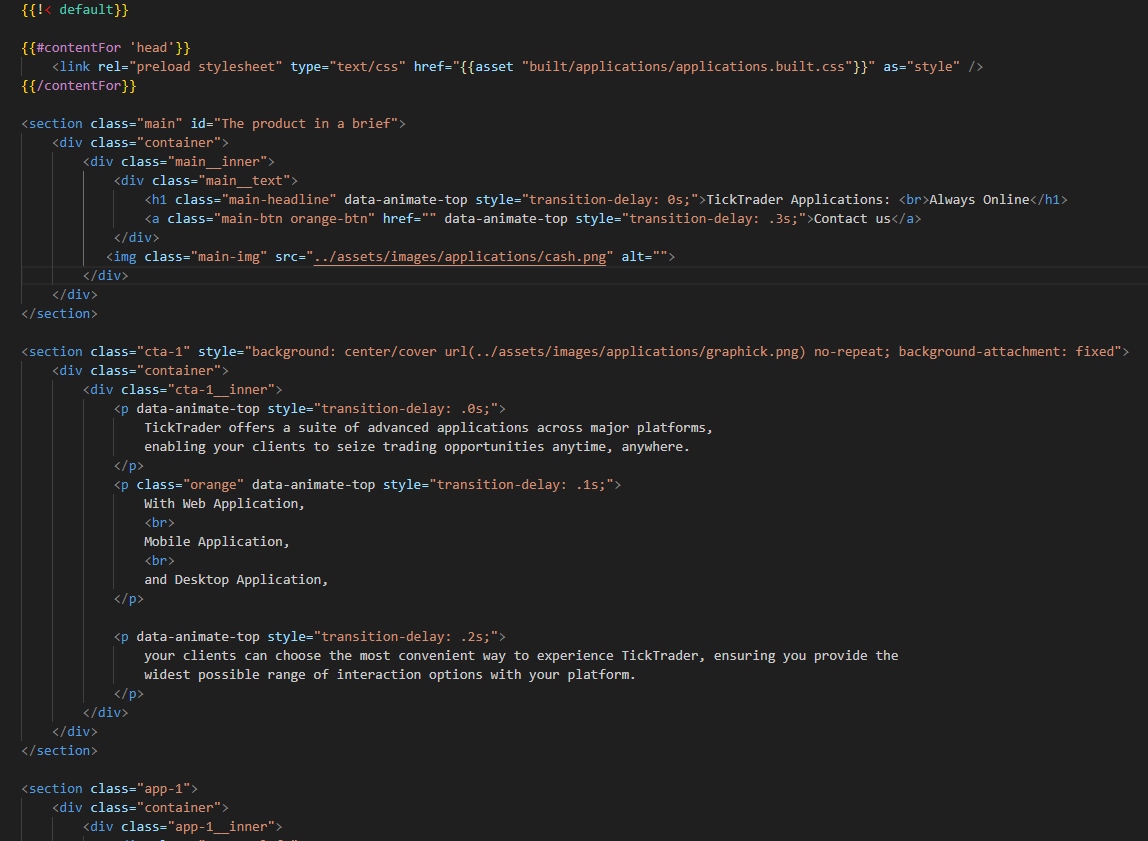


Рисунок 3.2.1 – часть html кода страницы

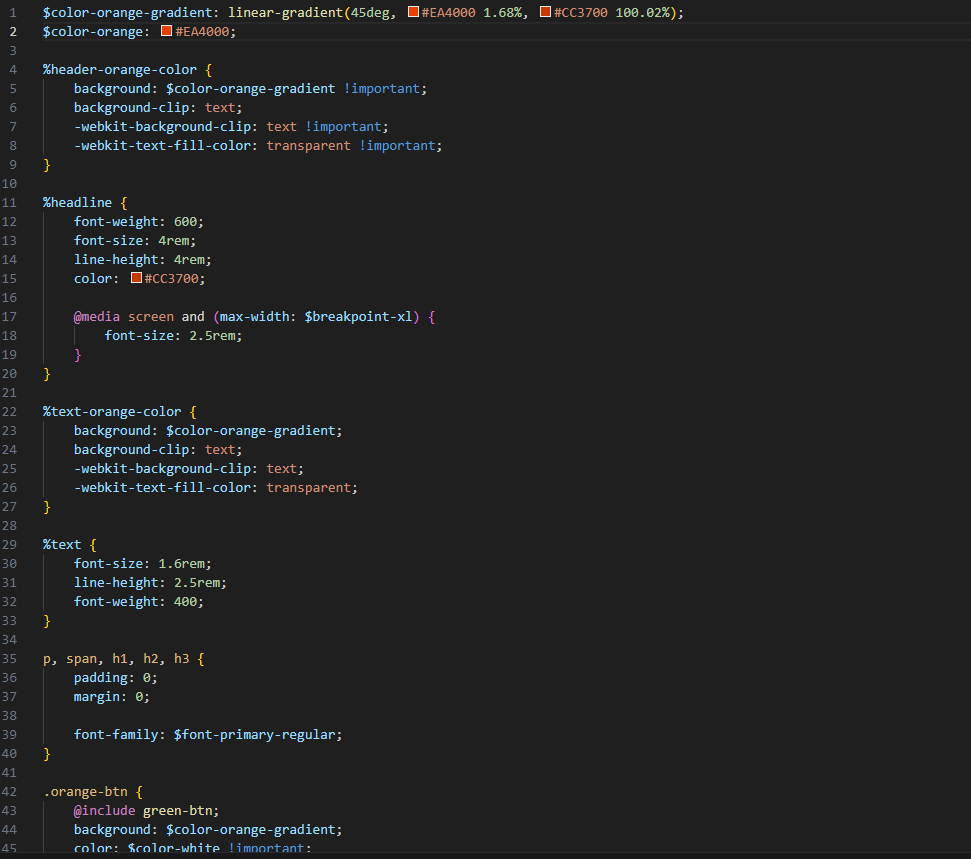


Рисунок 3.2.2 – часть SCSS кода страницы

## 3.3 Ревью проделанной работы

После окончания работы, разработчик отправляет результат своей работы на ревью другим членам команды:

* SEO
* Контент-менеджеру
* Дизайнеру

Команда проверяет новую страницу на наличие несоответствий в дизайне, ошибок в тексте и многим дргуим критериям, влиящим на качество страницы. Если имеются недочеты, создается документ, где описываются ошибки. Разработчик должен оперативно их исправить.

# 4 РАЗВЕРТЫВАНИЕ БОЛЬШОЙ ЯЗЫКОВОЙ МОДЕЛИ LLAMA3

В последние годы ИИ стал активно развиваться и внедрятся в комерческие проекты. С помощью ИИ можно упросить большое количество рутинных задач с которыми сталкиваются сотрудники или пользователи системы. Например чат-бот. Или веб помощник по поиску нужной информации по всему сайту.

На фоне данной тенденции была поставлена задача развернуть собственную Большую Языковую Модель.

Большие Языковые Модели (или LLM) - это очень большие модели [глубокого обучения](https://aws.amazon.com/what-is/deep-learning/), которые предварительно обучены на огромных объемах данных. Лежащий в основе трансформер – это набор [нейронных сетей](https://aws.amazon.com/what-is/neural-network/), каждая из которых состоит из кодера и декодера с возможностью самонаблюдения. Кодер и декодер извлекают значения из последовательности текста и понимают отношения между имеющимися в ней словами и фразами.

Большие языковые модели невероятно гибкие. Одна модель может выполнять совершенно разные задачи, такие как ответы на вопросы, обобщение документов, языковые переводы и составление предложений. LLM могут кардинально повлиять на создание контента и использованию людьми поисковых систем и виртуальных помощников.

В качестве основы была взята открытая языковая модель от компании Meta – Llama3-8b. 8b означает, что каждый ответ – это совокупность анализа 8 миллиардов самых разных параметров.

## 4.1 Запуска модели

Запуск LLM будет произвоидится в Docker контейнере для уменьшения влияния на остальную систему и независимости приложения от сторонних факторов.

Ollama – приложение позволяющее работать с различными языковыми моделями. По сути Ollama является сервером.

Чтобы скачать и запустить Ollama понадобится выполнить всего 2 команды:

1) docker pull ollama/ollama. Скачивание приложения с удаленного репозитория docker hub

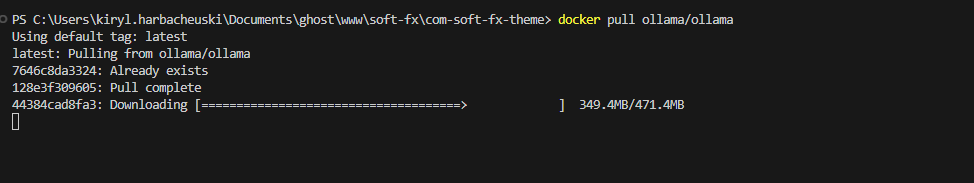


Рисунок 4.1.1 – скачивание приложения Ollama

2) docker run -d -v ollama:/root/.ollama -p 11434:11434 --name ollama ollama/ollama. Запуск приложения на определенном порту, в данном случае 11434.

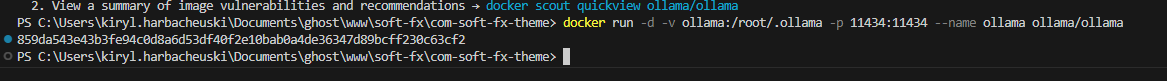


Рисунок 4.1.2 – запуск Ollama

Теперь приложение готово принимать запросы и давать ответы.

## 4.2 Веб-интерфейс

Для удобства использования можно установить web-ui и использовать Ollama как чат. В качестве UI был использован open source проект open-webui.

Наш сервер Ollama находится на порту 11434. Теперь, чтобы запустить UI потребуется всего лишь ввести следующую команду:

docker run -d -p 3000:8080 -e OLLAMA\_BASE\_URL=http://localhost:11434/ -v open-webui:/app/backend/data --name open-webui --restart always ghcr.io/open-webui/open-webui:main

На рисунках 4.2.2 и 4.2.3 показаны скриншоты Open WebUI

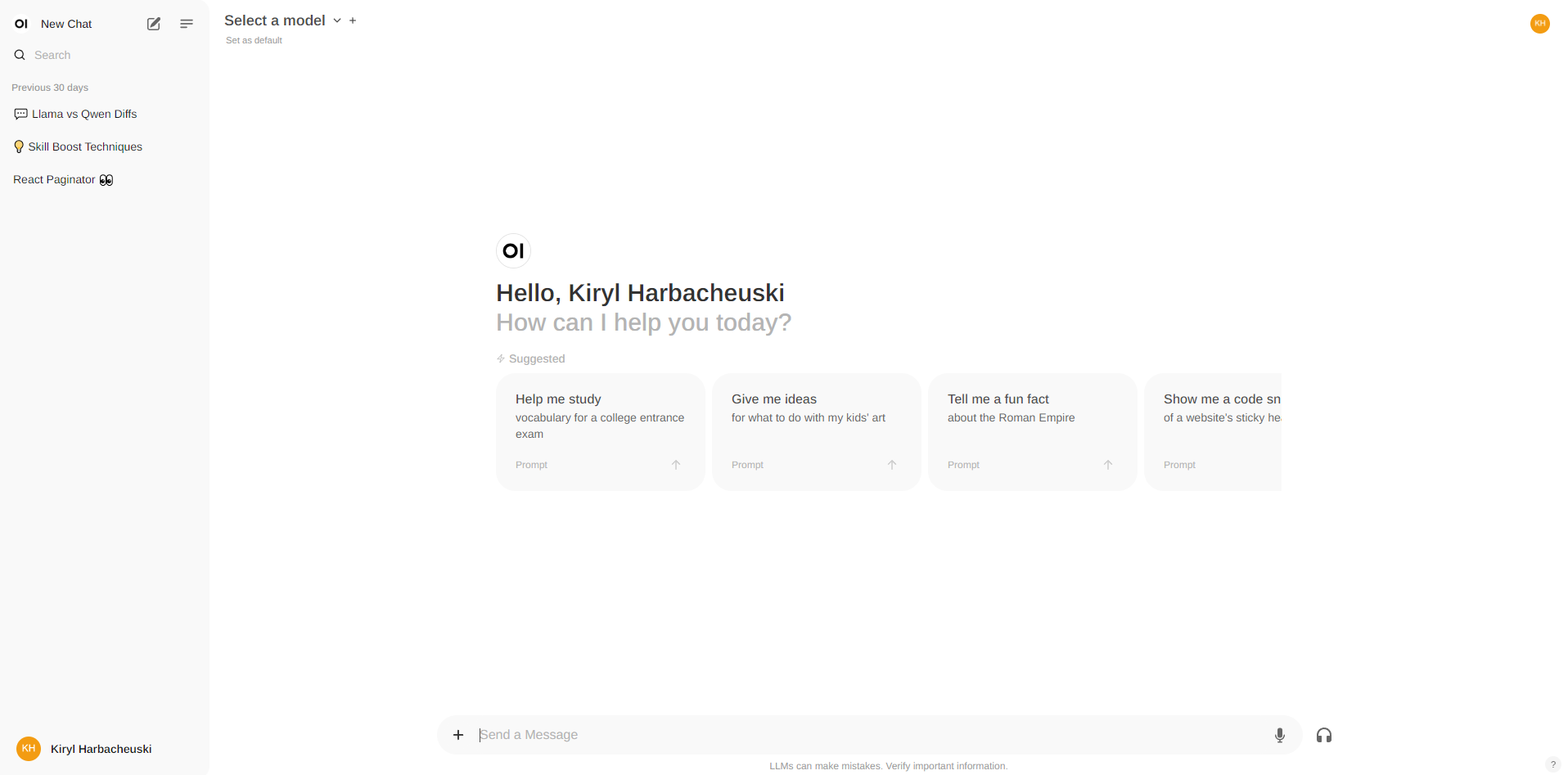


Рисунок 4.2.1 – интерфейс для работы с LLM

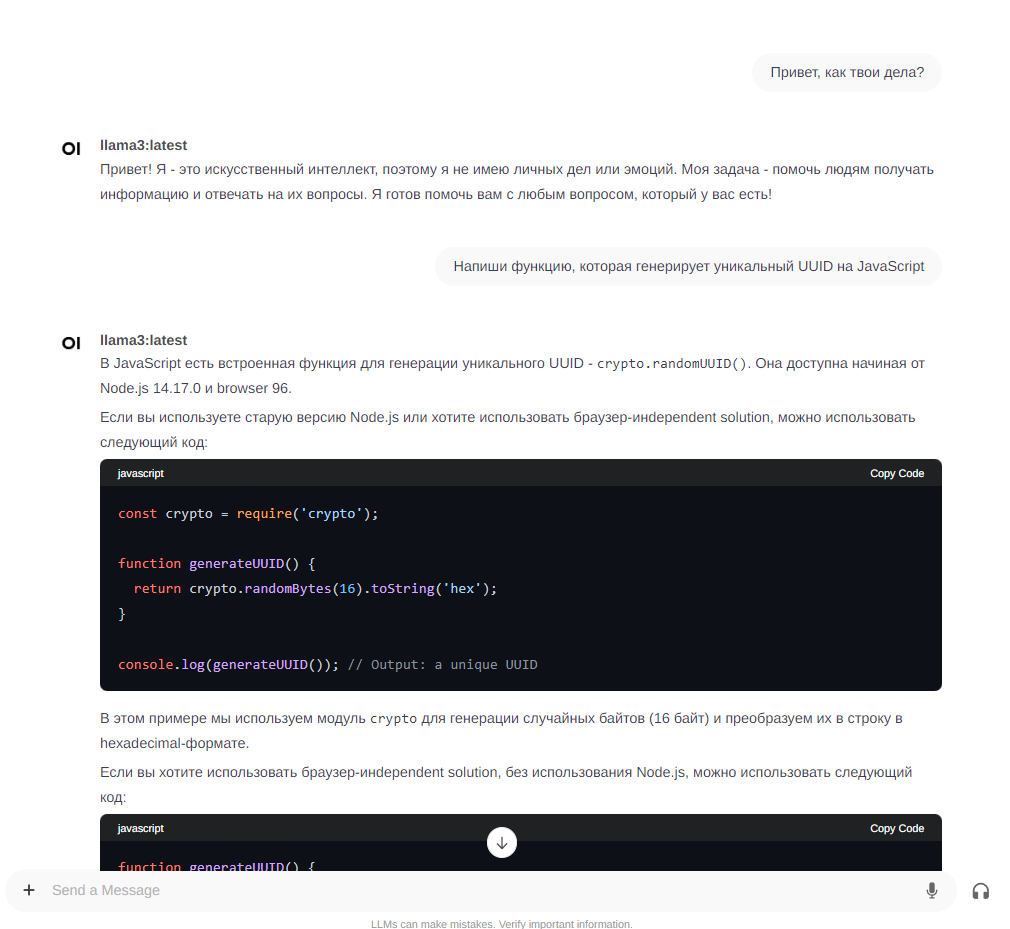


Рисунок 4.2.2 – вопрос и ответ нейросети

Модель при ответе использует ресурсы компьютера, а в частности CPU. По этой причине ответ может быть не очень быстрым. По замерам скорость ответа примерно на 30% ниже чем ответ от ChatGPT 3. Но скорость и точность ответа можно увеличить, если задействовать GPU – графический процессор. Но на рабочих компьютерах в компании не предусмотрено наличии отдельных GPU интерфейсов. Несмотря на это, модель отвечает на вопросы довольно быстро и точно.

Для того чтобы грамотно использовать модель на комерческом проекте нужно дообучить ее на специфичных для компании данных.

Развернутая модель абсолютна независима, может работать без подключения в интернет в локальной сети компании и доступна любому компьютеру в этой сети.

## 4.3 Сравнение популярных Больших Языковых Моделей

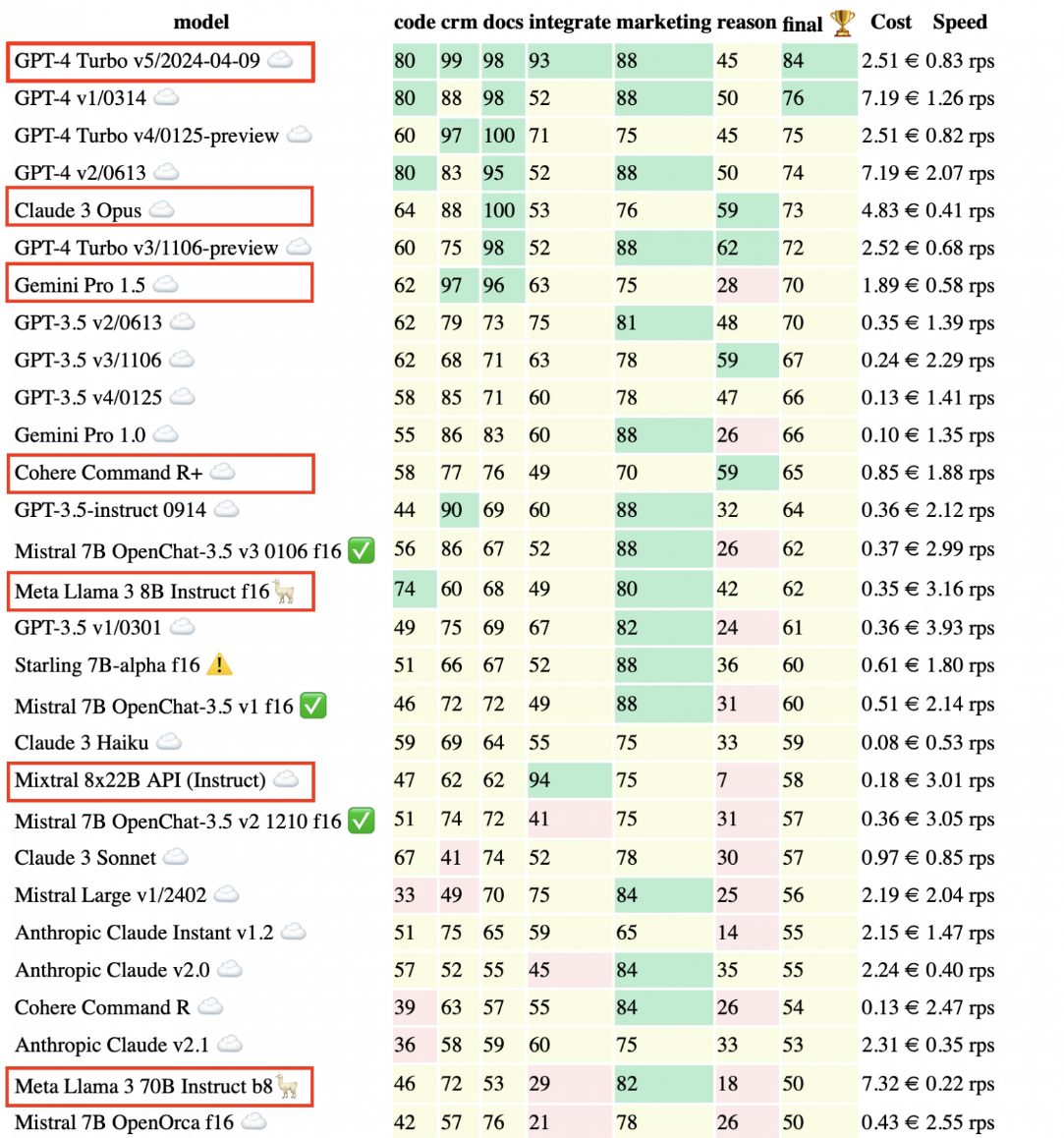


Рисунок 4.1 – сравнение языковых моделей

# ЗАКЛЮЧЕНИЕ

Задачи, поставленные перед началом работы, были выполнены. Изучена работа предприятия, документооборот, стандарты.

Изучена специфика работы и комерческое использование исскуственного интеллекта в виде большой языковой модели (LLM) Llama 3 от компании Meta.

Изучены принципы развёртывания программного обеспечения, неоднократно выполнялись работы по установке программ и системного программного обеспечения.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

[1] Официальный сайт Hugging Face [Электронный ресурс]. – Главная – Режим доступа: https://huggingface.co/– Дата доступа: 05.07.2024

[2] Официальный сайт Soft-Fx [Электронный ресурс]. – Главная– Режим доступа: https://www.soft-fx.com/ – Дата доступа: 05.07.2024

[3] Llama Meta [Электронный ресурс]. – Большая языковая модель Llama 3 – Режим доступа: https://llama.meta.com/llama3/– Дата доступа: 05.07.2024

[4] Репозиторий образов [Электронный ресурс]. – Docker образ инструмента для работы с LLM – Режим доступа: <https://hub.docker.com/r/ollama/ollama> – Дата доступа: 05.07.2024

[5] Официальный сайт [Электронный ресурс]. – Что такое Большие Языковые Модели – Режим доступа: https://aws.amazon.com/ru/what-is/large-language-model/ – Дата доступа: 05.07.2024

[6] Официальный сайт [Электронный ресурс]. – Репозиторий Open WebUI – Режим доступа: https://github.com/open-webui/open-webui– Дата доступа: 05.07.2024