

**Федеральное государственное автономное образовательное учреждение
высшего образования
«Санкт-Петербургский политехнический университет Петра Великого»
ОГРН 1027802505279
ИНН 7802719547**

**Описание функциональных характеристик программного
обеспечения «ПОЛАНИС (Политехническая аналитическая
информационная система)»**

СОДЕРЖАНИЕ

1.	Общее положение	4
2.	Общее описание Программы	5
2.1.	Назначение Программы	5
2.2.	Состав программы	5
3.	Описание функций программы	6
3.1.	Модуль управления пользователями	6
3.2.	Модуль управления учетными записями пользователей	6
3.3.	Библиотека для генерации токенов.....	8
3.4.	Модуль управления ролями и правами	8
3.5.	Модуль интеграции с LDAP	9
3.6.	Модуль нотификации	10
3.7.	Модуль информации о событиях	10
3.8.	Менеджер лицензий.....	11
3.9.	Файловый менеджер	12
3.10.	Модуль импорта и экспорта данных	13
3.11.	Модуль управления группами и проектами	13
3.12.	Модуль управления входными данными.....	14
3.13.	Модуль управления вычислителями	16
3.14.	Модуль управления формами параметров.....	16
3.15.	Модуль управления расчетами	18
3.16.	Модуль управления визуализацией.....	20
3.17.	Модуль локализации	21

3.18.	Библиотека общей функциональности	21
3.19.	Модуль управления приложением	22
3.20.	Вспомогательные модули.....	23
3.21.	Модуль пользовательского интерфейса.....	23
3.22.	Сервис обработки задач.....	24
4.	Пользовательские сценарии	26

1. ОБЩЕЕ ПОЛОЖЕНИЕ

Основной целью программного продукта «ПОЛАНИС (Политехническая аналитическая информационная система)» является обеспечение возможности управления данными и расчетами компьютерного моделирования производственных и технологических процессов.

Настоящий документ содержит описание функциональных характеристик программного обеспечения «ПОЛАНИС (Политехническая аналитическая информационная система)» (далее – Программа).

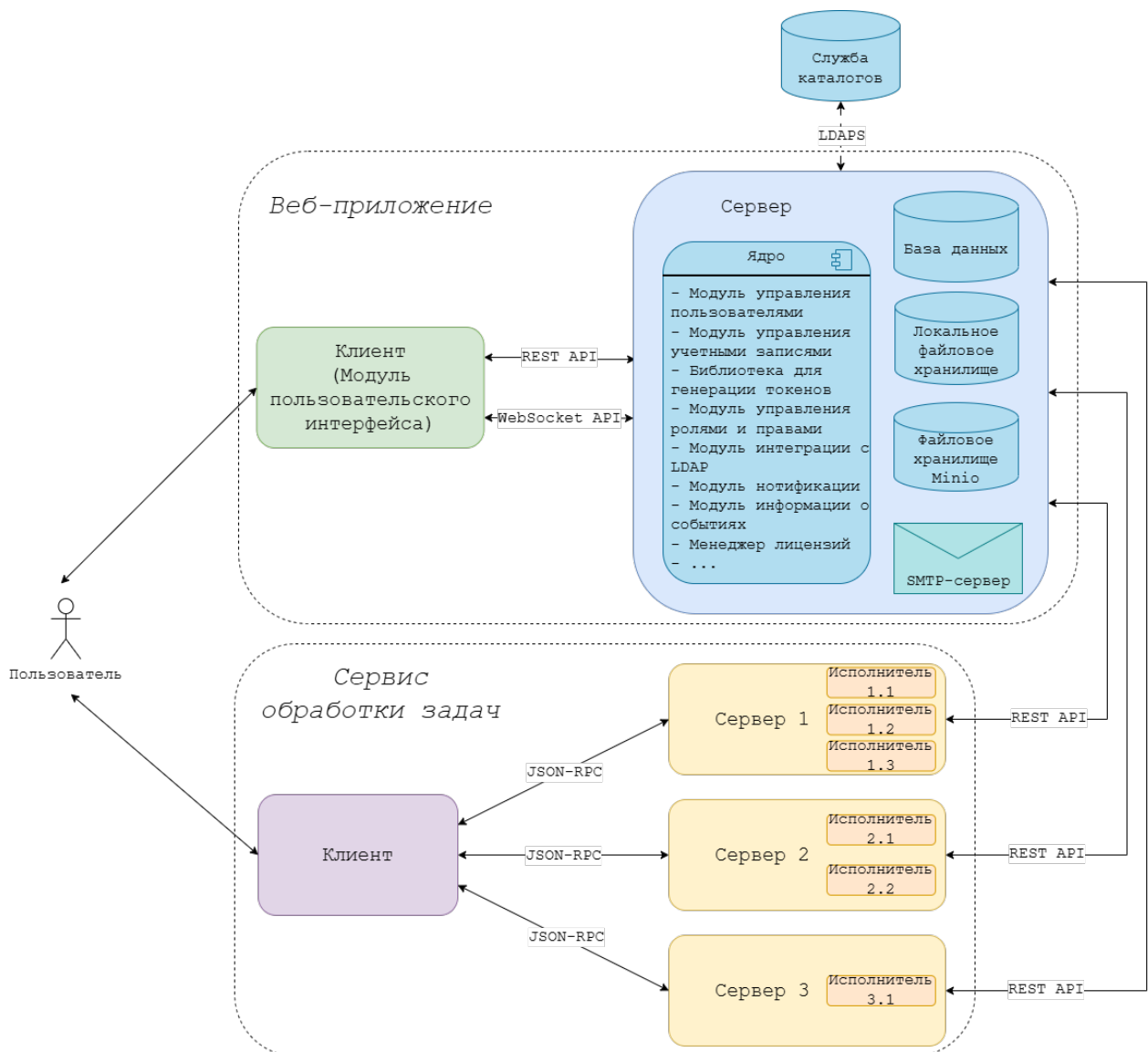
2. ОБЩЕЕ ОПИСАНИЕ ПРОГРАММЫ

2.1. Назначение Программы

Программа предназначена для управления данными и расчетами компьютерного моделирования производственных и технологических процессов с использованием нейросетевых технологий.

Программа может быть использована в любой сфере, где применяются методы обработки, анализа, моделирования и визуализации данных.

2.2. Состав программы



3. ОПИСАНИЕ ФУНКЦИЙ ПРОГРАММЫ

3.1. Модуль управления пользователями

Модуль управления пользователями предназначен для организации взаимодействия с пользователями системы и выполнения основных операций – создания, удаления и администрирования. Он обеспечивает хранение, обработку и актуализацию персональных данных.

Модуль предоставляет пользователям возможность редактировать собственные профили: изменять персональные сведения, загружать или удалять аватары. В частности, можно обновлять такие данные, как ФИО, наименование предприятия, подразделение, должность, адрес электронной почты, контактный телефон и другие параметры, что способствует поддержанию актуальности информации в системе.

Для администраторов предусмотрены расширенные функции управления пользователями. Они могут задавать формат адресов электронной почты, используемых при регистрации, чтобы ограничить доступ только для пользователей, чьи адреса соответствуют корпоративным шаблонам. В случае несоответствия указанного адреса требованиям администратор может подтвердить или отклонить регистрацию. Также администраторам доступен просмотр детализированной информации обо всех пользователях, что обеспечивает прозрачность и контроль процессов управления.

3.2. Модуль управления учетными записями пользователей

Модуль управления учетными записями пользователей предназначен для обеспечения надежного и безопасного функционирования процессов аутентификации и защиты пользовательских данных. Он реализует

функции создания, изменения и контроля учетных записей, а также механизмы, направленные на повышение уровня безопасности.

Модуль предоставляет возможность регистрации учетных записей пользователей. После активации по ссылке и установки пароля введенные данные шифруются с применением современных алгоритмов хэширования, что гарантирует их безопасное хранение.

Одной из ключевых задач модуля является реализация безопасной аутентификации – проверка логинов и паролей, поддержка различных методов верификации, а также предоставление инструментов для сброса, изменения и восстановления паролей. Это обеспечивает пользователям удобное и безопасное управление доступом к системе.

Безопасность учетных записей достигается за счет применения строгих политик паролей, предусматривающих минимальную и максимальную длину, обязательное использование специальных символов, заглавных букв и цифр, а также требование регулярной смены паролей. Кроме того, система поддерживает защитные меры, включая ограничение числа неудачных попыток входа и временную блокировку учетной записи при превышении допустимого лимита.

Администраторы могут настраивать параметры безопасности учетных записей в соответствии с политикой организации и требованиями системы. Также модуль обеспечивает мониторинг активности пользователей – отслеживаются последние входы, история изменений паролей и попытки подозрительных входов. Дополнительно реализовано автоматическое завершение пользовательской сессии после определенного периода бездействия.

Таким образом, модуль управления учетными записями пользователей является ключевым компонентом системы безопасности, обеспечивая защиту данных и предотвращая несанкционированный доступ.

3.3. Библиотека для генерации токенов

Библиотека отвечает за генерацию токенов, используемых для аутентификации и авторизации пользователей в системе. Access-токены служат основным инструментом безопасности, обеспечивая доступ пользователя к функционалу системы в соответствии с его правами.

Дополнительно библиотека формирует Refresh-токены, позволяющие продлевать срок действия Access-токенов без повторного входа в систему. Это обеспечивает удобство работы пользователей при сохранении высокого уровня защиты.

Также предусмотрена генерация сервисных токенов, предназначенных для безопасного взаимодействия внешних сервисов с системой. При их создании задается набор разрешений, определяющий уровень доступа интегрируемого сервиса к функционалу системы.

Все типы токенов имеют ограниченный срок действия, что повышает общую безопасность: даже при возможной компрометации токен станет недействительным через заданный промежуток времени.

Для генерации токенов используется алгоритм шифрования HS256, обеспечивающий надежную защиту данных и исключающий возможность подделки или изменения токенов без знания секретного ключа.

Кроме того, библиотека предоставляет механизмы получения информации о текущем аутентифицированном пользователе, что позволяет системе корректно управлять его правами и персонализировать взаимодействие в соответствии с ролью и параметрами учетной записи.

3.4. Модуль управления ролями и правами

Модуль отвечает за управление правами доступа в системе на основе ролевой модели RBAC (Role-Based Access Control). Он обеспечивает

гибкую настройку прав, их привязку к ролям и управление этими ролями для пользователей системы.

В рамках модуля можно определять различные типы прав, включая доступ на чтение, запись, удаление и выполнение. Администраторы имеют возможность создавать новые роли, изменять существующие и удалять неактуальные.

Кроме того, модуль включает набор predefined ролей с заранее настроенными правами доступа, что упрощает администрирование и ускоряет настройку системы для типовых сценариев использования.

Таким образом, модуль реализует полный цикл управления ролями и правами доступа, обеспечивая гибкость и адаптивность системы к изменяющимся требованиям.

3.5. Модуль интеграции с LDAP

Модуль интеграции с LDAP отвечает за аутентификацию и авторизацию пользователей через внешнюю службу каталогов, такую как OpenLDAP или Active Directory. При входе в систему учетные данные пользователя проверяются через LDAP, после чего система определяет его уровень доступа в соответствии с заданными настройками безопасности.

Если пользователь ранее не входил в систему через LDAP, его учетные данные, полученные из службы каталогов, используются для автоматического создания пользователя в системе. Это позволяет избежать ручной регистрации и упрощает процесс управления пользователями.

При каждом входе в систему через LDAP данные учетной записи пользователя синхронизируются с актуальными данными из службы каталогов. Это гарантирует, что информация о пользователе в системе всегда соответствует данным в OpenLDAP или Active Directory.

Если учетная запись пользователя в службе каталогов заблокирована или срок ее действия истек, система не позволит пользователю войти. Это обеспечивает дополнительный уровень безопасности и предотвращает доступ неактивных или заблокированных пользователей.

Подключение к OpenLDAP или Active Directory осуществляется с использованием защищенного протокола LDAPS, обеспечивающего безопасный обмен данными между системой и службой каталогов.

Переключение на аутентификацию через LDAP, а также выбор используемой службы каталогов (OpenLDAP или Active Directory), настраивается в модуле управления приложением. Администратор может задать параметры подключения, определить политику аутентификации и настроить правила синхронизации учетных записей в соответствии с требованиями организации.

3.6. Модуль нотификации

Модуль нотификаций отвечает за доставку сообщений пользователям как внутри системы, так и по электронной почте. Пользователи могут самостоятельно настраивать параметры e-mail-уведомлений в своем профиле.

Администраторы обладают возможностью управлять правилами отправки уведомлений, а также создавать и настраивать шаблоны сообщений для различных типов событий.

3.7. Модуль информации о событиях

Модуль автоматически отслеживает и регистрирует все значимые события в системе, включая события, связанные с информационной безопасностью. Каждое событие фиксируется в Журнале с указанием

детальной информации: временной метки, описания действия (например, «пользователь зашел под аккаунтом <...>»), данных о пользователе, инициировавшем это действие, критичности события, и т.д.

3.8. Менеджер лицензий

Менеджер лицензий предназначен для управления лицензионными ключами и их привязки к конкретному серверу. При установке системы формируется уникальный идентификатор сервера, основанный на его аппаратных характеристиках (например, серийном номере материнской платы и идентификаторе процессора). Этот идентификатор используется для закрепления лицензионного ключа за конкретным сервером, что предотвращает использование одного ключа на нескольких инсталляциях системы.

Лицензионный ключ содержит сведения о лицензии, включая название организации, тип и срок действия лицензии, а также ограничения по числу пользователей и проектов. Администраторы могут просматривать всю связанную информацию по каждой лицензии.

Также администраторам предоставляется возможность добавлять новые лицензионные сертификаты. Перед сохранением нового сертификата или просмотром данных активной лицензии выполняется процесс верификации, в ходе которого проверяется соответствие сертификата установленному формату.

Процесс верификации включает следующие этапы:

- проверку подлинности подписи зашифрованных данных в соответствии с алгоритмом подписи;
- расшифровку данных с использованием указанного алгоритма шифрования;
- проверку совместимости параметров лицензии.

Кроме того, система автоматически контролирует соблюдение лицензионных ограничений перед выполнением определенных операций:

- при активации или реактивации аккаунта проверяется допустимое количество учетных записей;
- при авторизации пользователя – соответствие ограничениям по числу активных пользователей;
- при создании нового проекта – наличие доступных лицензий на проекты.

Также выполняется проверка срока действия лицензии, что обеспечивает использование только актуальных лицензий.

Таким образом, менеджер лицензий обеспечивает централизованное и безопасное управление лицензионными ключами, предотвращает их несанкционированное использование и контролирует соблюдение заданных ограничений.

3.9. Файловый менеджер

Модуль реализует функционал для загрузки, хранения и управления файлами, поддерживая как локальные, так и внешние файловые хранилища. Это обеспечивает гибкую организацию работы с данными в зависимости от потребностей системы.

Пользователи могут загружать и скачивать файлы из хранилища, что упрощает доступ к сохраненным данным и способствует эффективному обмену информацией между модулями системы.

Модуль поддерживает широкий спектр форматов файлов (CSV, XLSX, PNG, JPEG, TIFF, WEBP, ZIP и др.). Перед загрузкой выполняется проверка соответствия файлов допустимым расширениям, что позволяет предотвратить ошибки и повысить уровень безопасности системы. Такая

проверка гарантирует, что загружаемые данные соответствуют установленным требованиям.

3.10. Модуль импорта и экспорта данных

Модуль импорта и экспорта данных отвечает за организацию и управление процессами обмена информацией между системой и внешними источниками.

Ключевые функции модуля включают:

- прием входных файлов или потоков данных;
- асинхронную обработку и загрузку данных (импорт);
- формирование и сохранение выходных данных в заданных форматах (экспорт).

Работа модуля обеспечивает надежность и целостность передаваемой информации, ее согласованность и готовность к дальнейшей обработке внутри системы.

3.11. Модуль управления группами и проектами

Модуль предназначен для управления группами и проектами, обеспечивая удобную организацию и администрирование рабочих пространств. Группы формируют иерархическую структуру, объединяющую связанные проекты, что позволяет централизованно управлять доступом участников и создавать сервисные токены. Проекты представляют собой полноценные среды для моделирования бизнес-процессов, в рамках которых можно работать с таблицами, вычислителями, формами параметров, расчетами и отчетами. Такой подход делает модуль универсальным инструментом для анализа данных, планирования и поддержки принятия решений.

Модуль предоставляет широкий функционал. Пользователи могут создавать и редактировать группы и проекты, изменять их метаданные (название, описание), а также удалять ненужные элементы. Управление участниками включает добавление и удаление пользователей, назначение ролей и определение уровней доступа.

Кроме того, модуль координирует создание и удаление различных сущностей – папок, таблиц, файлов, вычислителей, форм параметров, расчетов, шаблонов дашбордов и сервисных токенов. Он направляет соответствующие запросы в специализированные модули, ответственные за работу с этими объектами. После выполнения операции модуль связывает созданную сущность с проектом либо отвязывает ее при удалении, обеспечивая целостность и согласованность структуры проекта.

Дополнительно модуль поддерживает экспорт и импорт проектов, что облегчает перенос данных между системами и создание резервных копий.

3.12. Модуль управления входными данными

Модуль предназначен для управления входными данными и предоставляет удобный интерфейс для их организации и обработки. Директории используются для группировки таблиц и файлов в виде древовидной структуры, что позволяет логично упорядочивать информацию и поддерживать иерархию данных. Таблицы служат основным инструментом хранения и представления информации, обеспечивая структурированный и наглядный способ работы с данными.

Главная задача таблиц – упростить процесс создания и управления таблицами для пользователей, не обладающих опытом работы с базами данных. Благодаря интуитивному интерфейсу, можно легко организовывать и хранить информацию без необходимости изучения сложных технических механизмов.

Пользователи могут создавать новые таблицы и изменять их структуру в соответствии с потребностями. В рамках редактирования таблицы доступно:

1. добавление и удаление столбцов с указанием их характеристик:
 - Название – понятное обозначение столбца;
 - Тип данных – формат хранимой информации (текст, число, дата и т.д.);
 - Уникальность – запрет на дублирование значений;
 - Обязательность – требование обязательного заполнения поля;
 - Порядок в таблице – позиция столбца относительно других;
2. настройка ограничений для проверки данных, включая:
 - левую часть условия (столбец, к которому применяется ограничение);
 - оператор сравнения (например, больше или равно, меньше или равно);
 - правую часть условия (значение или поле для сравнения);
3. установление связей между таблицами, аналогичных внешним ключам (foreign key), с указанием:
 - родительской таблицы и поля-источника;
 - дочерней таблицы и связанного поля;
 - типа связи (один к одному, один ко многим).

Для поддержания актуальности данных модуль позволяет редактировать, добавлять и удалять записи в таблицах. Все изменения проходят проверку на соответствие установленным ограничениям и связям, что гарантирует целостность данных. Например, при нарушении ограничений система выдаёт предупреждение, а при удалении записей выполняется проверка зависимостей, чтобы избежать потери связанных данных.

Модуль также поддерживает импорт данных из файлов форматов CSV и XLSX, что значительно упрощает процесс заполнения таблиц. Пользователь может загрузить данные из внешних источников, например, из электронных таблиц, и система автоматически преобразует их в структуру таблицы, проверяя соответствие типов данных и структуры столбцов.

Дополнительно модуль позволяет загружать файлы любых форматов в качестве входных данных. Доступные типы файлов определяются администратором в панели администрирования.

3.13. Модуль управления вычислителями

Модуль предназначен для управления вычислителями, обеспечивая их создание, редактирование, удаление, хранение и получение данных. При добавлении или изменении вычислителя пользователь может указать его название, описание и загрузить ZIP-архив, содержащий скрипт.

Архив должен включать:

- исполняемый файл `main.py`, содержащий логику выполнения скрипта;
- файл `requirements.txt` со списком зависимостей, устанавливаемых через `pip`.

Для корректного функционирования вычислителя скрипт должен быть совместим с поддерживаемыми версиями Python.

3.14. Модуль управления формами параметров

Модуль управления формами предназначен для создания, редактирования, удаления и хранения форм, а также для получения информации о них. Формы выполняют роль шаблонов, определяющих

набор параметров, необходимых для выполнения расчетов. Каждая форма может быть индивидуально настроена под конкретные задачи, что делает процесс подготовки входных данных удобным, стандартизированным и контролируемым.

При создании или редактировании формы пользователь может задать ее название и описание, настроить параметры и их ограничения, а также объединить параметры в группы. Параметры являются ключевыми элементами формы, определяющими структуру входных данных. Для каждого параметра можно указать:

- название;
- тип данных;
- ключ;
- значение по умолчанию;
- принадлежность к группе;
- обязательность заполнения;
- множественность значений.

Также поддерживается задание ограничений, которые проверяются при вводе данных. Например, можно установить условие, что значение параметра должно быть больше нуля. Ограничение формируется путем указания левого операнда (параметра), оператора сравнения ("больше", "меньше", "равно" и др.) и правого операнда (значения).

Каждая форма включает обязательные дефолтные параметры, которые не могут быть удалены. К ним относятся:

- Вычислитель – определяет вычислитель, используемый для обработки данных;
- Входные данные – задает файлы и таблицы, применяемые в качестве исходных данных;
- Время моделирования – задает интервал моделирования (в секундах);

- Дашборд – определяет панель визуализации результатов расчетов;
- Мнемосхема – задает схему визуализации результатов;
- Результаты другого расчета – позволяет использовать результаты ранее выполненного расчета в качестве входных данных;
- Сохранение структуры входных данных – указывает необходимость сохранения структуры загруженных данных.

Модуль поддерживает группировку параметров, что особенно полезно при большом количестве элементов. Группы помогают структурировать форму, визуально разделяя параметры по вкладкам. Каждая группа имеет название, описание, уникальный ключ и список связанных параметров.

После настройки формы она используется для подготовки входных данных. При создании расчета пользователь выбирает форму, на основе которой система автоматически генерирует интерфейс для ввода параметров. Все введенные значения проходят проверку на соответствие установленным характеристикам, что обеспечивает корректность и целостность данных.

Таким образом, модуль управления формами стандартизирует процесс подготовки входных данных, повышает точность расчетов и снижает риск ошибок. Благодаря гибкой настройке параметров и встроенным механизмам проверки он обеспечивает удобство, надежность и адаптивность при работе с расчетными моделями.

3.15. Модуль управления расчетами

Модуль управления расчетами обеспечивает создание, редактирование, удаление, запуск и остановку расчетов. Он отслеживает их

состояние на всех этапах жизненного цикла, динамически изменяя статус в зависимости от хода выполнения и действий пользователей. Это позволяет в реальном времени контролировать процесс выполнения расчетов.

После создания расчет получает начальный статус «Создан», что подтверждает успешную инициализацию. В дальнейшем он проходит ряд промежуточных статусов, отражающих текущую стадию обработки:

- Подготовка входных данных – сбор и настройка необходимых данных;
- Готов к запуску – данные подготовлены, расчет ожидает начала выполнения;
- В очереди – расчет добавлен в очередь задач, ожидающих обработки;
- Ожидает скачивания – подготовка к загрузке файлов и вспомогательных данных;
- Скачивание – активная загрузка необходимых данных;
- В работе – расчет выполняется, система обрабатывает заданные параметры.

Финальные статусы определяются результатом выполнения или действиями пользователей:

- Успешно – расчет завершен корректно, результаты готовы к использованию;
- Ошибка – возникла проблема при обработке данных или выполнении задачи;
- Отменен сервером – процесс остановлен сервером по техническим причинам;
- Отменен хостом – расчет прерван пользователем или инициатором запуска;
- Время истекло – выполнение остановлено из-за превышения допустимого времени моделирования;

- Архивирован – расчет удален пользователем.

Изменение статусов происходит автоматически, в зависимости от действий пользователя (запуск, остановка, отмена) или системных событий (успешное завершение, ошибка, тайм-аут). Это обеспечивает прозрачность и точное отслеживание состояния расчетов на каждом этапе их выполнения.

3.16. Модуль управления визуализацией

Модуль управления визуализацией предназначен для создания, редактирования, удаления и использования дашбордов и мнемосхем, обеспечивающих наглядное представление и анализ данных расчетов. Он формирует связующее звено между вычислителями, расчетами и визуальными элементами, создавая единый контур работы с результатами моделирования.

Дашборды предназначены для отображения результатов расчетов в виде интерактивных панелей с настраиваемыми виджетами. Пользователь может задать название дашборда, выбрать связанный вычислитель, определить дашборд по умолчанию, а также настраивать состав и внешний вид элементов визуализации. Редактирование включает добавление и удаление виджетов, изменение их расположения, параметров и формата отображения в зависимости от типа данных.

Мнемосхемы используются для представления структуры модели и взаимосвязей между ее компонентами с помощью графических элементов и соединений. Пользователь может добавлять стандартные или пользовательские элементы из библиотеки, задавать связи между ними и привязывать к ним данные расчетов. Редактирование осуществляется в интерактивном режиме (drag-and-drop), с возможностью изменения размеров, положения и логики взаимодействия элементов.

Модуль также предоставляет инструменты для сравнения результатов расчетов. Сопоставление может выполняться как на уровне содержимого файлов – для детального анализа различий в данных, – так и с использованием настраиваемых графиков сравнения, позволяющих визуально анализировать расхождения между результатами разных расчетов.

Благодаря гибкой настройке визуализации и средствам интерактивного взаимодействия, модуль предоставляет пользователю удобные инструменты для анализа, интерпретации и сравнения результатов расчетов в наглядной форме.

3.17. Модуль локализации

Модуль локализации предназначен для управления переводами системных сообщений и настройки шаблонов электронных уведомлений.

Он обеспечивает хранение и выбор текстовых ресурсов в зависимости от выбранного языка интерфейса, а также редактирование почтовых шаблонов с использованием переменных и средств форматирования.

Функциональность модуля позволяет поддерживать корректное отображение интерфейса на русском и английском языках, а также изменять содержание уведомлений без необходимости внесения изменений в программный код, что упрощает адаптацию системы под разные языковые и корпоративные стандарты.

3.18. Библиотека общей функциональности

Модуль предназначен для предоставления базовых классов и вспомогательных компонентов, используемых другими модулями системы.

Основные функции модуля заключаются в реализации унифицированных классов, утилит, констант и методов, обеспечивающих

согласованную работу всех компонентов системы, повышение переиспользуемости кода и единообразия архитектуры.

3.19. Модуль управления приложением

Модуль управления приложением отвечает за инициализацию и настройку ключевых параметров системы. Он обеспечивает конфигурирование режима работы, подключение к базам данных, настройку сервисов аутентификации, параметров безопасности и других системных характеристик, что позволяет гибко управлять основными аспектами функционирования приложения.

Помимо этого, модуль регулирует визуальное оформление приложения и работу с внешними ссылками. Пользователь может добавлять, редактировать и удалять ссылки, загружать логотипы для страницы авторизации и интерфейса приложения, а также изменять цветовую палитру для адаптации внешнего вида к корпоративным требованиям.

Модуль также обеспечивает взаимодействие как с внутренними компонентами системы, так и с внешними сервисами, расширяя возможности приложения и упрощая интеграцию с внешними источниками данных.

Дополнительно модуль управляет процессами обработки и выполнения расчетов – предоставляет задачи вычислительному сервису, выполняет выгрузку входных данных, синхронизирует статусы расчетов и загружает результаты обратно в систему, поддерживая непрерывный цикл вычислений и анализа.

3.20. Вспомогательные модули

В данном разделе представлены вспомогательные модули:

- Модуль buildSrc – вспомогательный Gradle-проект с необходимыми для сборки скриптами, конфигурациями и плагинами.
- Модуль gradle – директория для хранения спецификаций версий зависимостей проекта и конфигурации сервиса обработки вычислений.

Эти модули обеспечивают стабильную работу системы, упрощая управление зависимостями и сборкой приложения.

3.21. Модуль пользовательского интерфейса

Модуль реализует удобную и интуитивно понятную систему навигации по основным разделам и функциям приложения, позволяя пользователям быстро находить нужные элементы и эффективно взаимодействовать с системой. Четко структурированный интерфейс обеспечивает логичную организацию навигации и снижает порог вхождения для новых пользователей.

В основе модуля лежит использование стандартных элементов интерфейса, что делает работу предсказуемой и комфортной. Благодаря этому пользователи могут быстро освоить функционал без дополнительного обучения.

Система обеспечивает мгновенную обратную связь на каждое действие пользователя – будь то нажатие кнопки, ввод данных или выполнение операции. При успешных действиях отображаются визуальные уведомления, например сообщения о сохранении данных или завершении операции, что повышает прозрачность работы приложения и уверенность пользователя в корректности своих действий.

Интерфейс построен по модульному принципу: каждый компонент функционирует независимо, что позволяет легко модифицировать, дополнять или удалять отдельные элементы без затрагивания остальных

частей системы. Такая архитектура делает интерфейс гибким, расширяемым и удобным для адаптации под различные сценарии использования.

Кроме того, модуль поддерживает переиспользование интерфейсных компонентов в разных частях приложения, что ускоряет разработку, упрощает сопровождение и обеспечивает единообразие визуального стиля и поведения элементов.

3.22. Сервис обработки задач

Сервис обработки задач представляет собой распределенную систему, включающую серверную и клиентскую части. Его основная задача — автоматизация процессов поиска, выполнения и управления вычислительными задачами, обеспечивая надежную и масштабируемую инфраструктуру для распределённых вычислений.

Серверная часть является ядром системы и отвечает за непосредственное выполнение задач. Она функционирует автономно, может быть развернута в нескольких экземплярах и не требует постоянного взаимодействия с клиентской частью.

Основные функции серверной части:

- Поиск задач – автоматическое определение и получение задач, доступных для обработки.
- Загрузка данных – скачивание необходимых файлов и параметров из веб-приложения.
- Выполнение вычислений – запуск и обработка задач с использованием встроенных вычислительных модулей.
- Передача результатов – выгрузка полученных данных обратно в веб-приложение.
- Оповещение о прогрессе – передача информации о ходе выполнения задачи в режиме реального времени.

Особенности работы серверной части:

- Поддержка нескольких экземпляров серверов, развернутых в разных локациях.
- Автономный режим работы, не требующий постоянного подключения к клиентскому интерфейсу.
- Возможность добавления нескольких исполнителей, обеспечивая параллельное выполнение задач.

Клиентская часть представляет собой инструмент управления и мониторинга серверов. Она обеспечивает полный контроль над процессом вычислений, позволяя пользователю наблюдать за состоянием системы, управлять ресурсами и оптимизировать распределение задач.

Основные функции клиентской части:

- Подключение к нескольким серверам – возможность централизованного управления всеми вычислительными узлами независимо от их местоположения.
- Мониторинг вычислителей и задач – отображение состояния серверов, прогресса выполнения и статусов активных задач.
- Добавление и настройка вычислителей – регистрация новых вычислительных узлов и конфигурация их параметров.
- Остановка задач – возможность принудительно завершить выполнение задачи при необходимости.
- Безопасное завершение работы серверов – корректное завершение всех процессов с сохранением данных и предотвращением потери результатов.

Таким образом, сервис обработки задач обеспечивает масштабируемое, гибкое и контролируемое выполнение вычислений, предоставляя пользователям эффективные средства управления распределенными ресурсами.

4. ПОЛЬЗОВАТЕЛЬСКИЕ СЦЕНАРИИ

Самостоятельная регистрация пользователя

Краткое описание сценария	Самостоятельная регистрация пользователя в системе
Участники	Пользователь Администратор Система
Предусловие (контекст)	Пользователь не зарегистрирован в системе. Система предоставляет возможность для самостоятельной регистрации. Администратор может настроить формат адреса электронной почты.
Постусловие (результат)	Пользователь зарегистрирован в системе.
Основной сценарий	<ol style="list-style-type: none">1. Пользователь заходит на главную страницу.2. Пользователь нажимает кнопку “Зарегистрироваться”.3. Пользователь вводит имя, фамилию и e-mail и нажимает кнопку “Создать аккаунт”.4. Система проверяет наличие шаблона e-mail, установленного администратором:<ol style="list-style-type: none">4а. Если шаблон не установлен или e-mail соответствует шаблону, то переходим к п.6.4б. Если e-mail не соответствует шаблону, то регистрация требует подтверждения администратора, переходим к п.5.5. Администратор подтверждает регистрацию.

	<p>6. Система отправляет письмо со ссылкой для активации на почту пользователя.</p> <p>7. Пользователь переходит по ссылке из письма и перенаправляется на страницу для создания пароля.</p> <p>8. Пользователь вводит новый пароль, подтверждает его и нажимает кнопку “Создать”.</p> <p>9. Система завершает процесс регистрации и активирует аккаунт пользователя.</p> <p>10. Пользователь получает возможность входа в систему.</p>
<p>Альтернативный сценарий</p>	<p><i>A1. Неверный формат e-mail</i></p> <p>1. Система отображает сообщение об ошибке и просит пользователя ввести корректный e-mail.</p> <p><i>A2. Пользователь с указанным e-mail уже существует</i></p> <p>1. Система отображает сообщение об ошибке и просит пользователя ввести другой e-mail.</p> <p><i>A3. Администратор отклоняет регистрацию пользователя</i></p> <p>1. Система не отправляет пользователю код активации.</p> <p><i>A4. Пароль пользователя не отвечает требованиям политики</i></p> <p>1. Система отображает сообщение об ошибке, указывая конкретный критерий безопасности, которому не соответствует пароль.</p>
<p>Исключения</p>	<p>Сбой в работе системы, отсутствие доступа к панели администратора и/или почтовому серверу.</p>

Регистрация пользователя через администратора

Краткое описание сценария	Регистрация пользователя через администратора
Участники	Пользователь Администратор Система
Предусловие (контекст)	1. Администратор авторизован в системе. 2. Пользователь не зарегистрирован в системе.
Постусловие (результат)	Пользователь зарегистрирован в системе.
Основной сценарий	<ol style="list-style-type: none">1. Администратор переходит в раздел “Администрирование” на вкладку “Пользователи”.2. Администратор нажимает кнопку “Создать аккаунт”.3. Администратор вводит имя, фамилию, e-mail пользователя и нажимает кнопку “Создать аккаунт”.4. Система отправляет письмо со ссылкой для активации на почту пользователя.5. Пользователь переходит по ссылке из письма и перенаправляется на страницу для создания пароля.6. Пользователь вводит новый пароль, подтверждает его и нажимает кнопку “Создать”.7. Система завершает процесс регистрации и активирует аккаунт пользователя.8. Пользователь получает возможность входа в систему.
Альтернативный сценарий	<i>A1. Неверный формат e-mail</i>

	<p>1. Система отображает сообщение об ошибке и просит администратора ввести корректный e-mail.</p> <p><i>A2. Пользователь с указанным e-mail уже существует</i></p> <p>1. Система отображает сообщение об ошибке и просит администратора ввести другой e-mail.</p> <p><i>A4. Пароль пользователя не отвечает требованиям политики</i></p> <p>1. Система отображает сообщение об ошибке, указывая конкретный критерий безопасности, которому не соответствует пароль.</p>
Исключения	Сбой в работе системы, отсутствие доступа к панели администратора и/или почтовому серверу.

Вход пользователя в систему через внутреннюю учетную запись

Краткое описание сценария	Вход пользователя в систему через внутреннюю учетную запись
Участники	Пользователь Система
Предусловие (контекст)	Пользователь зарегистрирован в системе.
Постусловие (результат)	Пользователь получил доступ к системе.
Основной сценарий	<p>1. Пользователь заходит на главную страницу.</p> <p>2. Пользователь вводит логин, пароль, (при необходимости активирует флажок “Запомнить меня”) и нажимает кнопку “Войти”.</p>

	<p>3. Система идентифицирует пользователя, проверяя, имеется ли в локальной базе данных пользователь с указанным логином.</p> <p>4. Система аутентифицирует пользователя, проверяя правильность введенного пароля.</p> <p>5. Система авторизует пользователя, предоставляя ему доступ к функционалу и ресурсам, соответствующим его роли и правам, установленным в системе.</p> <p>6. Система отображает вкладку "Проекты" с деревом групп и проектов (если они были созданы ранее).</p>
<p>Альтернативный сценарий</p>	<p><i>A1. Пользователя с указанным логином не существует</i></p> <p>1. Система отображает сообщение об ошибке и просит пользователя ввести корректные данные.</p> <p>2. Пользователь исправляет ошибку и повторяет попытку входа.</p> <p><i>A2. Указанный пользователем пароль не совпадает</i></p> <p>1. Система отображает сообщение об ошибке и просит пользователя ввести корректные данные.</p> <p>2. Пользователь исправляет ошибку и повторяет попытку входа.</p> <p><i>A3. Пользователь заблокирован</i></p> <p>1. Система отображает сообщение о блокировке пользователя.</p> <p><i>A4. Истек срок действия пароля</i></p> <p>1. Система отображает сообщение об ошибке и рекомендует сменить пароль.</p>

Исключения	Сбой в работе системы.
------------	------------------------

Вход пользователя в систему через LDAP

Краткое описание сценария	Вход пользователя в систему через LDAP
Участники	Пользователь Система Служба каталогов (OpenLDAP или Active Directory)
Предусловие (контекст)	Пользователь зарегистрирован в службе каталогов.
Постусловие (результат)	Пользователь получил доступ к системе.
Основной сценарий	<ol style="list-style-type: none"> 1. Пользователь заходит на главную страницу. 2. Пользователь вводит логин, пароль, и нажимает на кнопку “Войти”. 3. Система отправляет полученные данные на проверку в службу каталогов. 4. Служба каталогов проверяет учетные данные. 5. Служба каталогов возвращает информацию о пользователе. 6. Система соотносит роли пользователя, полученные из службы каталогов, с внутренними правами доступа. 7. Система авторизует пользователя, предоставляя ему доступ к функционалу и ресурсам, соответствующим его правам.

	8. Система отображает вкладку "Проекты" с деревом групп и проектов (если они были созданы ранее).
Альтернативный сценарий	<p><i>A1. Служба каталогов вернула ошибку аутентификации</i></p> <p>1. Система отображает сообщение об ошибке.</p> <p><i>A2. Пользователь заблокирован в службе каталогов</i></p> <p>1. Система отображает сообщение о блокировке пользователя.</p> <p><i>A3. Истек срок действия пароля в службе каталогов</i></p> <p>1. Система отображает сообщение об ошибке и рекомендует сменить пароль.</p>
Исключения	Сбой в работе системы, отсутствие доступа к службе каталога.

Создание проекта

Краткое описание сценария	Предоставить пользователю возможность создавать проекты
Участники	Пользователь Система
Предусловие (контекст)	Пользователь авторизован в системе. Пользователь находится на вкладке «Проекты».
Постусловие (результат)	Пользователь успешно создал проект.
Основной сценарий	1. Пользователь нажимает на кнопку «+» в левом верхнем углу.

2. В выпадающем списке пользователь нажимает на кнопку «Проект».

3. В модальном окне пользователь указывает название и описание проекта, а также выбирает родительскую группу или корневой уровень, к которому будет относиться создаваемый проект.

4. После заполнения всех необходимых полей пользователь нажимает кнопку «Сохранить».

5. Система создает проект на основе введенных данных.

6. Система автоматически добавляет создателя проекта в список участников, присваивая ему дефолтную роль Owner.

7. Система отображает пользователю сообщение «Успешно создано».

8. Система отображает проект в дереве на уровне вложенности, заданном при создании, с указанным названием.

9. Пользователь нажимает на название проекта.

10. Система отображает пользователю страницу проекта, включая:

10.1. Название проекта.

10.2. Описание проекта.

10.3. Панель инструментов для управление проектом (изменение метаданных, экспорт и удаление проекта).

10.3. Панель навигации с доступными разделами (Входные данные, Вычислители,

	Формы, Расчеты, Дашборды, Мнемосхемы, Настройки, Участники, Экспорт, События).
Альтернативный сценарий	<p><i>4А. Пользователь не указал название проекта</i></p> <p>1. Система заблокирует возможность создания проекта.</p> <p>2. Система подсвечивает незаполненное поле красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.</p> <p><i>4Б. Пользователь не выбирает родительскую группу или корневой уровень.</i></p> <p>1. Система автоматически присваивает проекту корневой уровень.</p> <p><i>4В. Пользователь отменяет создание проекта до сохранения:</i></p> <p>1. Система отменяет создание проекта и не сохраняет введенные данные.</p>
Исключения	Сбой в работе системы.

Создание таблицы

Краткое описание сценария	Предоставить пользователю возможность создавать таблицы в проекте
Участники	Пользователь Система
Предусловие (контекст)	Пользователь авторизован в системе. Пользователь создал проект.

	Пользователь находится в разделе «Входные данные».
Постусловие (результат)	Пользователь успешно создал таблицу.
Основной сценарий	<ol style="list-style-type: none"> 1. Пользователь нажимает на кнопку «+» в левом верхнем углу. 2. В выпадающем списке пользователь нажимает на кнопку «Таблица». 3. В модальном окне пользователь указывает название таблицы, а также выбирает родительскую директорию или корневой уровень, к которому будет относиться создаваемая таблица. 4. После заполнения всех необходимых полей пользователь нажимает кнопку «Сохранить». 5. Система создает таблицу на основе введенных данных. 6. Система отображает пользователю сообщение «Успешно создано». 7. Система отображает таблицу в дереве на уровне вложенности, заданном при создании, с указанным названием. 8. Система направляет пользователя в раздел «Структура» для задания структуры созданной таблицы.
Альтернативный сценарий	<p><i>4А. Пользователь не указал название таблицы</i></p> <ol style="list-style-type: none"> 1. Система блокирует возможность создания таблицы.

	<p>2. Система подсвечивает незаполненное поле красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.</p> <p><i>4Б. Пользователь указывает название таблицы, которая уже существует на выбранном уровне дерева</i></p> <p>1. Система заблокирует возможность создания таблицы.</p> <p>2. Система подсвечивает поле «Название» красным цветом и отображает сообщение об ошибке «Объект с таким именем уже существует на этом уровне» при наведении на это поле.</p> <p><i>4В. Пользователь не выбирает родительскую директорию или корневой уровень.</i></p> <p>1. Система автоматически привязывает таблицу к корневому уровню.</p> <p><i>4Г. Пользователь отменяет создание таблицы до сохранения:</i></p> <p>1. Система отменяет создание таблицы и не сохраняет введенные данные.</p>
Исключения	Сбой в работе системы.

Настройка структуры таблицы

Краткое описание сценария	Предоставить пользователю возможность задавать структуру таблицы
---------------------------	--

Участники	<p>Пользователь</p> <p>Система</p>
Предусловие (контекст)	<p>Пользователь авторизован в системе.</p> <p>Пользователь создал проект.</p> <p>Пользователь находится на вкладке «Входные данные».</p> <p>Пользователь создал таблицу.</p> <p>Пользователь находится в разделе «Структура» созданной таблицы.</p>
Постусловие (результат)	<p>Пользователь успешно задал структуру таблицы.</p>
Основной сценарий	<ol style="list-style-type: none"> 1. Система отображает пользователю перечень столбцов с описанием их характеристик (название, тип данных, и т.д.). 2. Пользователь нажимает на кнопку «Добавить столбец таблицы» в шапке таблицы. 3. Система отображает пользователю модальное окно создания столбца таблицы. 4. Пользователь задает: <ol style="list-style-type: none"> 4.1. Название столбца 4.2. Тип данных столбца 4.3. Признак уникальности столбца 4.4. Признак обязательности столбца 4.5. Необходимость закрепления столбца 5. После заполнения всех необходимых полей пользователь нажимает кнопку «Сохранить». 6. Система создает столбец на основе введенных данных.

	<p>7. Система отображает пользователю сообщение «Успешно создано».</p> <p>8. Созданный столбец отображается в перечне столбцов.</p>
Альтернативный сценарий	<p><i>5А. Пользователь не заполнил обязательные поля</i></p> <p>1. Система заблокирует возможность создания столбца.</p> <p>2. Система подсвечивает незаполненное поле красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.</p> <p><i>5Б. Пользователь указывает название столбца, который уже существует в таблице</i></p> <p>1. Система заблокирует возможность создания столбца.</p> <p>2. Система подсвечивает поле «Название» красным цветом и отображает сообщение об ошибке «В таблице уже существует столбец с таким названием» при наведении на это поле.</p> <p><i>5В. Пользователь отменяет создание столбца до сохранения:</i></p> <p>1. Система отменяет создание столбца и не сохраняет введенные данные.</p>
Исключения	Сбой в работе системы.

Наложение ограничений на столбцы таблицы

Краткое описание сценария	Предоставить пользователю возможность накладывать ограничения на столбцы таблицы
Участники	Пользователь Система
Предусловие (контекст)	Пользователь авторизован в системе. Пользователь создал проект. Пользователь находится на вкладке «Входные данные». Пользователь создал таблицу. Пользователь создал хотя бы один столбец.
Постусловие (результат)	Пользователь успешно создал ограничение на столбец таблицы.
Основной сценарий	<ol style="list-style-type: none"> 1. Пользователь переходит на вкладку «Ограничения». 2. Система отображает пользователю перечень ограничений с описанием их характеристик (левая часть оператора, операнд, правая часть оператора). 3. Пользователь нажимает на кнопку «Добавить ограничение» в шапке таблицы. 4. Система отображает пользователю модальное окно создания ограничения. 5. Пользователь задает: <ol style="list-style-type: none"> 5.1. Левую часть оператора 5.2. Операнд 5.3. Правую часть оператора 6. После заполнения всех необходимых полей пользователь нажимает кнопку «Сохранить».

	<p>7. Система создает ограничение на основе введенных данных.</p> <p>7. Система отображает пользователю сообщение «Успешно создано».</p> <p>8. Созданное ограничение отображается в перечне ограничений.</p>
<p>Альтернативный сценарий</p>	<p><i>6А. Пользователь не заполнил обязательные поля</i></p> <p>1. Система заблокирует возможность создания ограничения.</p> <p>2. Система подсвечивает незаполненное поле красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.</p> <p><i>6Б. Ограничение, заданное пользователем, противоречит данным в таблице</i></p> <p>1. Система заблокирует возможность создания ограничения.</p> <p>2. Система отображает сообщение об ошибке «Не удалось создать ограничение в таблице».</p> <p><i>6Б. Ограничение, заданное пользователем, противоречит другому ограничению</i></p> <p>1. Система заблокирует возможность создания ограничения.</p> <p>2. Система отображает сообщение об ошибке «Не удалось создать ограничение в таблице».</p>

	<p><i>бВ. Пользователь отменяет создание ограничения до сохранения:</i></p> <p>1. Система отменяет создание ограничения и не сохраняет введенные данные.</p>
Исключения	Сбой в работе системы.

Создание связей между таблицами

Краткое описание сценария	Предоставить пользователю возможность создавать логическую связь (foreign key) между столбцами двух таблиц
Участники	Пользователь Система
Предусловие (контекст)	<p>Пользователь авторизован в системе.</p> <p>Пользователь создал проект.</p> <p>Пользователь находится на вкладке «Входные данные».</p> <p>Пользователь создал 2 таблицы (таблица-источник и дочерняя таблица).</p> <p>Пользователь задал структуру таблиц (создал хотя бы один столбец и, если необходимо, наложил ограничения).</p>
Постусловие (результат)	Пользователь успешно создал связь между таблицами.
Основной сценарий	<ol style="list-style-type: none"> 1. Пользователь переходит на страницу дочерней таблицы. 2. Пользователь переходит в раздел «Связи». 3. Система отображает пользователю таблицу с созданными связями.

	<p>4. Пользователь нажимает на кнопку «Добавить связь» в шапке таблицы.</p> <p>5. Система отображает пользователю модальное окно добавления связи, где текущая таблица отображается в качестве дочерней таблицы.</p> <p>6. Пользователь:</p> <ul style="list-style-type: none"> • Выбирает родительскую таблицу. • Выбирает поле-источник из родительской таблицы. • Устанавливает тип связи. • Выбирает связанное поле из дочерней таблицы. <p>7. Пользователь нажимает кнопку «Сохранить».</p> <p>8. Система создает связь на основе введенных данных.</p> <p>9. Система отображает пользователю сообщение «Успешно создано».</p> <p>10. Созданная связь отображается в таблице.</p>
<p>Альтернативный сценарий</p>	<p><i>7А. Пользователь не заполнил обязательные поля.</i></p> <p>1. Система заблокирует возможность создания связи.</p> <p>2. Система подсвечивает незаполненное поле красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.</p> <p><i>7Б. Связываемые столбцы имеют разные типы данных</i></p> <p>1. Система заблокирует возможность создания связи.</p> <p>2. Система отобразит сообщение «Колонки таблиц имеют разные типы данных».</p>

	<p><i>7В. Столбец, указывающий на одну запись в связи, неуникальный</i></p> <ol style="list-style-type: none"> 1. Система заблокирует возможность создания связи. 2. Система отобразит сообщение «Колонка таблицы имеет не имеет уникальные значения». <p><i>7Г. Связь дублирует другую связь</i></p> <ol style="list-style-type: none"> 1. Система заблокирует возможность создания связи. 2. Система отобразит сообщение об ошибке. <p><i>7Д. Образуется кольцевая связь</i></p> <ol style="list-style-type: none"> 1. Система заблокирует возможность создания связи. 2. Система отобразит сообщение об ошибке. <p><i>7Е. Пользователь отменяет создание связи</i></p> <ol style="list-style-type: none"> 1. Система отменяет создание записи в таблице и не сохраняет введенные данные.
Исключения	Сбой в работе системы.

Добавление данных в таблицу

Краткое описание сценария	Предоставить пользователю возможность добавлять данные в таблицу
Участники	Пользователь Система
Предусловие (контекст)	Пользователь авторизован в системе. Пользователь создал проект. Пользователь находится на вкладке «Входные данные».

	<p>Пользователь создал таблицу.</p> <p>Пользователь задал структуру таблицы (создал хотя бы один столбец и, если необходимо, наложил ограничения).</p>
Постусловие (результат)	Пользователь успешно добавил данные в таблицу.
Основной сценарий	<ol style="list-style-type: none"> 1. Пользователь переходит в раздел «Данные». 2. Система отображает пользователю таблицу с заданными столбцами. 3. Пользователь нажимает на кнопку «Добавить данные» в шапке таблицы. 4. Система отображает пользователю модальное окно добавления данных, где отображается название ранее созданного столбца и предоставляется поле для ввода значения. 5. Пользователь заполняет поле и нажимает кнопку «Сохранить». 6. Система создает запись на основе введенных данных. 7. Система отображает пользователю сообщение «Успешно создано». 8. Созданная запись отображается в таблице.
Альтернативный сценарий	<p><i>5А. Пользователь не заполнил обязательные поля.</i></p> <ol style="list-style-type: none"> 1. Система блокирует возможность добавления данных. 2. Система подсвечивает незаполненное поле красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.

	<p><i>5Б. Введенное значение не соответствует ограничению</i></p> <ol style="list-style-type: none"> 1. Система заблокирует возможность добавления данных. 2. Система подсвечивает неправильно заполненное поле красным цветом и отображает сообщение о несоответствии ограничению при наведении на это поле. <p><i>5Б. Введенное значение не соответствует ограничению</i></p> <ol style="list-style-type: none"> 1. Система заблокирует возможность добавления данных. 2. Система подсвечивает неправильно заполненное поле красным цветом и отображает сообщение о несоответствии ограничению при наведении на это поле. <p><i>5В. Пользователь отменяет добавление данных</i></p> <ol style="list-style-type: none"> 1. Система отменяет создание записи в таблице и не сохраняет введенные данные.
Исключения	Сбой в работе системы.

Загрузка файлов

Краткое описание сценария	Предоставить пользователю возможность загружать файлы в качестве входных данных для расчетов
Участники	Пользователь

	Система
Предусловие (контекст)	<p>Пользователь авторизован в системе.</p> <p>Пользователь создал проект.</p> <p>Пользователь находится на вкладке «Входные данные».</p>
Постусловие (результат)	Пользователь успешно загрузил один или несколько файлов.
Основной сценарий	<ol style="list-style-type: none"> 1. Пользователь нажимает на кнопку «+» в левом верхнем углу. 2. В выпадающем списке пользователь нажимает на кнопку «Файлы». 3. В модальном окне пользователь выбирает один или несколько файлов для загрузки, а также выбирает родительскую директорию или корневой уровень, куда будут загружаться файл(ы). 4. После заполнения всех необходимых полей пользователь нажимает кнопку «Сохранить». 5. Система загружает и сохраняет в системе файл(ы). 6. Система отображает пользователю сообщение «Успешно создано». 7. Система отображает файл(ы) в дереве на уровне вложенности, заданном при создании.
Альтернативный сценарий	<p><i>4А. Пользователь не заполнил обязательные поля.</i></p> <ol style="list-style-type: none"> 1. Система блокирует возможность загрузки файлов 2. Система подсвечивает незаполненное поле красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.

	<p><i>4Б. Расширение файла не является допустимым</i></p> <ol style="list-style-type: none"> 1. Система заблокирует возможность загрузки файлов. 2. Система отображает сообщение об ошибке «Неподдерживаемое расширение для загрузки файла:...». <p><i>4В. Пользователь загружает файл, размер которого превышает допустимый лимит</i></p> <ol style="list-style-type: none"> 1. Система блокирует возможность загрузки файла. 2. Система отображает сообщение об ошибке «Превышен максимальный размер файла». <p><i>4Г. Название файла дублирует название существующего файла</i></p> <ol style="list-style-type: none"> 1. Система блокирует возможность загрузки файла. 2. Система отображает сообщение об ошибке «Контентный узел с названием:... уже существует среди узлов того же уровня». <p><i>4Д. Пользователь отменяет загрузку файлов.</i></p> <ol style="list-style-type: none"> 1. Система отменяет загрузку файлов.
Исключения	Сбой в работе системы.

Создание вычислителя

Краткое описание сценария	Предоставить пользователю возможность добавлять данные в таблицу
Участники	Пользователь Система
Предусловие (контекст)	Пользователь авторизован в системе. Пользователь создал проект. Пользователь создал и заполнил данными таблицу.
Постусловие (результат)	Пользователь успешно создал вычислитель.
Основной сценарий	<ol style="list-style-type: none"> 1. Пользователь переходит на вкладку «Вычислители». 2. Система отображает таблицу с перечнем вычислителей. 3. Пользователь нажимает на кнопку «Добавить» в шапке таблицы. 4. Система отображает пользователю модальное окно добавления вычислителя. 5. Пользователь задает название и описание вычислителя, а также загружает ZIP-архив со скриптом. 6. Пользователь нажимает кнопку «Сохранить». 7. Система создает вычислитель. 8. Система отображает пользователю сообщение «Успешно создано». 9. Созданный вычислитель отображается в таблице.
Альтернативный сценарий	<i>6А. Пользователь не заполнил обязательные поля</i>

1. Система заблокирует возможность создание вычислителя.

2. Система подсвечивает незаполненное поле красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.

бБ. Пользователь указывает название вычислителя, который уже существует в системе

1. Система заблокирует возможность создания вычислителя.

2. Система подсвечивает поле «Название» красным цветом и отображает сообщение об ошибке «Поле не должно содержать следующие значения:...» при наведении на это поле.

бВ. Пользователь загружает файл неправильного формата

1. Система заблокирует возможность создания вычислителя.

2. Система отображает сообщение об ошибке «Неподдерживаемое расширение для загрузки файла:...».

бГ. Пользователь загружает файл, размер которого превышает допустимый лимит

1. Система создает вычислитель, не прикрепляя к нему невалидный ZIP-архив.

	<p>2. Система отображает сообщение «Превышен максимальный размер файла».</p> <p><i>бД. Пользователь отменяет создание вычислителя</i></p> <p>1. Система отменяет создание вычислителя и не сохраняет введенные данные.</p>
Исключения	Сбой в работе системы.

Создание формы параметров

Краткое описание сценария	Предоставить пользователю возможность создавать форму параметров.
Участники	Пользователь Система
Предусловие (контекст)	Пользователь авторизован в системе. Пользователь создал проект. Пользователь находится на странице проекта.
Постусловие (результат)	Пользователь успешно создал форму параметров.
о	<ol style="list-style-type: none"> 1. Пользователь переходит на вкладку «Формы». 2. Система отображает таблицу с перечнем форм. 3. Пользователь нажимает на кнопку «Добавить» в шапке таблицы. 4. Система отображает пользователю модальное окно создания формы. 5. Пользователь задает название и описание формы и нажимает кнопку «Сохранить». 6. Система выводит страницу редактирования формы, содержащую таблицу с параметрами. В этой таблице

отображаются семь обязательных параметров, общих для всех форм: «Вычислитель», «Время моделирования, с», «Входные данные», «Дашборд», «Мнемосхема», «Результаты другого расчета», «Сохранение структуры входных данных».

7. [опционально] Пользователь нажимает кнопку редактирования обязательного параметра.

8. Система отображает модальное окно редактирования параметра.

9. Пользователь задает значение по умолчанию для обязательного параметра и нажимает кнопку «Сохранить».

10. Система сохраняет изменения и отображает сообщение «Успешно обновлено».

11. Система отображает изменения в таблице параметров.

12. [опционально] Пользователь переходит в раздел «Группы».

13. Система отображает таблицу с перечнем групп.

14. Пользователь нажимает кнопку «Добавить» в шапке таблицы.

15. Пользователь задаёт:

15.1. Название группы.

15.2. Описание группы [опционально].

15.3. Ключ группы.

16. Пользователь нажимает кнопку «Сохранить».

17. Система создает группу параметров.

18. Система отображает сообщение «Успешно создано».
19. Созданная группа отображается в таблице параметров.
20. Пользователь переходит в раздел «Параметры».
21. Пользователь нажимает кнопку «Добавить» в шапке таблицы.
22. Система отображает модальное окно добавления параметра.
23. Пользователь задаёт:
 - 14.1. Название параметра.
 - 14.2. Тип параметра.
 - 14.3. Ключ параметра .
 - 14.4. Значение по умолчанию [опционально].
 - 14.5. Группу [опционально].
 - 14.6. Обязательность заполнения параметра.
 - 14.7. Возможность множественных значений параметра.
24. Пользователь нажимает кнопку «Сохранить».
25. Система создает параметр формы.
26. Система отображает сообщение «Успешно создано».
27. Созданный параметр отображается в таблице параметров.
28. [опционально] Пользователь переходит в раздел «Ограничения».
29. Система отображает таблицу ограничений.

	<p>30. Пользователь нажимает кнопку «Добавить ограничение» в шапке таблицы.</p> <p>31. Система отображает модальное окно наложения ограничения на параметр.</p> <p>32. Пользователь задает левую часть оператора, оператор, правую часть оператора и нажимает кнопку «Сохранить».</p> <p>33. Система отображает созданное ограничение в соответствующей таблице.</p> <p>34. Пользователь нажимает кнопку «Назад».</p> <p>35. Система отображает страницу с перечнем форм, содержащий также форму, созданную пользователем ранее.</p>
<p>Альтернативный сценарий</p>	<p><i>9А. Пользователь указал невалидное значение по умолчанию для обязательного параметра «Время моделирования, с».</i></p> <p>1. Система заблокирует возможность редактирования параметра.</p> <p>2. Система подсвечивает поле красным цветом и отображает сообщение об ошибке, например, «Число должно быть больше или равно 0» при наведении на это поле.</p> <p><i>9Б. Пользователь отменяет редактирование параметра:</i></p> <p>1. Система отменяет редактирование параметра и не сохраняет введенные данные.</p>

16А. Пользователь не заполнил обязательные поля

1. Система заблокирует возможность создания группы.
2. Система подсвечивает незаполненные поля красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.

16Б. Пользователь вводит некорректное значение ключа

1. Система подсвечивает поле «Ключ» красным цветом и отображает сообщение об ошибке «Поле должно содержать только латинские буквы и цифры» при наведении на это поле.

16В. Пользователь указывает название группы, которая уже существует в форме

1. Система заблокирует возможность создания группы и отобразит сообщение об ошибке.

16Г. Пользователь указывает ключ группы, которая уже существует в форме

1. Система заблокирует возможность создания группы и отобразит сообщение об ошибке.

16Д. Пользователь отменяет создание группы

1. Система отменяет создание параметра и не сохраняет введенные данные.

24А. Пользователь не заполнил обязательные поля

1. Система заблокирует возможность создания параметра.
2. Система подсвечивает незаполненные поля красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.

24Б. Пользователь вводит некорректное значение ключа

1. Система подсвечивает поле «Ключ» красным цветом и отображает сообщение об ошибке «Поле должно содержать только латинские буквы и цифры» при наведении на это поле.

24В. Пользователь вводит некорректное значение по умолчанию

1. Система подсвечивает поле «Значение по умолчанию» красным цветом и отображает сообщение об ошибке при наведении на это поле.

24Г. Пользователь указывает название параметра, который уже существует в форме

1. Система заблокирует возможность создания параметра и отобразит сообщение об ошибке.

24Д. Пользователь указывает ключ параметра, который уже существует в форме

	<p>1. Система заблокирует возможность создания параметра и отобразит сообщение об ошибке.</p> <p><i>24Е. Пользователь отменяет создание параметра</i></p> <p>1. Система отменяет создание параметра и не сохраняет введенные данные.</p> <p><i>32А. Пользователь не заполнил обязательные поля</i></p> <p>1. Система заблокирует возможность создания ограничения.</p> <p>2. Система подсвечивает незаполненное поле красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.</p> <p><i>32Б. Ограничение, заданное пользователем, противоречит значению по умолчанию параметра</i></p> <p>1. Система заблокирует возможность создания ограничения.</p> <p>2. Система отображает сообщение об ошибке «Значение не удовлетворяет ограничению».</p> <p><i>32В. Пользователь отменяет создание ограничения до сохранения:</i></p> <p>1. Система отменяет создание ограничения и не сохраняет введенные данные.</p>
Исключения	Сбой в работе системы.

Создание и настройка дашборда

Краткое описание сценария	Предоставить пользователю возможность создавать дашборд
Участники	Пользователь Система
Предусловие (контекст)	Пользователь авторизован в системе. Пользователь создал проект. Пользователь создал вычислитель. Пользователь находится на странице проекта. Пользователь создал и запустил на выполнение расчет. Расчет имеет статус «Успешно».
Постусловие (результат)	Пользователь успешно создал дашборд.
Основной сценарий	<ol style="list-style-type: none"> 1. Пользователь переходит на вкладку «Дашборды». 2. Система отображает пустую таблицу шаблонов дашбордов. 3. Пользователь нажимает на кнопку «Добавить» в шапке таблицы. 4. Система отображает модальное окно создания нового дашборда. 5. Пользователь задает название дашборда, выбирает вычислитель, указывает, является ли дашборд дефолтным для данного вычислителя, выбирает расчет в качестве источника данных и нажимает кнопку «Сохранить». 6. Система отображает пустую область редактирования дашборда. 7. Пользователь нажимает кнопку «Создать виджет».

	<p>8. Система отображает выпадающий список с доступными типами виджетов (линейный график, круговая диаграмма, гистограмма, и т.д.).</p> <p>9. Пользователь выбирает тип виджета.</p> <p>10. Система отображает модальное окно редактирования виджета выбранного типа.</p> <p>11. Пользователь задает параметры виджета и нажимает кнопку «Сохранить».</p> <p>14. Система сохраняет настройки виджета и отображает его в доступной области редактирования.</p> <p>15. [опционально] Пользователь меняет размер виджета и его позицию.</p> <p>16. Система сохраняет настройки отображения виджета.</p> <p>17. Пользователь заканчивает настройку дашборда и нажимает кнопку возвращения на предыдущую страницу.</p> <p>18. Система отображает страницу с таблицей шаблонов, содержащей также дашборд, созданный пользователем ранее.</p>
<p>Альтернативный сценарий</p>	<p><i>5А. Пользователь не заполнил обязательные поля</i></p> <p>1. Система заблокирует возможность создания дашборда.</p> <p>2. Система подсвечивает незаполненные поля красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.</p>

	<p><i>5Б. Пользователь указывает название дашборда, который уже существует в системе</i></p> <ol style="list-style-type: none"> 1. Система заблокирует возможность создания дашборда. 2. Система отображает сообщение об ошибке «Дашборд с таким названием уже существует в системе». <p><i>5В. Пользователь отменяет создание дашборда</i></p> <ol style="list-style-type: none"> 1. Система отменяет создание дашборда и не сохраняет введенные данные. <p><i>11А. Пользователь не заполнил обязательные поля</i></p> <ol style="list-style-type: none"> 1. Система заблокирует возможность создания виджета. 2. Система подсвечивает незаполненные поля красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.
Исключения	Сбой в работе системы.

Создание и настройка мнемосхемы

Краткое описание сценария	Предоставить пользователю возможность создавать и настраивать мнемосхему
Участники	Пользователь Система
Предусловие (контекст)	Пользователь авторизован в системе. Пользователь создал проект. Пользователь создал вычислитель.

	<p>Пользователь находится на странице проекта.</p> <p>Пользователь создал и запустил на выполнение расчет.</p> <p>Расчет имеет статус «Успешно».</p>
Постусловие (результат)	Пользователь успешно создал мнемосхему.
Основной сценарий	<ol style="list-style-type: none"> 1. Пользователь переходит на вкладку «Мнемосхемы». 2. Система отображает раздел «Мнемосхемы» с перечнем мнемосхем. 3. [опционально] Пользователь переходит в раздел «Библиотека элементов». 4. Пользователь нажимает кнопку добавления каталога. 5. Система отображает модальное окно создания нового каталога. 6. Пользователь задает название каталога и нажимает кнопку «Сохранить». 7. Система отображает созданный каталог в перечне. 8. Пользователь нажимает на название каталога. 9. Пользователь нажимает кнопку «Добавить элемент». 10. Пользователь указывает название элемента и загружает изображение элемента. 11. Система сохраняет данные об элементе и отображает элемент в перечне. 12. Пользователь переходит в раздел «Мнемосхемы».

	<p>13. Пользователь нажимает кнопку «Добавить мнемосхему» .</p> <p>14. Пользователь задает название мнемосхемы, выбирает вычислитель, указывает, является ли мнемосхема дефолтной для данного вычислителя, выбирает расчет в качестве источника данных и нажимает кнопку «Сохранить».</p> <p>15. Система отображает пустую область редактирования мнемосхемы.</p> <p>16. Пользователь нажимает кнопку добавления элемента.</p> <p>17. Пользователь выбирает вид элемента, а также, если необходимо, задает настройки отображения и привязывает данные к элементу.</p> <p>18. Пользователь нажимает кнопку «Сохранить».</p> <p>19. Система отрисовывает элемент в доступной области мнемосхемы.</p> <p>20. [опционально] Пользователь меняет размер и расположение элемента.</p> <p>21. Пользователь создает второй элемент (п.16-п.20).</p> <p>22. Пользователь соединяет созданные элементы.</p> <p>23. Пользователь задает настройки линии соединения.</p>
Альтернативный сценарий	<p><i>6А. Пользователь не заполнил обязательные поля</i></p> <p>1. Система заблокирует возможность создания каталога.</p>

2. Система подсвечивает незаполненные поля красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.

6Б. Пользователь указывает название каталога, который уже существует в системе

1. Система заблокирует возможность создания каталога.

2. Система отображает сообщение об ошибке «Каталог с таким названием уже существует в системе».

10А. Пользователь не заполнил обязательные поля

1. Система заблокирует возможность создания элемента.

2. Система подсвечивает незаполненные поля красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.

10Б. Пользователь указывает название элемента, который уже существует в системе

1. Система заблокирует возможность создания элемента.

2. Система отображает сообщение об ошибке «Элемент с таким названием уже существует в системе».

14А. Пользователь не заполнил обязательные поля

	<p>1. Система заблокирует возможность создания мнемосхемы.</p> <p>2. Система подсвечивает незаполненные поля красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.</p> <p><i>14Б. Пользователь указывает название мнемосхемы, которая уже существует в системе</i></p> <p>1. Система заблокирует возможность создания мнемосхемы.</p> <p>2. Система отображает сообщение об ошибке «Мнемосхема с таким названием уже существует в системе».</p> <p><i>14В. Пользователь отменяет создание мнемосхемы</i></p> <p>1. Система отменяет создание мнемосхемы и не сохраняет введенные данные.</p>
Исключения	Сбой в работе системы.

Интеграция с сервисом обработки задач

Краткое описание сценария	Предоставить пользователю возможность сгенерировать сервисный токен и подключить сервер для обработки вычислений
Участники	Пользователь Система Сервис обработки задач

<p>Предусловие (контекст)</p>	<p>Пользователь авторизован в системе. Пользователь создал проект. Пользователь находится на странице проекта. Системный администратор предоставил пользователю токен для подключения сервера.</p>
<p>Постусловие (результат)</p>	<p>Пользователь успешно подключил сервис обработки вычислений к системе.</p>
<p>Основной сценарий</p>	<ol style="list-style-type: none"> 1. Пользователь переходит на вкладку «Настройки». 2. Пользователь нажимает на раздел «Токены». 3. Система отображает пустую таблицу токенов. 4. Пользователь нажимает на кнопку «Сгенерировать токен» в шапке таблицы. 4. Система отображает модальное окно генерации токена. 5. Пользователь задает название токена, описание, дату истечения срока действия и роль. 6. Пользователь нажимает кнопку «Сохранить». 7. Система отображает пользователю сгенерированный токен. 8. Пользователь нажимает кнопку «Скопировать». 9. Система отображает сообщение «Токен скопирован в буфер обмена». 10. Пользователь закрывает модальное окно. 11. Система отображает страницу с таблицей токенов, содержащей также токен, созданный пользователем ранее.

12. Пользователь открывает приложение для управления серверами обработки вычислений.
13. Сервис отображает пользователю список серверов и обработчиков (если таковые есть).
13. Пользователь нажимает «+».
14. Сервис отображает окно для подключения к серверу.
15. Пользователь вводит адрес сервера и токен, переданный ему системным администратором.
16. Пользователь нажимает кнопку «Добавить».
17. Сервис добавляет новый сервер в список доступных серверов и отображает его в перечне.
18. Пользователь нажимает на кнопку просмотра информации о сервере (i).
19. Пользователь нажимает на кнопку «Добавить исполнителя».
20. Пользователь задает:
 - 20.1. Название
 - 20.2. Токен (сгенерированный в п.7)
 - 20.3. Рабочую директорию исполнителя
 - 20.4. Путь к интерпретатору Python
 - 20.5. Интервал поиска новых задач на расчет
 - 20.6. Интервал оповещения о прогрессе задачи
 - 20.7. Количество параллельно выполняемых задач
21. Пользователь нажимает кнопку «Сохранить».

	<p>22. Сервис добавляет нового исполнителя и отображает его в списке активных исполнителей для выбранного сервера.</p>
<p>Альтернативный сценарий</p>	<p><i>5А. Дата истечения срока действия, указанная пользователем, меньше, чем текущая дата</i></p> <ol style="list-style-type: none"> 1. Система заблокирует возможность генерация токена. 2. Система подсвечивает поле «Дата истечения срока действия» красным цветом и отображает сообщение об ошибке «Дата не может быть меньше <текущая дата>» при наведении на это поле. <p><i>6А. Пользователь не заполнил обязательные поля</i></p> <ol style="list-style-type: none"> 1. Система заблокирует возможность генерации токена. 2. Система подсвечивает незаполненные поля красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле. <p><i>6Б. Пользователь отменил генерацию токена</i></p> <ol style="list-style-type: none"> 1. Система отменяет генерацию токена и не сохраняет введенные данные. <p><i>8А. Пользователь закрыл модальное окно, не скопировав токен</i></p> <ol style="list-style-type: none"> 1. Система не предоставляет возможность повторного просмотра сгенерированного токена из соображений безопасности.

	<p>2. Пользователь генерирует новый токен.</p> <p><i>16A. Сервер недоступен</i></p> <p>1. Сервис отображает пользователю сообщение об ошибке.</p>
Исключения	Сбой в работе системы, сбой в работе сервиса обработки вычислений

Создание и запуск расчета

Краткое описание сценария	Предоставить пользователю возможность создать и запустить расчет.
Участники	<p>Пользователь</p> <p>Система</p> <p>Сервис обработки вычислений</p>
Предусловие (контекст)	<p>Пользователь авторизован в системе.</p> <p>Пользователь создал проект.</p> <p>Пользователь находится на странице проекта.</p> <p>Пользователь создал таблицу.</p> <p>Пользователь создал вычислитель.</p> <p>Пользователь создал форму.</p>
Постусловие (результат)	Пользователь успешно запустил расчет.
Основной сценарий	<p>1. Пользователь переходит на вкладку «Расчеты».</p> <p>2. Система отображает пустую таблицу расчетов.</p> <p>3. Пользователь нажимает кнопку «Создать расчет» в шапке таблицы.</p>

4. Система отображает модальное окно создания расчета.
5. Пользователь задает название расчета, описание и форму параметров, которая определяет набор параметров, необходимых для выполнения расчета.
6. Система автоматически отображает параметры, предусмотренные выбранной формой.
7. Пользователь вводит значения для каждого из отображенных параметров.
8. Пользователь нажимает кнопку «Сохранить».
9. Система обновляет статус расчета на «Создан».
10. Начинается процесс подготовки данных для расчета, о чем система информирует пользователя, отображая статус «Подготовка входных данных».
11. После завершения подготовки данных система автоматически переводит задачу в статус «Готов к запуску» и отображает его пользователю.
12. Пользователь нажимает кнопку «Запустить расчет».
13. Система отображает сообщение «Расчет успешно запущен».
14. Система добавляет расчет в очередь и меняет статус расчета с «Готов к запуску» на «В очереди», отображая его пользователю.
15. Сервис обработки задач периодически обращается по токenu в систему для получения новых задач в статусе «В очереди».

	<p>16. Система оповещает Сервис обработки задач о наличии подходящей задачи и меняет её статус на «Ожидает скачивания», отображая его пользователю.</p> <p>17. Сервис обработки задач запрашивает у системы данные для расчета задачи.</p> <p>18. Система запаковывает необходимые данные в ZIP-архив и выгружает его сервису.</p> <p>19. Сервис обработки задач скачивает ZIP-архив, сообщая системе об этом.</p> <p>20. Система меняет статус задачи на «Скачивание», отображая его пользователю.</p> <p>21. Сервис распаковывает ZIP-архив, настраивает среду выполнения, запускает задачу на расчет и оповещает об этом систему.</p> <p>22. Система меняет статус задачи на «В работе» и отображает его пользователю.</p> <p>23. После окончания расчета сервис обработки задач выгружает результаты задачи в виде ZIP-архива в систему.</p> <p>23. Система меняет статус задачи на «Успешно» и отображает его пользователю.</p> <p>24. [опционально] Пользователь нажимает кнопку скачивания результатов расчета в контекстном меню задачи.</p>
Альтернативный сценарий	<p><i>5А. Пользователь не заполнил обязательные поля</i></p> <p>1. Система заблокирует возможность создания расчета.</p>

2. Система подсвечивает незаполненные поля красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.

5Б. Пользователь отменил создание расчета

1. Система отменяет создание расчета и не сохраняет введенные данные.

8А. Пользователь не заполнил обязательные поля

1. Система блокирует возможность создания расчета.

2. Система подсвечивает незаполненные поля красным цветом и отображает сообщение об ошибке «Обязательное поле» при наведении на это поле.

8Б. Пользователь отменил создание расчета

1. Система отменяет создание расчета и не сохраняет введенные данные.

15А. Срок действия токена истек

1. Система отказывает Сервису обработки задач в доступе в систему.

15Б. Сервис обработки задач имеет недостаточно прав на получение данных задачи

1. Система отказывает Сервису обработки задач в доступе в систему.

	<p><i>17А. Сервис обработки задач имеет недостаточно прав на получение данных задачи</i></p> <p>1. Система отказывает Сервису обработки задач в доступе в систему.</p> <p><i>19А. Сервис обработки задач имеет недостаточно прав на получение данных задачи</i></p> <p>1. Система отказывает Сервису обработки задач в доступе в систему.</p> <p><i>21А. Структура ZIP-архива не соответствует требованиям</i></p> <p>1. Сервис обработки задач оповещает об этом систему.</p> <p>2. Система меняет статус задачи на «Ошибка».</p> <p><i>21Б. Возникли проблемы с настройкой среды выполнения</i></p> <p>1. Сервис обработки задач оповещает об этом систему.</p> <p>2. Система меняет статус задачи на «Ошибка».</p>
Исключения	Сбой в работе системы, сбой в работе сервиса обработки вычислений

Просмотр результатов расчета на дашборде

Краткое описание сценария	Предоставить пользователю возможность просматривать результаты расчета на дашборде
Участники	Пользователь

	Система
Предусловие (контекст)	<p>Пользователь авторизован в системе.</p> <p>Пользователь находится на странице проекта.</p> <p>Пользователь создал и запустил расчет.</p> <p>Расчет завершен с результатом «Успешно».</p> <p>Пользователь находится на вкладке «Расчеты».</p>
Постусловие (результат)	Пользователь успешно просмотрел результаты расчета на дашборде.
Основной сценарий	<ol style="list-style-type: none"> 1. Пользователь нажимает кнопку просмотра результатов расчета через дашборд. 2. Система проверяет, какой шаблон дашборда указан в параметрах задачи. 3. Система перенаправляет пользователя на новую вкладку и отображает дашборд, заполняя виджеты данными из результатов расчета.
Альтернативный сценарий	<p><i>2А. Пользователь не указал дашборд в параметрах задачи</i></p> <ol style="list-style-type: none"> 1. Система определяет, какой вычислитель был применен для расчета задачи. 2. Система проверяет, задан ли шаблон по умолчанию для вычислителя. 3.а. Если шаблон задан, система отображает результаты расчета с использованием этого шаблона дашборда. 3.б. Если шаблон не задан, система предлагает пользователю выбрать шаблона дашборда для отображения результатов.

	<p><i>3А. Система не смогла применить шаблон дашборда к результатам расчета</i></p> <p>1. Система выделяет красным цветом виджеты, которые не удалось сгенерировать.</p>
Исключения	Сбой в работе системы

Просмотр результатов расчета на мнемосхеме

Краткое описание сценария	Предоставить пользователю возможность просматривать результаты расчета на мнемосхеме
Участники	<p>Пользователь</p> <p>Система</p>
Предусловие (контекст)	<p>Пользователь авторизован в системе.</p> <p>Пользователь находится на странице проекта.</p> <p>Пользователь создал и запустил расчет.</p> <p>Расчет завершен с результатом «Успешно».</p> <p>Пользователь находится на вкладке «Расчеты».</p>
Постусловие (результат)	Пользователь успешно просмотрел результаты расчета на мнемосхеме.
Основной сценарий	<p>1. Пользователь нажимает кнопку просмотра результатов расчета через мнемосхему.</p> <p>2. Система проверяет, какая мнемосхема указана в параметрах расчета.</p> <p>3. Система перенаправляет пользователя на новую вкладку и отображает мнемосхему, заполняя элементы данными из результатов расчета.</p>
Альтернативный сценарий	<i>2А. Пользователь не указал мнемосхему в параметрах задачи</i>

	<p>1. Система определяет, какой вычислитель был применен для расчета задачи.</p> <p>2. Система проверяет, задана ли мнемосхема по умолчанию для вычислителя.</p> <p>3.а. Если мнемосхема задана, система отображает результаты расчета с использованием этой мнемосхемы.</p> <p>3.б. Если мнемосхема не задана, система предлагает пользователю выбрать мнемосхему для отображения результатов.</p> <p><i>3А. Система не смогла применить мнемосхему к результатам расчета</i></p> <p>1. При нажатии на соответствующий элемент система показывает ошибку отображения данных расчета.</p>
Исключения	Сбой в работе системы