



AppTest

ООО «ИБС АпТест»

ОГРН 1207700109088, ИНН/КПП 9731061862/773101001

Россия, 121205, г. Москва, Сколково инновационного центра тер.,

Большой б-р, дом 42, строение 1, этаж 3, помещ./РМ 1102/10

телефон/факс: +7 (495) 967 80 80

info@ibsapptest.ru, www.ibsapptest.ru

ИИ-ассистент «Стажёр»

Описание программы



Оглавление

1. ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММЕ	3
1.1 Назначение программы	3
1.2 Функции программы	3
1.3. Сведения о технических и программных средствах, обеспечивающих выполнение программы	4
1.3.1. Минимальный состав технических средств	4
1.3.2. Минимальный состав программных средств	4
1.3.3. Используемые технологии.....	5
2. СТРУКТУРА ПРОГРАММЫ	6
2.1 Сведения о структуре программы.....	6
2.2 Сведения о составных частях программы	7
2.2.1 Состав Компонента пользовательского интерфейса.....	7
2.2.2 Состав Компонента управления данными	7
2.3 Сведения о связях между составными частями программы.....	8
2.3.1 Взаимодействие Компонента пользовательского интерфейса с другими компонентами продукта	8
2.3.2 Взаимодействие Компонента управления данными с другими компонентами продукта.....	8
2.4 Сведения о связях программы с другими программами	8
ПЕРЕЧЕНЬ ТЕРМИНОВ	9
ПЕРЕЧЕНЬ СОКРАЩЕНИЙ	9



1. ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММЕ

1.1 Назначение программы

Программа является ИИ-ассистентом, задача которой автоматизировать часть работ по тест-анализу, которые обычно выполняет инженер-тестировщик. В рамках программы можно генерировать тестовую документацию, проводить анализ требований, получать техническую консультацию. Программа состоит из основного модуля, включающего в себя фронтную и серверную части, базы данных, разделенной на базу данных приложения и базу данных RAG.

Программа предназначена для работы с локальными большими языковыми моделями. Большая языковая модель находится под контролем заказчика. В программе отсутствуют критические зависимости от иностранных сервисов и лицензионных программных продуктов.

1.2 Функции программы

Продукт состоит из единого функционального модуля, который позволяет пользователю использовать Большую языковую модель (далее - LLM) для генерации определенного рода тестовой документации, а также анализа требований и технической консультации по заранее сформированным пресетам, используя технологию RAG для обогащения контекста проектными данными. Продукт выполняет следующие функции:

- Ревью требований
- Создание чек-листа
- Написание тест-кейсов
- Подготовка тестовых данных
- Техническая консультация

Программа обеспечивает конфиденциальность проектных данных, так как использует локально развернутую LLM и RAG (Retrieval Augmented Generation), а также обеспечивает диверсификацию данных за счет загрузки данных о проекте в RAG.



1.3. Сведения о технических и программных средствах, обеспечивающих выполнение программы

1.3.1. Минимальный состав технических средств

Продукт может быть запущен на технических средствах со следующими характеристиками:

Клиентская часть:

CPU: от 2 ядер

RAM: от 2 ГБ

Серверная часть:

Минимальная конфигурация:

CPU: 4 ядра

RAM: 16 ГБ

SSD: 200 ГБ (зависит от количества проектных данных, которые нужно загрузить в RAG)

Рекомендуемая конфигурация

CPU 8 ядер

RAM 32 ГБ

SSD 200 ГБ (зависит от количества проектных данных, которые нужно загрузить в RAG)

1.3.2. Минимальный состав программных средств

Для функционирования ИИ-ассистента необходимо наличие браузера Chrome, Chromium или Yandex browser и ОС, которая поддерживает работу этих браузеров.

На серверной части необходимо:

- ОС Linux x64 или Windows Server начиная от 2016
- Docker
- Git

1.3.3. Используемые технологии

Продукт построен на принципах трехуровневой архитектуры.

Продукт содержит фронттовую часть, обеспечивающую доступность клиентской части приложения, серверную часть, которая отвечает за обработку ввода пользователя, работу с RAG и LLM, и сервер базы данных, где содержатся данные приложения и RAG.

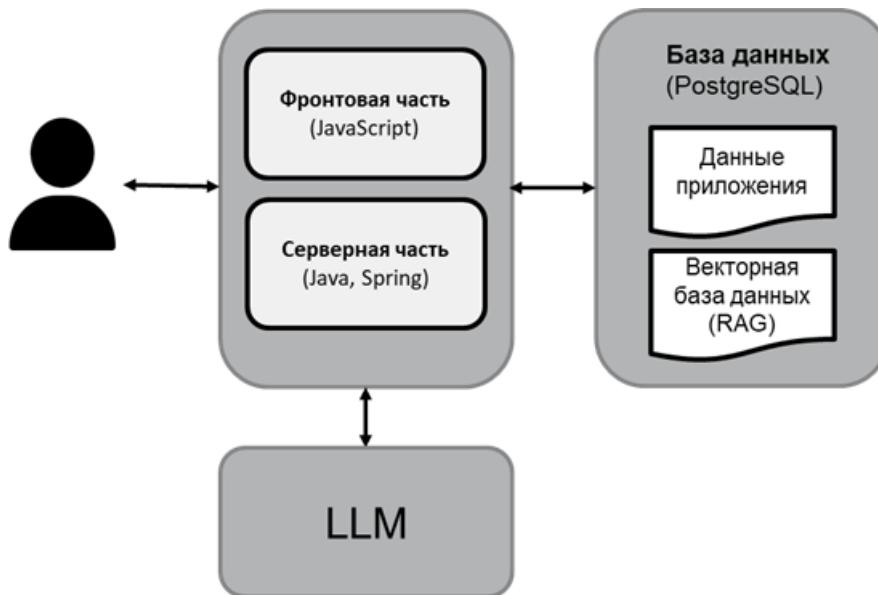


Рис 1. Схема архитектуры

Продукт написан на языках Java (серверная часть) и JavaScript, HTML (Фронттовая часть)

- **Веб-сервер:** Tomcat
- **Фронт:** Bootstrap
- **Технологии работы с ИИ:** Spring AI, JPA, Hibernate, Security
- **База данных (для хранения данных приложения и RAG):** PostgreSQL, pgvector
- **Контейнеризация:** Docker, Docker Compose

2. СТРУКТУРА ПРОГРАММЫ

2.1 Сведения о структуре программы

Структурно ИИ-ассистент «Стажёр» состоит из следующих структурных компонентов:

- Компонент пользовательского интерфейса;
- Компонент управления данными.

Компонент пользовательского интерфейса содержит функциональные блоки, необходимые для задействования всего функционала Продукта со стороны пользователя. Взаимодействие между компьютером пользователя (браузером) и Продуктом происходит по протоколу HTTP или HTTPS.

Компонент управления данными содержит логические блоки, отвечающие за получение данных от компонента пользовательского интерфейса и управляющие взаимодействием Продукта с LLM и Базой данных. Взаимодействие с LLM происходит по протоколу HTTPS через REST API. Взаимодействие с Базой данных Postgres происходит при помощи драйвера JDBC через протокол TCP.

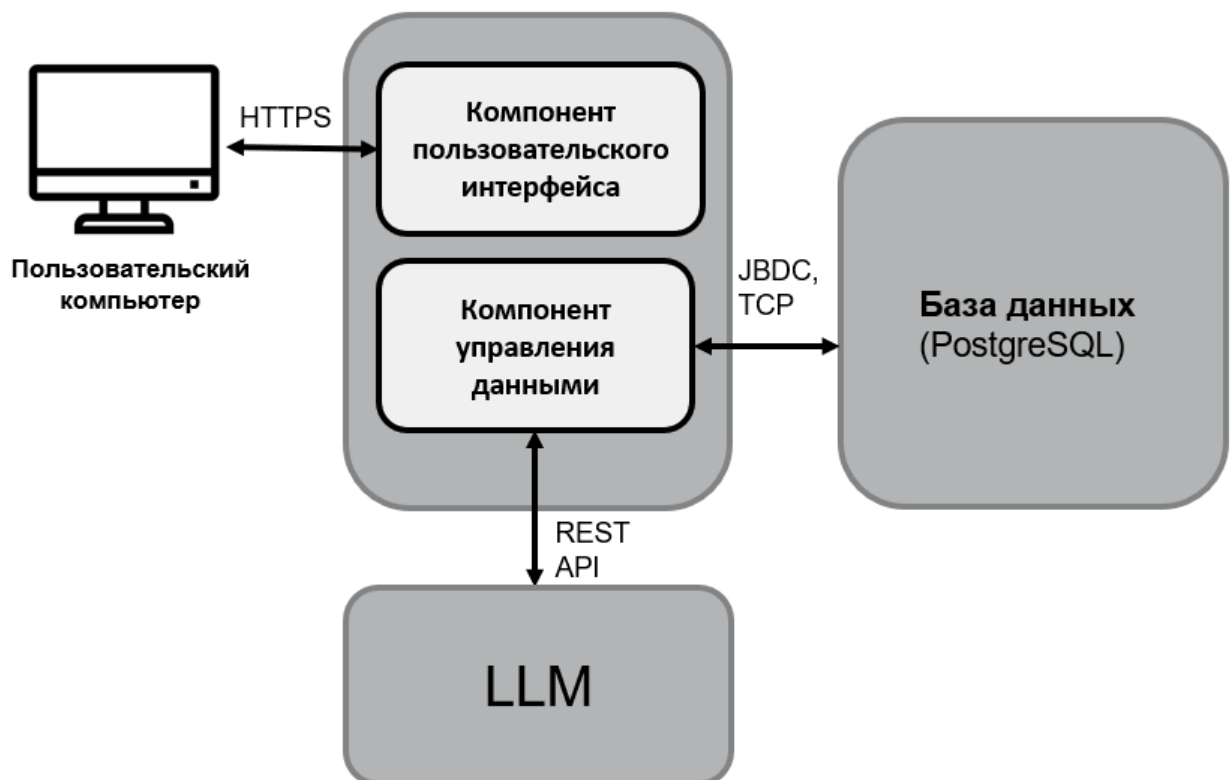


Рис 2. Структурная схема продукта



2.2 Сведения о составных частях программы

2.2.1 Состав Компонента пользовательского интерфейса

Компонент пользовательских интерфейсов включает в себя следующие блоки:

Общая часть:

- Авторизация пользователя
- Регистрация нового пользователя

Пользовательская часть:

- Составление чек-листа
- Составление тест-кейсов
- Генерация тестовых данных
- Ревью требований
- Техническая консультация

Административная часть:

- Teams (Команды)
- Users (Пользователи)
- RAG
- Prompts (Промпты).

2.2.2 Состав Компонента управления данными

Компонент управления данными включает в себя следующие блоки:

- блок обработки событий

Блок работы с базой данных

- блок работы с LLM
 - Формирование промпта
 - Отправка промпта в LLM
 - Получение ответа от LLM
 - Отображение ответа от LLM
- блок работы с RAG
 - Обращение в RAG
 - Обогащение промпта информацией из RAG
- блок работы с ролевой моделью



- блок конвертации файлов форматов pdf, doc, docx в текстовый формат
- блок парсинга промптов для обеспечения пайплайна (передачи информации из одной вкладки приложения в другую).

2.3 Сведения о связях между составными частями программы

Компоненты продукта связаны между собой посредством программных интерфейсов и сетевых протоколов взаимодействия: HTTP/HTTPS, TCP, JDBC.

2.3.1 Взаимодействие Компонента пользовательского интерфейса с другими компонентами продукта

Компонент пользовательского интерфейсов взаимодействует с Компонентом управления данными и является представлением его функций в виде пользовательского интерфейса. Взаимодействие осуществляется через веб-сервисы (HTTP-запросы).

2.3.2 Взаимодействие Компонента управления данными с другими компонентами продукта

Компонент управления данными взаимодействует с Компонентом пользовательского интерфейса, СУБД и LLM. Взаимодействие с Компонентом пользовательского интерфейса осуществляется через веб-сервисы (HTTP-запросы). Взаимодействие с СУБД осуществляется через драйвер JDBC.

2.4 Сведения о связях программы с другими программами

Продукт может взаимодействовать с внешними источниками, приёмниками и хранилищами данных, к которым относятся:

- сервер Confluence
- Различные TMS-системы с открытым API.



ПЕРЕЧЕНЬ ТЕРМИНОВ

Термин	Определение
Продукт	Условное обозначение ИИ-ассистента «Стажёр»

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

Термин	Определение
LLM	Большая языковая модель (Large Language Model)
RAG	Генерация, дополненная поиском (Retrieval-Augmented Generation)
HTTP	Протокол передачи гипертекста HyperText Transfer Protocol
HTTPS	Безопасный протокол передачи гипертекста HyperText Transfer Protocol Secure
JDBC	Соединение с базами данных на Java Java DataBase Connectivity
API	Описание способов взаимодействия одной компьютерной программы с другими Application programming interface
REST	Архитектурный стиль взаимодействия компонентов распределенного приложения в сети Representational State Transfer
TCP	Протокол управления передачей данных Transmission Control Protocol
СУБД	Система управление базами данных