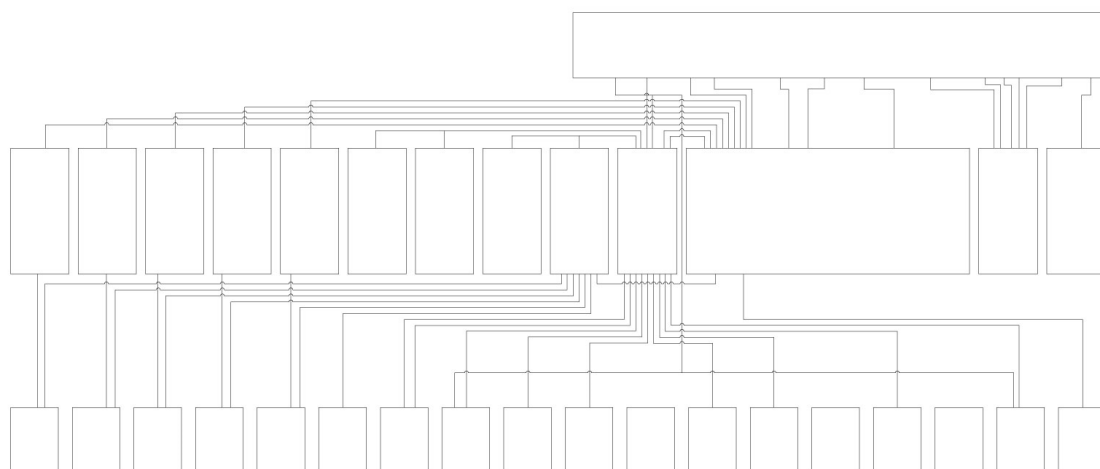


«Цифровая Образовательная Платформа «ELEUM®»

(ПК «ЦОП «ELEUM®»)

Описание архитектуры

Версия 1.0 от 23.04.2022



Инт. № подл.	Подп. и дата
Инт. № дубл.	Взам. инв. №
Подп. и дата	Подп. и дата

Описание архитектуры программного комплекса ПК «ЦОП «ELEUM®»

«Цифровая Образовательная Платформа «ELEUM®» (ПК «ЦОП «ELEUM®») предназначен для:

- обучения электротехнического персонала навыкам по эксплуатации цифровых устройств релейной защиты и противоаварийной автоматики, работающих в соответствии со стандартом МЭК 61850;

- приобретения навыков выполнения оперативных переключений в нормальном и аварийном режимах работы на оборудовании, работающем по протоколу МЭК 61850;

- повышения общего качества обучения электротехнического персонала.

Архитектура продукта и его компонентов (модулей):

ПК «ЦОП «ELEUM®» строится на основе клиент-серверной архитектуры, имеет клиентскую часть на основе веб-интерфейса и ориентирован на многопользовательский режим работы. Пользователи имеют возможность работать с ПК при помощи следующих основных браузеров: Safari, Google Chrome, Firefox, Opera.

Архитектура приложения включает в себя серверы приложений, систему управления базами данных, брокер сообщений и может быть визуализирована следующим образом (Рисунок 1 – Архитектура ПК «ЦОП «ELEUM®»):

Инф. № подл.	Подп. и дата	Инф. № дубл.	Взам. инф. №	Подп. и дата					
Лит.	Изм.	№ докум.	Подп.	Дата	ПК «ЦОП «ELEUM®». Описание архитектуры				Лист
									2

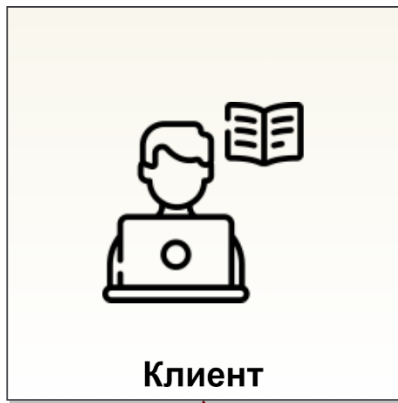


Рисунок 1 – Архитектура ПК «ЦОП «ELEUM®»

Инф. № подл.	Подп. и дата
Инф. № докл.	Подп. и дата
Взам. инф. №	Подп. и дата

Лит.	Изм.	№ докum.	Подп.	Дата

В данной архитектуре мы имеем следующие компоненты:

1. Клиент: Клиент – это пользовательский интерфейс, через который пользователи взаимодействуют с приложением через веб-браузер.

В данной архитектуре мы имеем два сервера приложений Tomcat и Wildfly, на которых работают два отдельных приложения: “Образовательная платформа” (с реализацией модулей администрирования, обучения и тестирования) и “Модуль интеграции (SCADA)”.

2. Сервер приложений Tomcat – это веб-сервер и контейнер сервлетов с открытым исходным кодом. Он обрабатывает входящие HTTP-запросы от клиента, реализует логику приложения и генерирует ответы.
3. Сервер приложений Wildfly – сервер приложений на базе Java. В данной архитектуре он выполняет приложение SCADA (Supervisory Control and Data Acquisition). Wildfly управляет процессами, связанными со SCADA, такими как сбор данных в реальном времени, мониторинг и функции управления цифровыми двойниками.
4. СУБД Postgres: Postgres – это реляционная система управления базами данных с открытым исходным кодом. Она хранит и управляет данными, используемые приложениями. СУБД выполняет все операции по хранению, поиску и управлению данными.
5. Kafka (брокер сообщений): – это распределенная потоковая платформа, используемая для построения конвейеров данных и потоковых приложений в реальном времени. Она выступает в качестве центрального канала связи между приложениями. Большинство коммуникаций и обмен информацией между различными компонентами и приложениями архитектуры происходят через Kafka.

Информационные потоки между компонентами можно изобразить следующим образом:

1. Клиент отправляет HTTP-запросы на сервер приложений Tomcat.
2. Сервер приложений Tomcat получает запрос. В случае необходимости перенаправляет его на сервер приложений Wildfly.
3. Tomcat или Wildfly обрабатывает запрос и генерируют ответ.

4. Кроме того, все приложения взаимодействуют друг с другом через Kafka. Они публикуют сообщения в очереди Kafka и получают сообщения из соответствующих очередей. Это позволяет обеспечить асинхронное и раздельное взаимодействие между приложениями.

В целом данная архитектура обеспечивает модульность, масштабируемость и гибкость за счет использования отдельных серверов приложений для разных приложений, надежной базы данных для управления данными и Kafka в качестве системы обмена сообщениями для межприкладного взаимодействия.

Подп. и дата	
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	
Инв. № подл.	