

Описание технической архитектуры программного обеспечения

Платформа SciOffice предназначена для создания бизнес-приложений, которые позволяют управлять данными организации. Бизнес-приложение представляет собой совокупность электронных документов в системе SciOffice, выполняющих правила бизнес-логики. Система может действовать как самостоятельное решение или быть частью другой системы. В случае использования в роли подсистемы взаимодействие с существующими ИС происходит с помощью API интерфейса. Основными действующими лицами являются администратор, создающий бизнес-приложения и пользователь, редактирующий документы приложения. На рисунке 1 представлена схема системного контекста платформы.

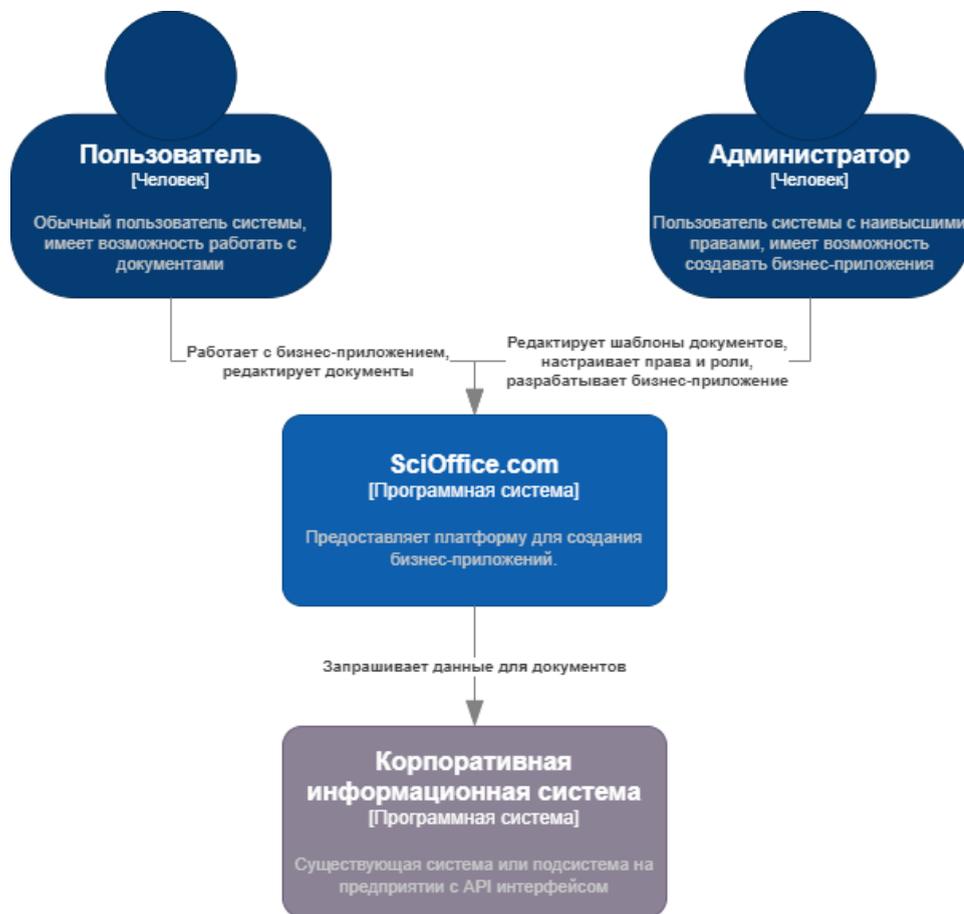


Рисунок 1 - Схема системного контекста

Стоит рассмотреть размещение контейнеров системы. Веб-приложение, написанное с использованием языка JavaScript и библиотеки React предоставляет пользовательский интерфейс. Отметим, что выполнение бизнес-логики приложений происходит с помощью

триггеров, которые выполняются при каком-либо событии документа, например, изменении значения поля. Таким образом веб-приложение используется для запуска этих триггеров. Собранные команды передаются на ядро системы, называемое сервером приложений, посредством GraphQL запросов. Сервер приложений принимает их и обрабатывает. В качестве хранилища данных используется две СУБД:

- 1) реляционная MySQL для хранения данных пользователей и настроек системы;
- 2) документоориентированная MongoDB для хранения данных документов.

Сервер разработан на технологическом стеке Python + Django, таким образом обращение к БД происходит посредством библиотек mysqlclient и pymongo. На рисунке 2 представлена схема контейнеров платформы SciOffice.

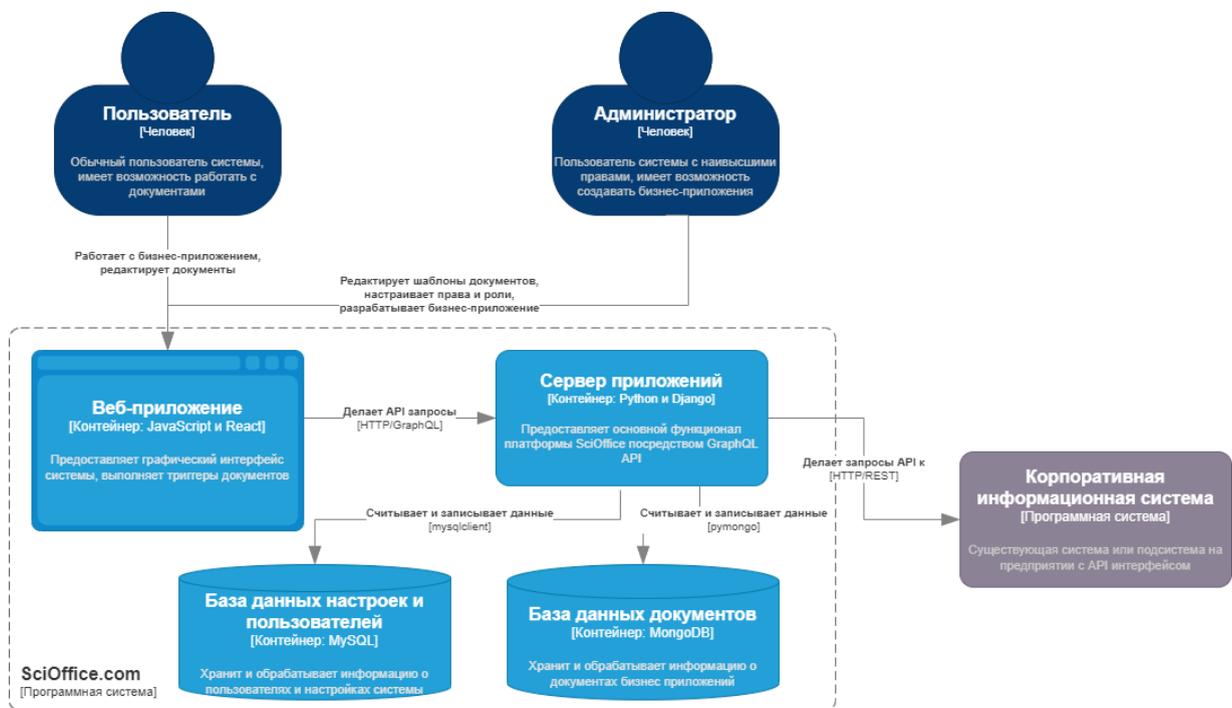


Рисунок 2 - Схема контейнеров системы SciOffice

Рассмотрим сервер приложений более детально. Он состоит из компонентов, которые отвечают за разный функционал в целом дающий все возможности по управлению документами. Каждый компонент представлен приложением Django, таким образом являющийся независимой единицей. Компоненты разработанные в системе перечислены ниже.

- Система экспорта - выполняет функции обработки введенных данных в документ, формированию из них экспортного документа в формате docx или html. Для формирования этого документа используется шаблон, который добавляется на этапе разработки бизнес-приложения администратором. Перед формированием происходит выполнение скрипта преобработчика - специального исполняемого файла, выполняющего преобразование входных данных для документа с целью их дальнейшего использования в шаблоне. Так, преобработчик позволяет обработать сложную текстовую конструкцию и затем использовать ее значение при формировании документа. Сформированный документ возвращается пользователю с возможностью загрузки в личное хранилище.
- Система импорта - обрабатывает функции импорта данных на основе составленного шаблона. В нем прописаны правила сопоставления полей из входного файла формата JSON или XML с полями разработанного документа. Произведя импорт, можно получить готовый набор документов, не прибегая к ручному вводу информации. В этом компоненте также доступно использование преобработчика.
- Система взаимодействия с API клиента - специальный компонент, который позволяет посредством запросов к REST API клиента получить необходимые данные для документов приложения. Он позволяет настроить доступ к ИС так, как работает обычный документ SciOffice, к которому применимы все стандартные методы обработки. Полученные данные по API компонент кеширует в базе данных документов MongoDB для повторного использования.
- Файловая система - компонент, выполняющий функции доступа к документам. Позволяет разделить их по папкам, тем самым добавляя структурированность. Позволяет создавать документы и шаблоны бизнес-приложения.
- Редактор документов является основным компонентом, с которым взаимодействует пользователь. Он позволяет редактировать документ, здесь обрабатываются команды триггера ядра. Редактор представлен древовидной структурой, где каждое поле может быть помещено в блок.
- Редактор шаблонов используется для разработки шаблонов документов, по которым происходит их создание. Каждый шаблон позволяет задать любую структуру вложенности полей. Также в данном компоненте происходит настройка триггеров на полях и

добавление в систему шаблонов экспорта и импорта с преобразователями.

- Редактор миграций - позволяет настроить изменение уже созданных документов в случае, если изменилась структура шаблона этих документов.
- Пользователи - дает возможность ограничить доступ к системе только для зарегистрированных пользователей.
- Система прав доступа - позволяет выделить роли пользователей, задать для каждого определенные права, и назначить их на доступ к документам.

Схематически взаимодействие компонентов между собой можно отобразить на схеме компонентов (рис. 3).

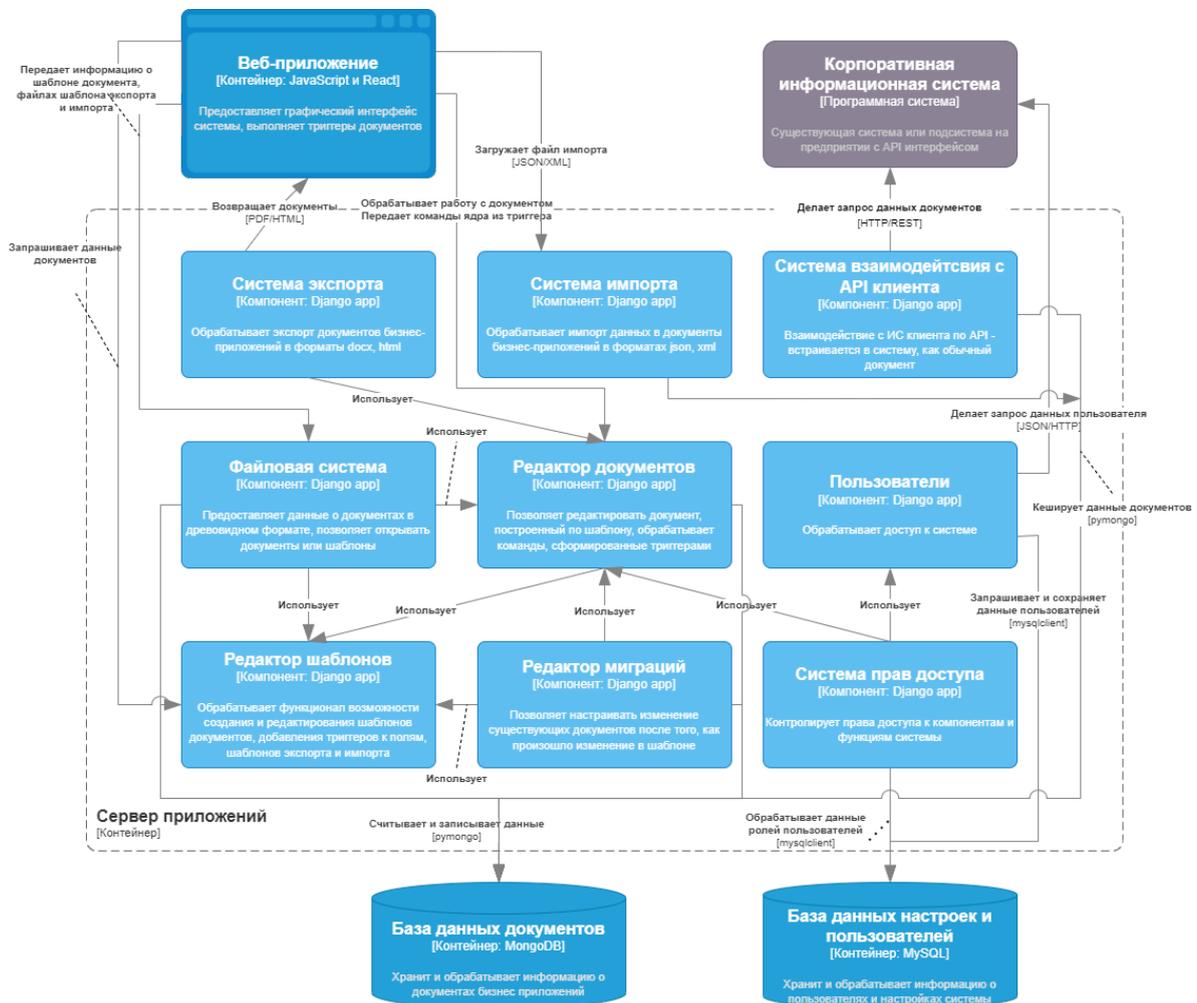


Рисунок 3 - Схема компонентов контейнера “Сервер приложений”

Таким образом, описанная выше архитектура позволяет отвечать основным критериям качества программного обеспечения: эффективности, эргономичности, функциональности, модифицируемости.