

Министерство цифрового развития, связи и массовых коммуникаций
Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Сибирский государственный университет телекоммуникаций и
информатики»
(СибГУТИ)

Кафедра _____ ТСиВС
Допустить к защите

Зав.каф. _____ Дроздова В.Г.

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

Разработка онлайн приложения для распределения пациентов в группы при
проведении клинического исследования

Пояснительная записка

Студент _____ Ворошин Илья Викторович /...../

Факультет _____ ИВТ _____ Группа _____ ИА-832

Руководитель _____ Лукинов Виталий Леонидович //

Новосибирск 2022 г.

Содержание

Оглавление

ВВЕДЕНИЕ.....	3
1 ПОСТАНОВКА ЗАДАЧИ.....	5
1.1 Цель задачи.....	5
1.2 Разновидности рандомизации.....	5
1.3 Принцип генерации случайных и псевдослучайных последовательностей.....	7
1.4 Ослепление	8
2 ПРОГРАММНАЯ РЕАЛИЗАЦИЯ РАЗРАБОТКИ СЕРВИСА....	9
2.1 Используемые инструменты разработки.....	9
2.2 Структура проекта	10
2.3 Модель клиент-сервер	13
2.4 Реализация базы данных	14
2.5 Алгоритм генерации распределения пациентов на группы .	17
2.6 Алгоритмы навигаций пользователя по сайту	20
2.6.1 Алгоритм навигации неавторизованного пользователя	20
2.6.2 Алгоритм регистрации и авторизации пользователя	21
2.6.3 Алгоритм навигации авторизованного пользователя	22
2.7 Описание пользовательского интерфейса	24
2.7.1 Интерфейс главной страницы	24
2.7.2 Интерфейс регистрации нового пользователя.....	25
2.7.3 Интерфейс авторизации пользователя	26
2.7.4 Интерфейс создания исследования	26
2.7.5 Интерфейс аккаунта пользователя	29
2.7.6 Интерфейс редактирования исследования.....	29
2.7.7 Интерфейс просмотра лога	31
2.7.8 Интерфейс просмотра пациентов	31
2.7.9 Интерфейс просмотра таблиц рандомизации	32
2.7.10 Интерфейс просмотра завершенных исследований	33
2.7.11 Интерфейс просмотра списка пользователей	33
2.7.12 Интерфейс страницы «О сайте»	33
ЗАКЛЮЧЕНИЕ	35
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	36

ВВЕДЕНИЕ

Большой частью клинических исследований являются медицинские исследования в параллельных группах объектов с целью выявления наилучшего способа вмешательства для лечения или профилактики. Для данного дизайна клинических исследований важной проблемой является контроль сопоставимости параллельных групп для чистоты выводов о результатах разных вмешательств в исследуемых группах.

Рандомизированное клиническое исследование (РКИ) – это клиническое исследование, при котором пациенты случайным образом разбиваются по группам вмешательств и получают одно из двух или более объектов исследования. При этом группа, получившая новое вмешательство, называется экспериментальной, а группа, получившая стандартное лечение или плацебо, – контрольной. Такие группы являются параллельными по отношению друг к другу и будут получать выбранное лечение от начала до конца исследования.



Рисунок 1 Схема проведения РКИ в параллельных группах

Важным моментом РКИ является максимальная схожесть групп для обеспечения уверенности в том, что результат исследования будет связан лишь с получаемым препаратом. Для обеспечения сопоставимости исследуемых групп в настоящее время применяются различные методы рандомизированного распределения объектов в группы и методы стратификации, а также комбинации этих методов.

Другой проблемой является «защита» схем распределения от недобросовестного изменения схем исследователями. Решением является «ослепление» исследователей, то есть сокрытие предварительной общей схемы.

Устаревшие методы рандомизации (например, метод конвертов) имеют ряд уязвимостей, не дающих гарантии, что исследование проведено честно и непредвзято. То есть в таких исследованиях существует вероятность недобросовестного вмешательства исследователя в процесс проведения исследования в целях фальсифицировать его результаты.

Поэтому важным является создать приложение для проведения клинических исследований с использованием современных технологии и надёжных методов рандомизации распределения пациентов по группам вмешательств.

В рамках диплома представлен сервис по созданию, проведению и завершению клинических исследований со всеми необходимыми и описанными выше требованиями.

1 ПОСТАНОВКА ЗАДАЧИ

1.1 Цель задачи

Цель работы – разработка онлайн-сервиса для распределения пациентов в группы в клинических исследованиях по схеме двойного слепого исследования (иными словами, чтобы о разбиении на группы не знали ни участники групп, ни исследователи). Также будет предоставлен выбор одного из трёх типов разбиения пациентов на группы: безблочный (простой), блочный с фиксированным и блочный со случайными размером и размерами блоков. Схема формирования групп будет скрыта до полного завершения исследования, принадлежность пациентов к группе будет выдаваться при его регистрации в исследование. Дополнительной особенностью будет ведение лога – записей всех событий, несколько уровней пользователей с разными правами доступа, возможность проведения многоцентровых исследований с ведением пациентов в разных центрах разными исследователями в рамках одного проекта, а также свободный доступ к схеме распределения после завершения проекта.

1.2 Разновидности рандомизации

Рандомизацию можно определить, как процедуру случайного выбора элементов. Её разделяют на фиксированную (простая, блоковая, стратифицированная) и динамическую (метод «несимметричной монеты», адаптивная рандомизация).

Методы динамической рандомизации заключаются в динамических изменениях вероятностей в зависимости от разницы количества набранных людей в ту или иную группу. Минус такого подхода заключается в сложности проведения, а также необходимости частичного раскрытия данных. Рассмотрим поподробнее фиксированные методы.

Простая (обычная) рандомизация создаёт последовательность случайных значений. Наглядный пример – подбрасывание монетки, определяющей, в какую группу пойдет испытуемый. К этому типу также относятся использование таблиц случайных чисел и метод конвертов. Проблема этих способов заключается в формировании неравных групп по количеству участников, дисбаланса по различным признакам (возрасту, полу, степени заболевания и т.п.), возможности подтасовки и нарушении сокрытия распределения. Поэтому к настоящему времени такой подход является устаревшим.

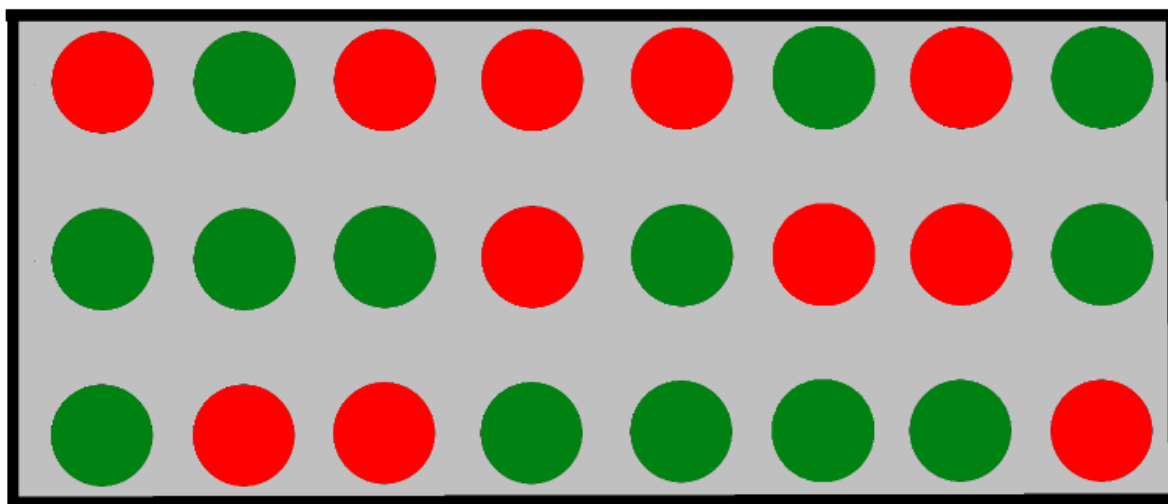


Рисунок 2. Наглядное представление простой рандомизации

Метод блочной рандомизации справляется с проблемой сбалансированности групп по количеству участников, так как, пока не закончится потребность в увеличении количества испытуемых, будут формироваться и заполняться блоки небольших размеров, кратных сумме методов исследования. В этих блоках и проводится рандомизация – в каждом будет случайным образом формироваться последовательность из методов вмешательств, которые в сумме равны друг другу по количеству.

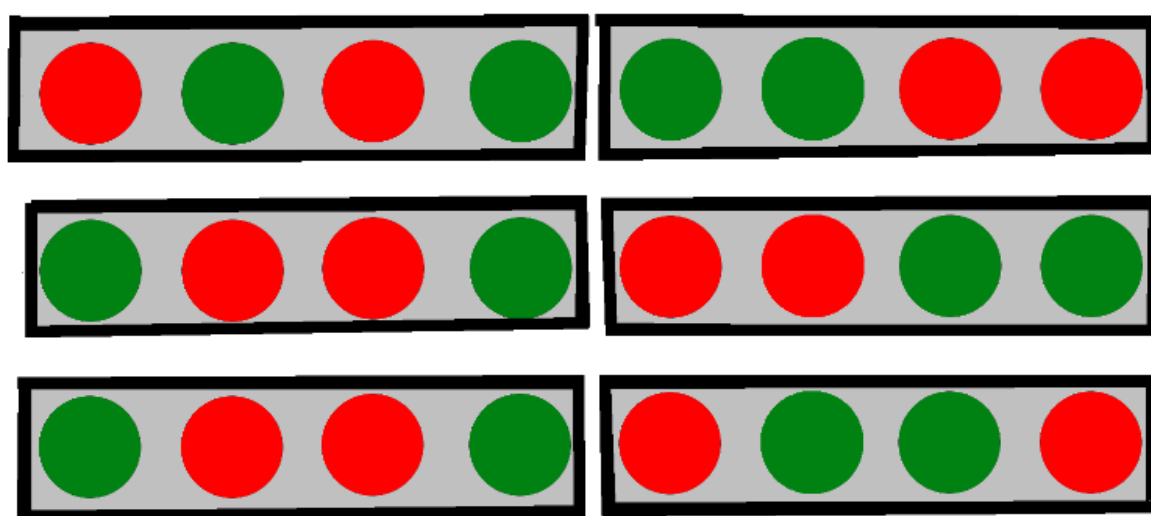


Рисунок 3. Наглядное представление блочной рандомизации с фиксированным размером блоков

Недостаток данного метода – то что исследователь, заранее зная последовательность, может «подтасовывать» пациентов по своему желанию или, зная информации о генераторе псевдослучайных чисел (ГПСЧ), спрогнозировать, в какую группу пойдет следующий испытуемый, что может серьезно повлиять на ход исследования и достоверность её результатов (например, если известен размер блока, предыдущие распределения внутри

блоков и одна из групп уже укомплектована). Данная проблема может решаться случайным определением размера блоков с помощью ГПСЧ, либо сокрытием информации о величине блоков.

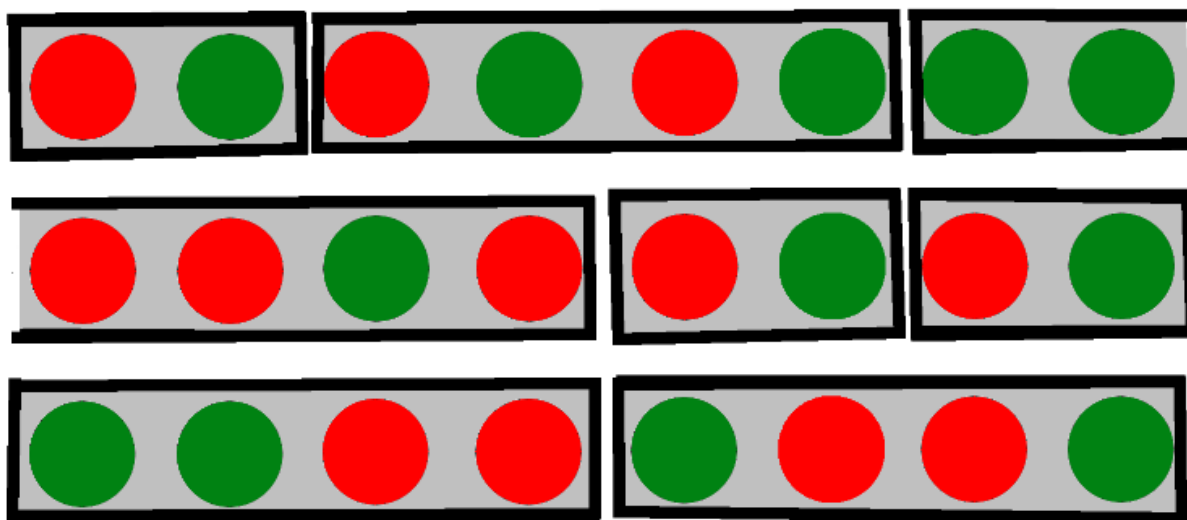


Рисунок 4. Наглядное представление блочной рандомизации со случайными размерами блоков

В стратифицированной рандомизации при формировании групп учитываются некоторые важные отличительные признаки, например, тяжесть состояния или возраст. По этим признакам пациенты разделяются на страты, в которых осуществляется рандомизированный отбор в одну из групп. Таким образом, сформированные выборки получатся репрезентативными по отношению ко всей группе в целом. Однако реализация данного способа является довольно сложной, особенно при необходимости учёта нескольких страт.

1.3 Принцип генерации случайных и псевдослучайных последовательностей

На сегодняшний день актуальными методами рандомизации являются блочные методы, использующие генераторы случайных и псевдослучайных чисел (ГСЧ и ГПСЧ).

ГСПЧ использует в качестве точки опоры начальное число (seed, value), от которого программно по некоему алгоритму преобразований и генерируется последовательность чисел, по своим свойствам близкая к случайной. Эта последовательность сводится к формуле (линейный конгруэнтный генератор):

$$y_{n+1} = (a * y_n + c) \bmod M$$

, где a – множитель, c – приращение, m – модуль.

Получается, проблема такой генерации заключается в невозможности создания полностью случайных последовательностей, так как каждое новое число зависит от предыдущего. То есть, наблюдая за

полученными последовательностями, имеется возможность найти закономерность между числами, решив систему уравнений, и предсказать следующую генерацию

Главное отличие ГСЧ от ГСПЧ в том, что ГСЧ в качестве значений берет высококачественные числа, полученные аппаратным путём из естественных источников случайных чисел, физических шумов, таких как космическое излучение или ионизирующая радиация. Полученные таким путём числа также можно использовать в качестве задаваемых начальных значений (seed) для ГСПЧ. Недостатки таких генераторов включают в себя высокую стоимость, трудозатратность установки и настройки, более медленный процесс генерации, а также невозможность воспроизведения ранее сгенерированных последовательностей.

1.4 Ослепление

Следующей важной задачей является «ослепление» - т.е. сокрытие исследования от лиц, которые могут повлиять на процесс. метод и результат рандомизации принято скрывать от участников: пациентов – во избежание эффекта плацебо, и исследователей – в случае двойного слепого метода, чтобы нейтрализовать возможность недобросовестного вмешательства в распределение пациентов (такой принцип считается более надежным и эффективным для достоверности результатов). В случае исследования, в котором результат зависит от субъективной оценки исследователя, а также есть возможность сделать вывод по видимым признакам – результаты должны быть выведены независимым экспертом, которого тоже можно ослепить (ослепление третьих лиц).

В разрабатываемом сервисе функция ослепления осуществляется в сокрытии блоков распределения пациентов до завершения исследования.

2 ПРОГРАММНАЯ РЕАЛИЗАЦИЯ РАЗРАБОТКИ СЕРВИСА

2.1 Используемые инструменты разработки

Весь код выполнен в интегрированной среде разработки (IDE) PyCharm на языке Python. Используется общая версия IDE Community Edition, чьих функций хватает для разработки данного приложения.

Работать с созданием веб-приложения позволяет фреймворк flask языка Python. С его использованием не только запускается приложение и обрабатываются страницы обработчиком, но и поддерживаются следующие пакеты-расширения: flask_sqlalchemy для работы с базами данных, flash для вывода сообщений пользователю, flask_login для выполнения пользователем входа в профиль и всем с этим связанным.

С помощью библиотеки SQLAlchemy выполняется вся работа с базой данных. Реализуется на технологии ORM (Object-Relational Mapping), которая позволяет связать базы данных с концепциями ООП (объектно-ориентированного программирования). С её помощью можно описать структуру базы данных и способы взаимодействия с ней прямо на языке Python.

За отображение сайта в браузере отвечает язык гипертекстовой разметки HTML (Hypertext Markup Language). HTML отвечает за элементы, отображаемые на сайте. Каждая страница сайта загружает файл с расширением .html, читает структурные элементы, записанные в файл с помощью определённых тэгов, и выводит это всё на экран окна браузера.

Для работы со стилями тэгов HTML по умолчанию используется CSS (Cascading Style Sheets) файл. В этих файлах с помощью определённой формы записи задаются стили для определённых тэгов. Так как стили тэга можно задать в самом html файле, в CSS записываются стили большой длины текста, чтобы не засорять html файл.

Также для использования общих и популярных стилей используется свободный набор инструментов для создания сайтов и веб-приложений Bootstrap. Это готовый набор стилей для дизайна html элементов. Добавляется с помощью ссылки на ресурс в заголовке html файла. Из него был использован шаблон каркаса главной страницы сайта, а также стили для дизайна компонентов веб-интерфейса. Активно применяется система сеток для контейнеров и других компонентов, позволяя быстро и удобно стилизовать их расположение на странице. Данная система построена на принципах и решениях модуля flexbox.

Чтобы использовать главный шаблон во всех отдельных html файлах, отображающих страницы сайта, применяется Python библиотека для рендеринга шаблонов Jinja2. То есть, чтобы исключить лишний текст в html файлах, не печатая в файлах одно и то же по многу раз, общий текст помещается в один единственный файл-шаблон, а остальные файлы, наследованные от него, автоматически принимают структуру шаблона.

Благодаря ей также возможна вставка значений, передаваемых из Python файла, в файл HTML.

В работе для выполнения таких вспомогательных действий к HTML, как динамическое создание блоков по значению другого блока или отображение формы по заданному условию, был использован язык JavaScript в виде скриптов к html документу.

Генерация схем рандомизации для функции создания клинического исследования происходит на языке R с помощью пакета создания блоков рандомизации blockrand.

Возможность использования языка R в интерфейсе Python предоставляет Python библиотека rpy2.

При помощи Python библиотеки werkzeug security пароли пользователей сохраняются в базу данных зашифровано в виде солевого хэша от введённого текста с помощью функции generate_password_hash. По умолчанию, в качестве хеш-функции используется SHA-256, длина соли – 16. Пароли в открытом виде нигде не используются. При авторизации пользователя с помощью функции check_password_hash сравнивается хэш от введённого текста и хэш пароля в базе данных, и, в случае совпадения, выполняется вход в профиль.

2.2 Структура проекта

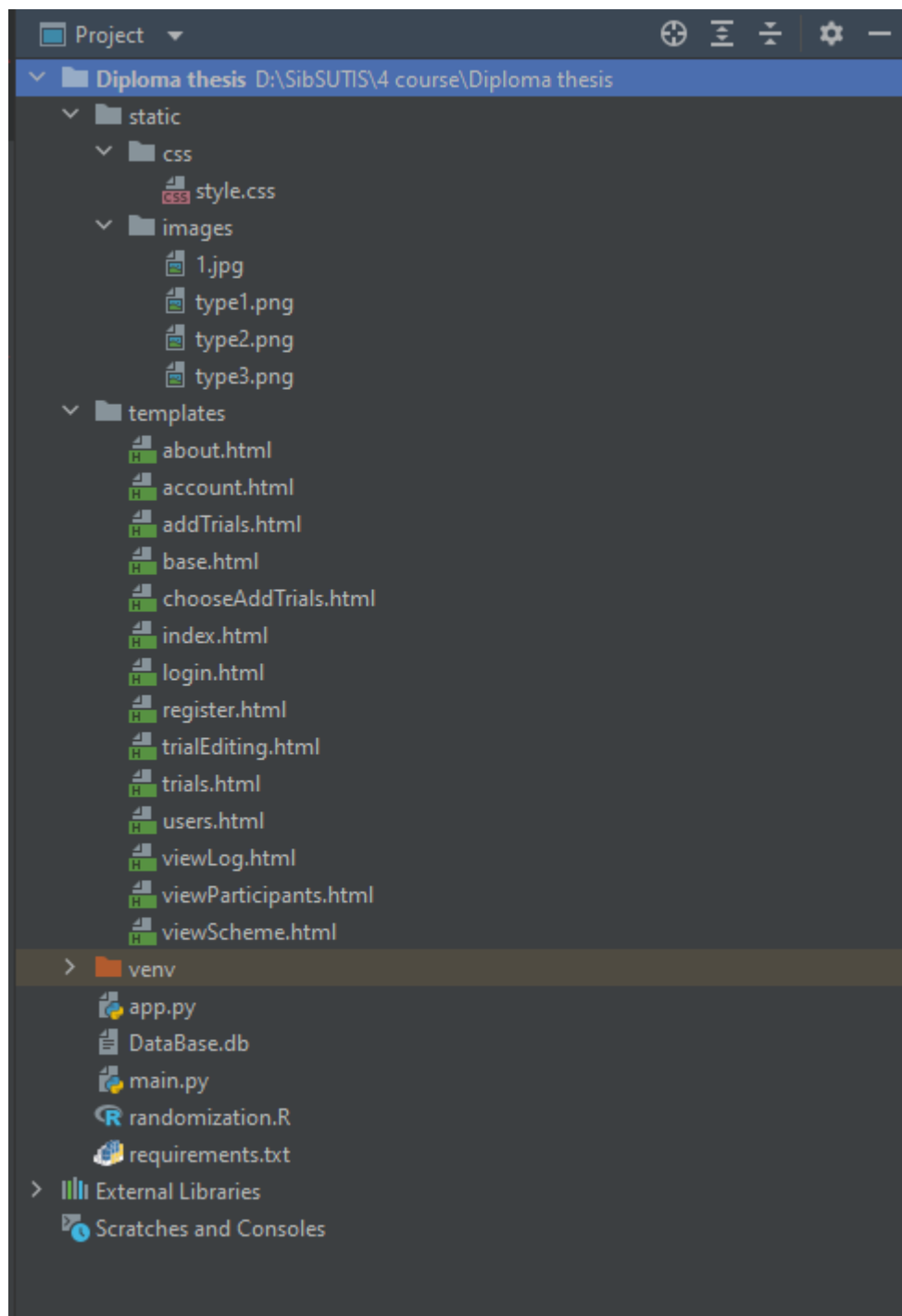


Рисунок 5. Структура проекта в IDE PyCharm

В корневой каталог входят такие файлы, как:

- main.py, из которого происходит запуск приложения
- app.py – основной файл, включает в себя подключение основных библиотек, подключение к базе данных, классы экземпляров базы данных, пути перехода навигации сайта, а также их обработчики.
- DataBase.db – файл базы данных

- randomization.R – содержит в себе функцию на языке R генерации блоков распределения пациентов на группы
- requirements.txt, представляющий из себя файл со списком всех зависимостей версий инструментов проекта

Далее идут папки «static» и «templates», «venv» и «External Libraries».

Папка «static» содержит в себе папку «css» с файлом «style.css», содержащий в себе нужные стили компонентов HTML, и папку «images» с картинками 1.jpg, являющейся фоновой картинкой для всех страниц сайта, а также type1.png, type2.png и type3.png, наглядно показывающие принципы рандомизации при распределении пациентов на группы трёх типов: простая (безблочная), блочная с фиксированным размером блоков и блочная со случайным размером блоков соответственно.

В папке «templates» содержатся каркасы страниц сайта в виде файлов с расширением html. Сюда входят следующие файлы:

- about.html – отображает страницу «О сайте», описывающую предназначение сайта в краткой форме
- account.html – отображает страницу с личным кабинетом авторизованного пользователя, где отображен список с активными и завершенными исследованными пользователя
- addTrials.html – отображает страницу создания исследования с вводом его параметров
- base.html – является шаблоном для всех остальных файлов HTML, отображает шапку сайта в виде ссылок навигации по страницам, а также фоновую картинку
- chooseAddTrials.html – отображает страницу с выбором типа рандомизации создаваемого исследования
- login.html – отображает страницу с функцией авторизации зарегистрированного пользователя
- register.html – отображает страницу с функцией регистрации нового пользователя
- trialEditing.html – отображает страницу редактирования активного исследования авторизованного пользователя с возможностью добавления в него нового пациента, либо завершения исследования при полном списке набранных пациентов
- trials.html – отображает страницу со списком всех завершенных исследований всех пользователей
- users.html – отображает страницу со списком всех зарегистрированных пользователей

- viewLog.html – отображает страницу с записями лога (журнала действий) исследования
- viewParticipants.html – отображает страницу со списком пациентов с их вмешательством
- viewScheme.html – отображает страницу со схемой распределения пациентов исследования

Папки «venv» и «External Libraries» генерируются автоматически при создании проекта в IDE PyCharm и нужны для корректной работы приложения.

2.3 Модель клиент-сервер

В работе используется модель клиент-сервер – вычислительная и сетевая архитектура, при которой программа-сервер ждёт от программы клиента запросы и выводит им свои ресурсы в виде данных или выполнения функций сервера.

Функции сервера выполняет загруженное на сайт приложение. В самом коде функции сервера реализовываются в файле «app.py» с помощью функций переходов по страницам (route) и их обработчиков при помощи написанных функций. Реализовывается с помощью библиотеки flask. Введённые в формы данные от пользователя сервер принимает функцией request.form.get. Далее при помощи SQLAlchemy совершаются необходимые операции с базой данных.

Клиент в своё время поддерживает связь с сервером через формы ввода на страницах. То есть клиент предоставляет необходимые данные через формы ввода, а сервер принимает эти данные методом POST Python библиотеки request, обрабатывает их, выполняет действия с базой данных, и, в случае надобности, предоставляет пользователю ответ.

Обратная связь может осуществляться через инструмент всплывающих сообщений flash библиотеки flask. База данных хранится в файле с расширением db. Связь с ней устанавливается библиотекой SQLAlchemy через СУБД SQLite.

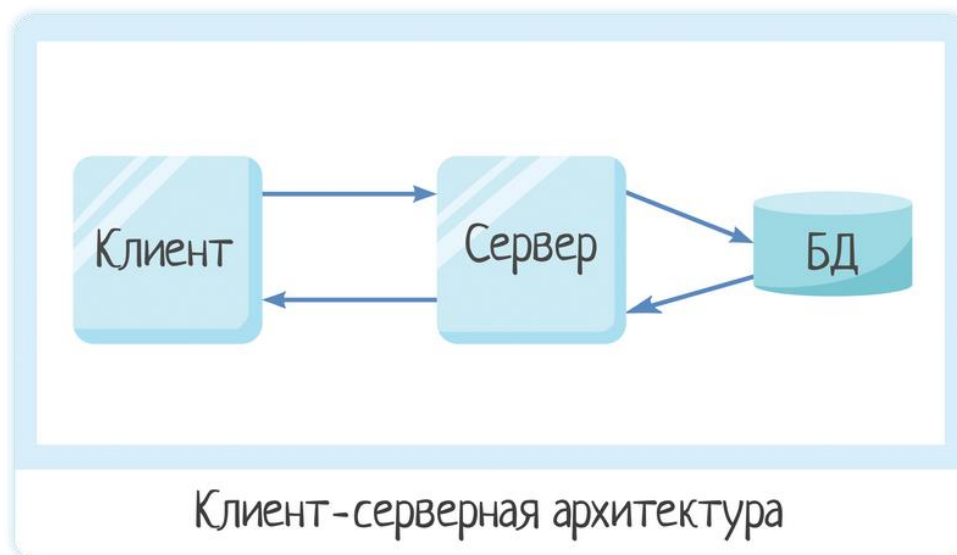


Рисунок 6. Схематичное представление клиент-серверной архитектуры

2.4 Реализация базы данных

База данных создаётся и редактируется программно с помощью библиотеки SQLAlchemy на языке Python. С помощью этой библиотеки устанавливается связь с СУБД SQLite, в экземпляре которой и хранятся все значения.

Схема таблиц данных:

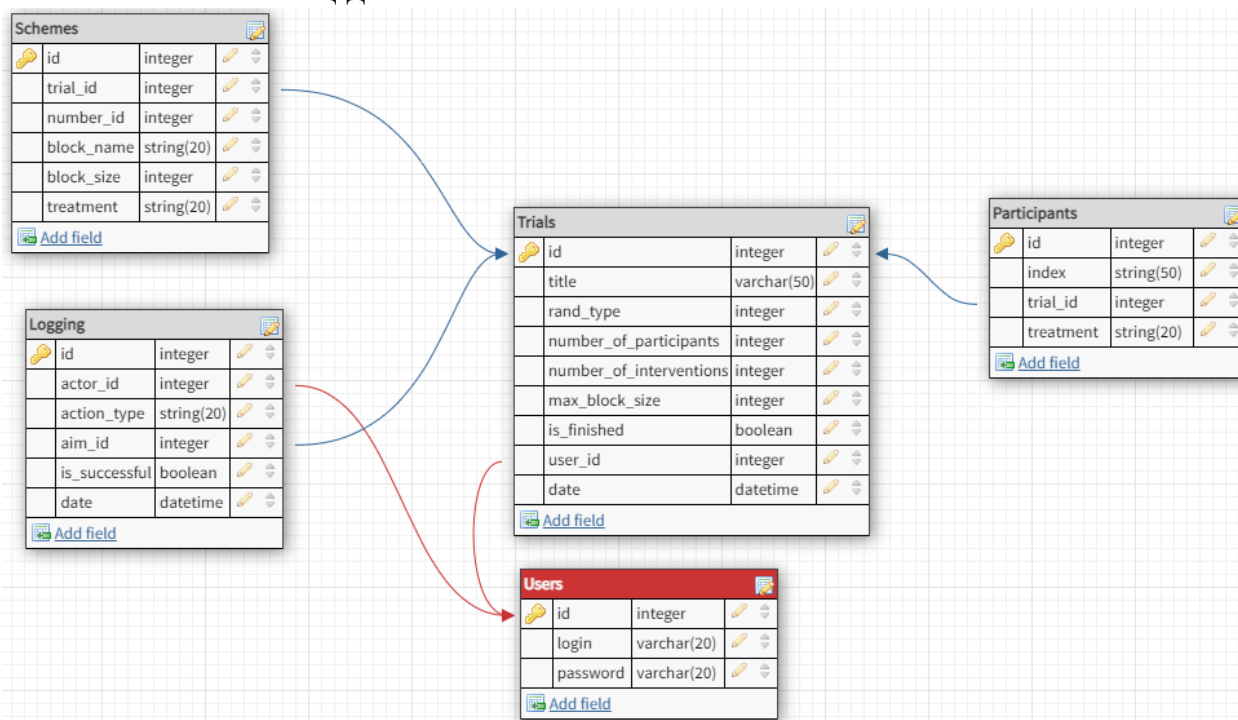


Рисунок 7. Схемы таблиц данных в БД

Таблица пользователей Users:






	Имя	Тип данных	Первичный ключ	Внешний ключ	Уникальность	Проверка	Не NULL
1	id	INTEGER					
2	login	VARCHAR (20)					
3	password	VARCHAR (30)					

Рисунок 8. Таблица Users

- Поле id – идентификатор записи
- Поле login – логин пользователя
- Поле пароль – пароль пользователя, который хранится в зашифрованном виде

Таблица исследований Trials:












	Имя	Тип данных	Первичный ключ	Внешний ключ	Уникальность	Проверка	Не NULL
1	id	INTEGER					
2	title	VARCHAR (50)					
3	date	DATETIME					
4	rand_type	INTEGER					
5	number_of_participants	INTEGER					
6	number_of_interventions	INTEGER					
7	max_block_size	INTEGER					
8	is_finished	BOOLEAN					
9	user_id	INTEGER					

Рисунок 9. Таблица Trials

- Поле id – идентификатор записи
- Поле title – название исследования
- Поле date – дата создания исследования
- Поле rand_type – тип рандомизации исследования. При создании даётся право выбора одного из трёх типов: 1 - безблочный, 2 - блочный с фиксированным размером блоков, 3 - блочный со случайным размером блоков.
- Поле number_of_participants – таблица участников (пациентов) исследования.
- Поле number_of_interventions – количество вмешательств в исследовании.
- Поле max_block_size – максимальный размер блока. При безблочной рандомизации значение равно количеству пациентов, при блочной с фиксированным размером блоков – значение задаётся создателем исследования, при блочной со

случайным количеством блоков – создателем исследования задаётся максимальный размер блока, а сами блоки генерируются случайным образом из множества по произведению множества натуральных чисел на количество вмешательств, не превышая максимальный размер блока.

- Поле `is_finished` – показатель, закончено(True), либо активно(False) исследование. Значение по умолчанию – False.
- Поле `user_id` – идентификатор создателя исследования, внешний ключ на поле `id` таблицы `Users`.

Таблица схем распределения пациентов `Schemes`:









	Имя	Тип данных	Первичный ключ	Внешний ключ	Уникальность	Проверка	Не NULL
1	<code>id</code>	INTEGER					
2	<code>trial_id</code>	INTEGER					
3	<code>number_id</code>	INTEGER					
4	<code>block_name</code>	VARCHAR (20)					
5	<code>block_size</code>	INTEGER					
6	<code>treatment</code>	VARCHAR (20)					

Рисунок 10. Таблица `Schemes`

- Поле `id` – идентификатор записи
- Поле `trial_id` – идентификатор исследования, внешний ключ на поле `id` таблицы `Trials`
- Поле `number_id` – нумерация записи внутри исследования
- Поле `block_name` – название(номер) блока
- Поле `block_size` – размер блока
- Поле `treatment` – название вмешательства

Таблица пациентов `Participants`:







	Имя	Тип данных	Первичный ключ	Внешний ключ	Уникальность	Проверка	Не NULL
1	<code>id</code>	INTEGER					
2	<code>index</code>	VARCHAR (50)					
3	<code>trial_id</code>	INTEGER					
4	<code>treatment</code>	VARCHAR (20)					

Рисунок 11. Таблица `Participants`

- Поле `id` – идентификатор записи
- Поле `index` – уникальный медицинский индекс пациента

- Поле trial_id – идентификатор исследования, внешний ключ на поле id таблицы Trials
- Поле treatment – название вмешательства

Таблица записей действий (лог) Logging:










	Имя	Тип данных	Первичный ключ	Внешний ключ	Уникальность	Проверка	Не NULL
1	id	INTEGER					
2	actor_id	INTEGER					
3	action_type	VARCHAR (20)					
4	aim_id	INTEGER					
5	is_successful	BOOLEAN					
6	date	DATETIME					

Рисунок 12. Таблица Logging

- Поле id – идентификатор записи
- Поле actor_id – идентификатор того, кто совершил действие, внешний ключ на поле id таблицы Users
- Поле action_type – тип совершаемого действия. Используемые значения: "addTrial" - при добавлении исследования, "deleteTrial" – при удалении исследования до его завершения, "addParticipant" – при добавлении нового пациента в исследование, "finishTrial" – при завершении исследования после полного заполнения списка пациентов
- Поле aim_id – идентификатор исследования, к которому применено действие, внешний ключ на поле id таблицы Trials
- Поле is_successful – показатель, является ли операция успешной (True), либо неуспешной (False)
- Поле date – дата совершения операции

2.5 Алгоритм генерации распределения пациентов на группы

В открытом источнике уже существует несколько библиотек программирования для проведения рандомизации в клинических исследованиях (blockrand, randomizeR, pwr, experiment). Одна из них – пакет «blockrand» под язык программирования R. В его функционал входит генерация блоковой рандомизации для РКИ, включая стратифицированный отбор и случайную последовательность сгенерированных блоков. При проведении стратифицированного клинического исследования необходимо запустить функцию по одному разу для каждой страты, а после объединить всё в одну таблицу стандартной функцией rbind.

Вот так выглядит вызов функции генерации списка рандомизации с не менее 60 назначениями и блоками случайных размеров по 3, 6 или 9 человек:

```
bfla <- blockrand(n = 60,
  num.levels = 3, # three treatments
  levels = c("CS", "CS/Tofa", "CS/Upa"), # arm names
  stratum = "Bfail.LowAlb", # stratum name
  id.prefix = "BfLA", # stratum abbrev
  block.sizes = c(1,2,3), # times arms = 3,6,9
  block.prefix = "BfLA") # stratum abbrev
)
```

, где:

`n` – минимальное количество назначений

`num.levels` задаёт сколько видов вмешательств будет использоваться

`stratum` – название страты

`id.prefix` задает приставку к идентификатору `id`

`block.sizes` – размерность блоков

`block.prefix` задаёт приставку к идентификатору блоков `block.id`

##		id	stratum	block.id	block.size	treatment
## 1	BfLA01	Bfail.LowAlb	BfLA01		3	CS
## 2	BfLA02	Bfail.LowAlb	BfLA01		3	CS/Upa
## 3	BfLA03	Bfail.LowAlb	BfLA01		3	CS/Tofa
## 4	BfLA04	Bfail.LowAlb	BfLA02		3	CS/Tofa
## 5	BfLA05	Bfail.LowAlb	BfLA02		3	CS
## 6	BfLA06	Bfail.LowAlb	BfLA02		3	CS/Upa
## 7	BfLA07	Bfail.LowAlb	BfLA03		3	CS/Tofa
## 8	BfLA08	Bfail.LowAlb	BfLA03		3	CS/Upa
## 9	BfLA09	Bfail.LowAlb	BfLA03		3	CS
## 10	BfLA10	Bfail.LowAlb	BfLA04		3	CS
## 11	BfLA11	Bfail.LowAlb	BfLA04		3	CS/Tofa
## 12	BfLA12	Bfail.LowAlb	BfLA04		3	CS/Upa
## 13	BfLA13	Bfail.LowAlb	BfLA05		6	CS/Upa
## 14	BfLA14	Bfail.LowAlb	BfLA05		6	CS/Tofa
## 15	BfLA15	Bfail.LowAlb	BfLA05		6	CS/Tofa
## 16	BfLA16	Bfail.LowAlb	BfLA05		6	CS/Upa
## 17	BfLA17	Bfail.LowAlb	BfLA05		6	CS
## 18	BfLA18	Bfail.LowAlb	BfLA05		6	CS

Рисунок 13. Часть получившейся таблицы распределения случайных размеров пакетом *blockrand* с тремя методами вмешательства

На основе этого примера была создана отдельная функция на языке R, принимающая следующие параметры: количество пациентов, количество вмешательств, названия вмешательств в виде списка, максимальный размер блока. Далее функция преобразовывает список вмешательств к необходимому формату, формирует список применяемых размеров блока и, собственно, генерирует схему распределения пациентов в группы на основе принимаемых параметров. Функция хранится в отдельном файле «randomization.R» и вызывается при создании пользователем нового исследования. То есть, схемы распределения создаются сразу же при создании исследования, и их никак уже не поменять.

Конечная функция рандомизации:

```
library(blockrand)
```

```
randomization <- function(number_of_participants, number_of_interventions,
interventions, max_group_size)
{
  sizes <- c()
  i <- 0
  while(TRUE)
  {
    i <- i+1
    if(i * number_of_interventions > max_group_size)
      break
    else
      sizes <- c(sizes, i)
  }

  bfla <- blockrand(number_of_participants,
                    num.levels = number_of_interventions,
                    levels = c(interventions, recursive=TRUE),
                    block.sizes = sizes,
  )

  return(bfla)
}
```

Создав таблицу распределения пациентов, при записи новых пациентов будут заполняться сгенерированные блоки, таким образом пациенты будут относиться к одному из видов вмешательств. В теории, исследователь, пользуясь этой схемой, должен распределять по группам людей из потока, но этот момент нельзя проследить (проконтролировать), и исследователь может воспользоваться этим слабым местом и распределять пациентов по своему усмотрению в целях фальсификации результатов исследования. Поэтому

схемы распределения будут скрыты до окончания проведения исследования – узнать схемы можно будет только в конце.

2.6 Алгоритмы навигаций пользователя по сайту

Данный раздел описывает возможности поведения пользователя, доступные ему страницы сайта и их функции, в зависимости от его статуса: является ли он авторизованным или неавторизованным в системе.

В таблицах отображены все доступные страницы сайта, какой HTML шаблон при этом генерируется, а также описание их функций.

В блок-схемах наглядно показаны пути переходов пользователя по страницам и их последовательность. В квадратных формах – путь страницы сайта. Стрелками показывается возможность перехода на следующую страницу (что осуществляется через кнопки-ссылки на определенную страницу). В фигуре в виде ромба – условие «если» с двумя ответвлениями: путь, если выражение в блоке истинно (да), и путь, если выражение в блоке ложно (нет).

2.6.1 Алгоритм навигации неавторизованного пользователя

Доступные страницы для неавторизованного пользователя отображены в таблице 1.

Таблица 1 - пути перехода по страницам неавторизованного пользователя

Путь страницы (route)	Генерация шаблона	Описание страницы
/	index.html	Главная страница
/users	users.html	Просмотр зарегистрированных пользователей
/register	register.html	Регистрация
/login	login.html	Авторизации
/trials	trials.html	Просмотр завершенных исследований всех пользователей
/about	about.html	Краткая информация о назначении сайта.
/trials/<Trials.id>/viewLog	viewLog.html	Просмотр лога исследования
/trials/<Trials.id>/viewParticipants	viewParticipants.html	Просмотр включенных в исследование пациентов

Продолжение таблицы 1

/trials/<Trials.id>/viewScheme	viewScheme.html	Просмотр схем рандомизации исследования
--------------------------------	-----------------	---

Как видно со схемы ниже, неавторизованному пользователю с главной страницы доступны пути для просмотра списка зарегистрированных пользователей, списка проведённых исследований, страницы «О сайте», регистрации и авторизации. Со страницы исследований он может перейти на страницы для просмотра журнала записей исследования, включенных в исследование пациентов, а также схем генерации блоков распределения вмешательств.

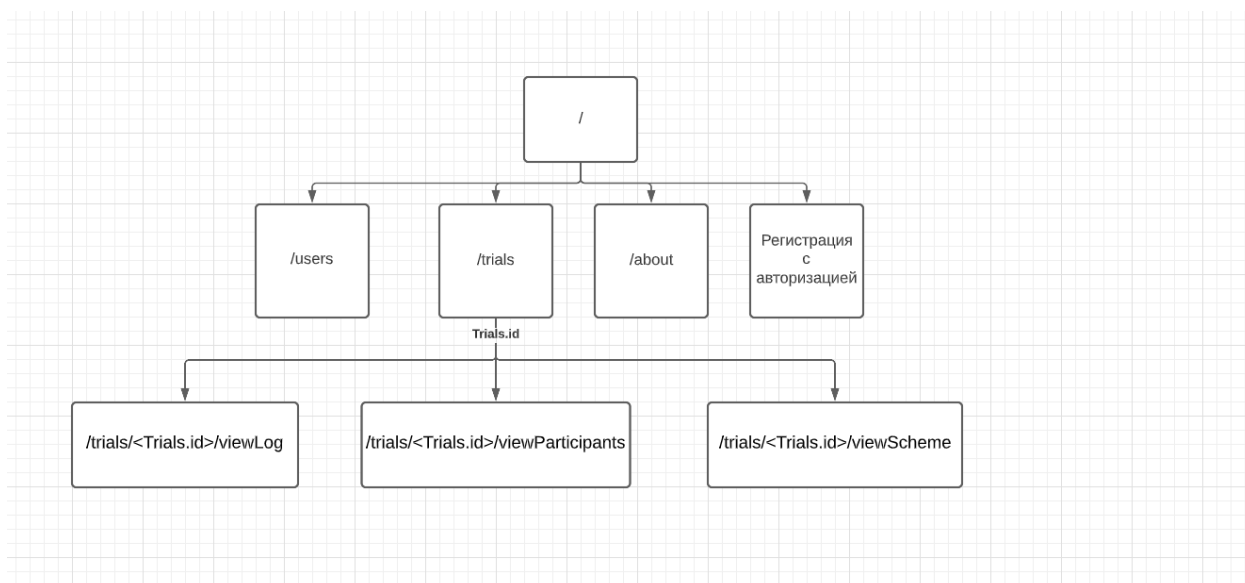


Рисунок 14. Блок-схема навигации неавторизованного пользователя

2.6.2 Алгоритм регистрации и авторизации пользователя

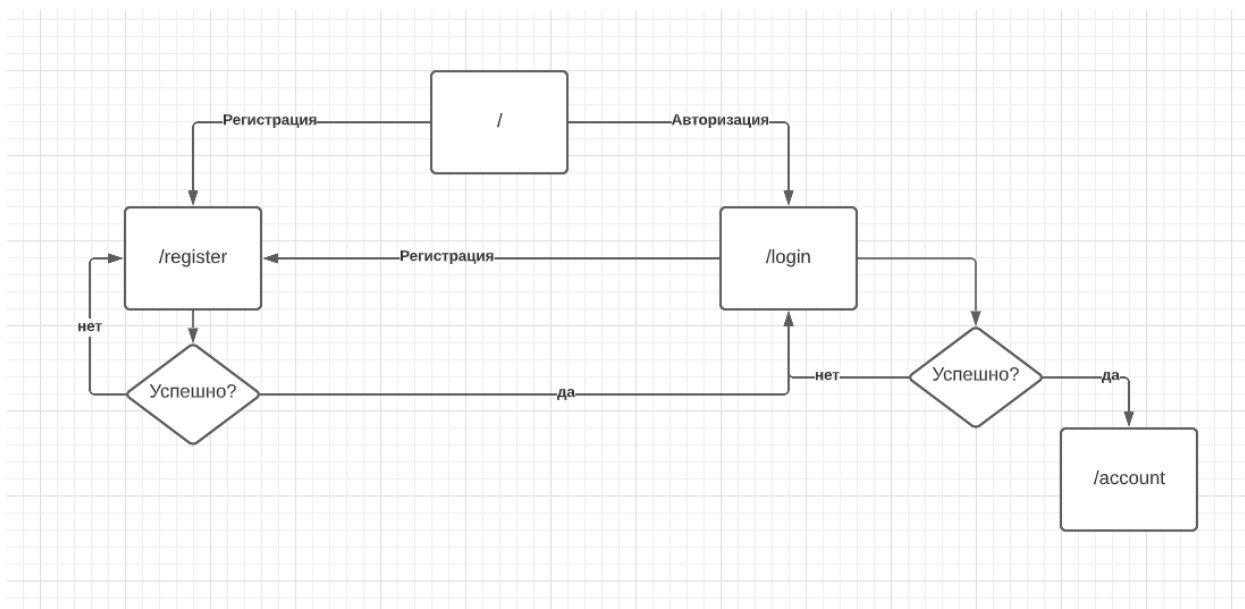


Рисунок 15. Блок-схема процесса регистрации и авторизации пользователя

При успешной регистрации пользователя сразу перебрасывает на страницу авторизации, при неуспешной – обновление страницы с соответствующим предупреждением.

Если авторизация не удалась, то страница обновляется, а пользователю высвечивается соответствующее предупреждение, иначе – пользователя перекидывает на страницу аккаунта.

2.6.3 Алгоритм навигации авторизованного пользователя

Помимо предоставления доступа к новым путям, логично, что сохраняются возможности неавторизованного пользователя. Поэтому далее описываются только новые маршруты в таблице 2.

Таблица 2 - пути перехода по страницам авторизованного пользователя

Путь страницы (route)	Генерация шаблона	Описание страницы
/account	account.html	Аккаунт пользователя со списком активных и завершенных исследований
/trials/<int:Trials.id>/delete	-	Полное удаление исследования до его завершения
/trials/<int:Trials.id>/editing	trialEditing.html	Редактирование активного исследования

Продолжение таблицы 2

/trials/<int:trial_id>/addParticipant	-	Добавление пациента в активное исследование
/trials/<int:trial_id>/finishTrial	-	Завершение активного исследования при полном наборе пациентов
/chooseAddTrials	chooseAddTrials.html	Выбор типа рандомизации при создании нового исследования
/addTrials	addTrials.html	Создание нового исследования с настройкой всех необходимых параметров исследования

С главной страницы сайта авторизованному пользователю представляется возможность войти в аккаунт профиля и создать новое исследование.

Аккаунт профиля представляет собой два списка исследований: активные и завершенные исследования пользователя. В завершенных исследованиях имеются кнопки для просмотра лога, пациентов и схемы распределения. В активных исследованиях есть функция досрочного удаления исследования, и его редактирование. На странице редактирования исследования есть кнопка добавления нового пациента и, при условии, что набор пациентов набран, завершить исследование, что переводит его в статус завершенного исследования.

При создании исследования пользователя сначала кидает на страницу выбора типа рандомизации исследования, а затем на страницу уже создания исследования с вводом его ключевых параметров.

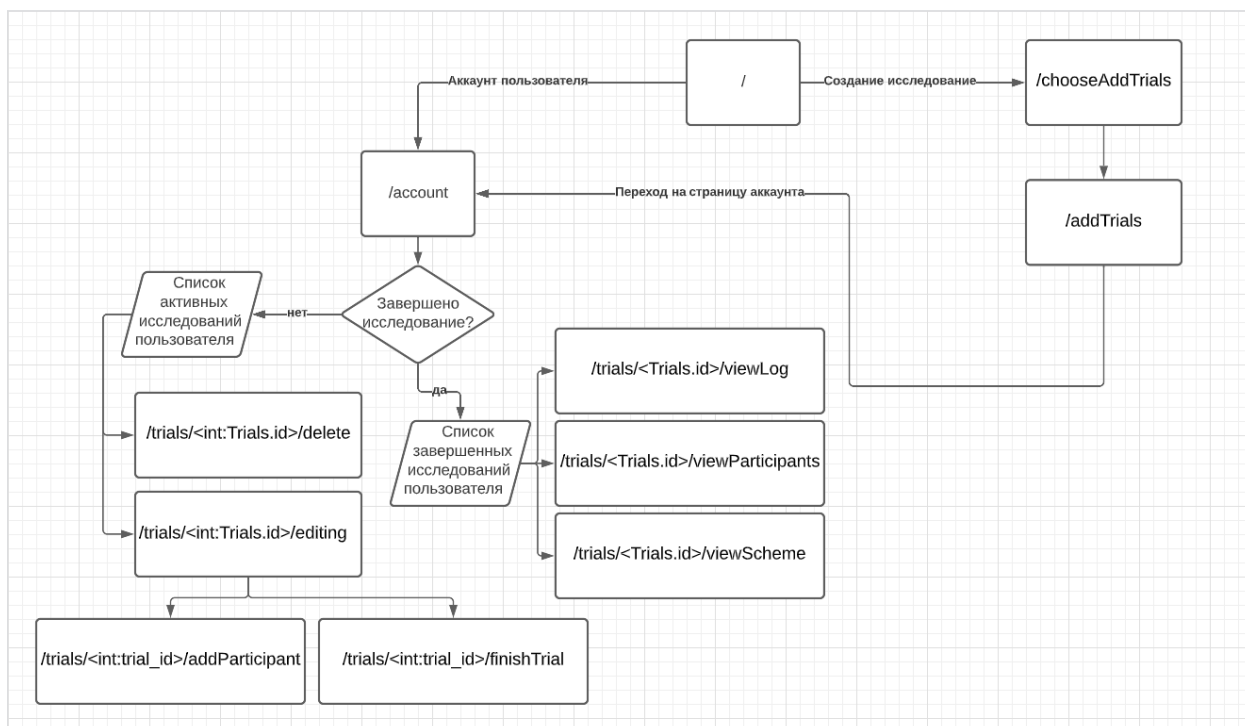


Рисунок 16. Блок-схема навигации авторизованного пользователя

2.7 Описание пользовательского интерфейса

Создание интерфейса пользователя является важной работой, потому что одним из гарантов качества программы является показатель, насколько пользователю будет интуитивно понятно разобраться в процессе работы с программой.

Программный интерфейс пользователя веб-приложения создан по принципу многостраничности сайта. То есть на каждой странице выполняется своя отдельная функция, причем пользователю не должно доставлять проблемы перейти с одной страницы на другую. Поэтому на каждой странице создаются ссылки для перехода на новую вкладку в виде кнопки с кратким текстом о том, куда и зачем ведет эта ссылка. Причём ссылки расставляются именно на нужных страницах для логичности продолжения пути пользователя.

2.7.1 Интерфейс главной страницы

Сайт начинается с главной страницы. Попадая на неё, пользователю отображаются ключевые слова использования сервиса, а также шапка сайта с кнопками-ссылками для перехода на следующие страницы с дальнейшими действиями функционала веб-приложения.

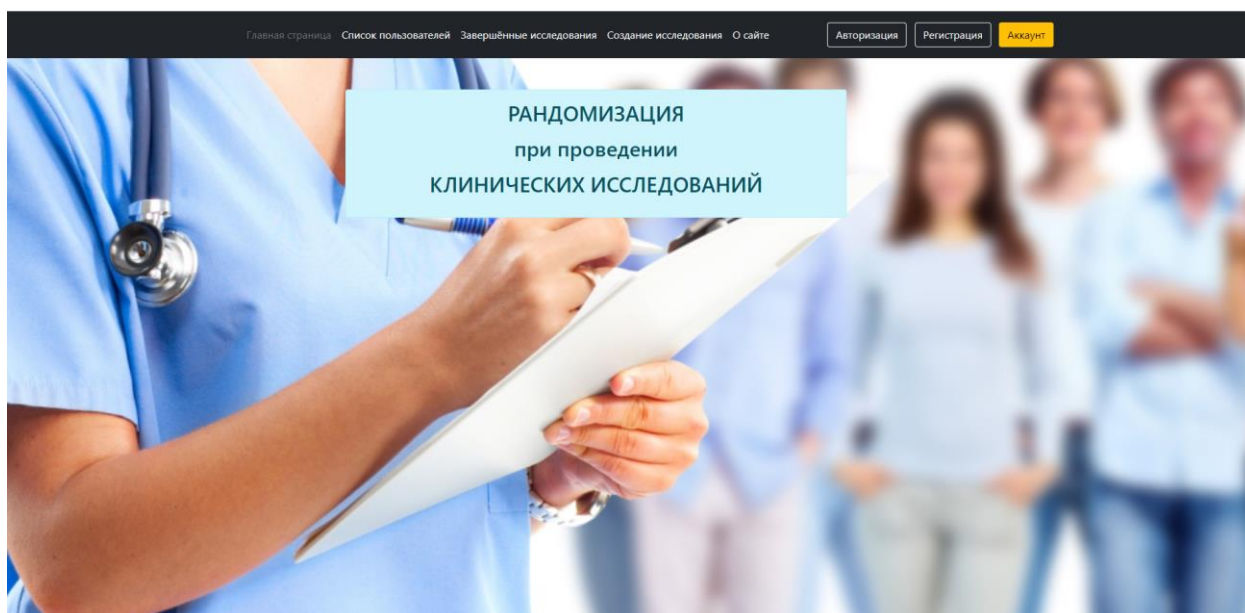


Рисунок 17. Интерфейс главной страницы

Кнопка «Аккаунт» при нажатии ничего не делает, однако при наведении на неё курсора появляется выпадающий список с кнопками «Исследования», переводящая пользователя на страницу аккаунта, и «Выйти», выполняющая функцию выхода из профиля (разавторизации).

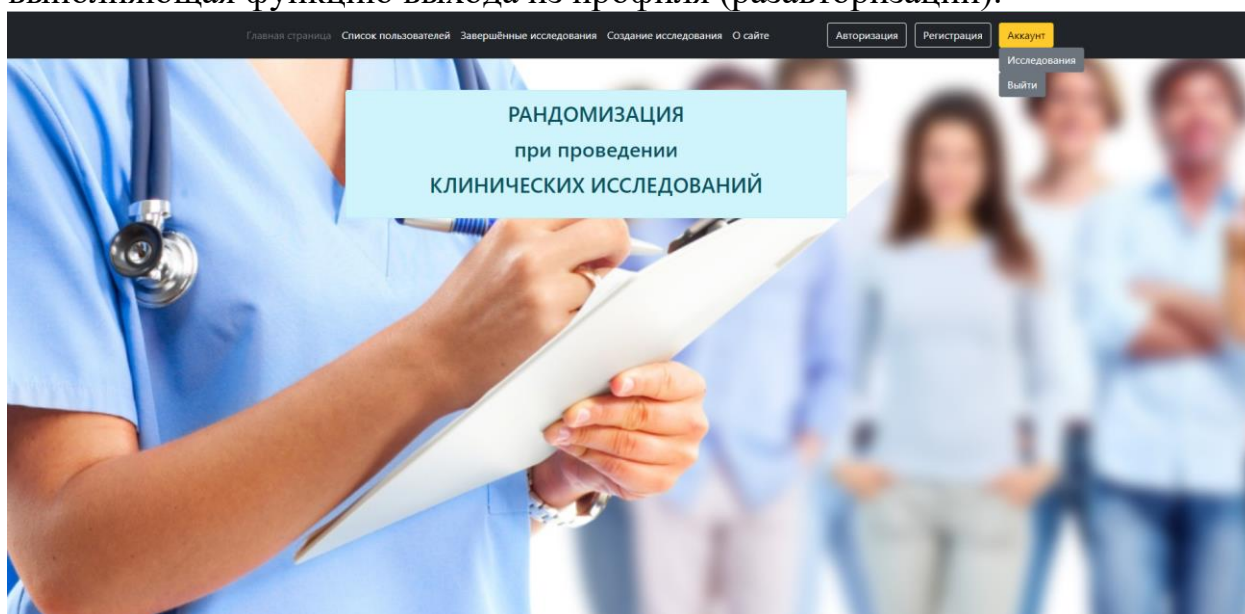


Рисунок 18. Интерфейс главной страницы при наведении курсора на кнопку "Аккаунт"

2.7.2 Интерфейс регистрации нового пользователя

Страница состоит из формы с полями для ввода данных, включающий в себя поле логина пользователя (его имени в системе), пароля и повтора пароля для подтверждения отсутствия опечатки. Последние два поля имеют тип ввода «пароль», поэтому вводимые символы отображаются в виде кружков.

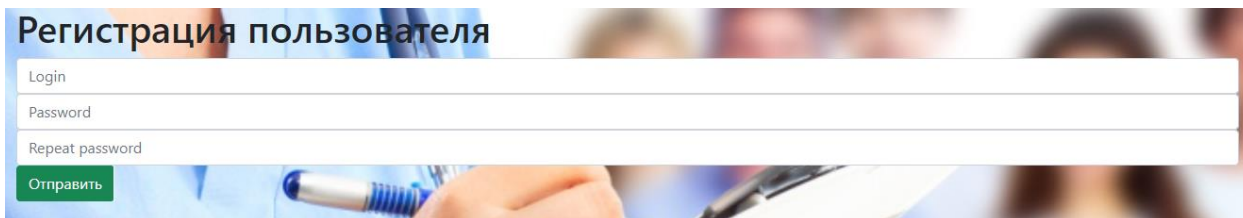


Рисунок 19. Интерфейс формы регистрации пользователя

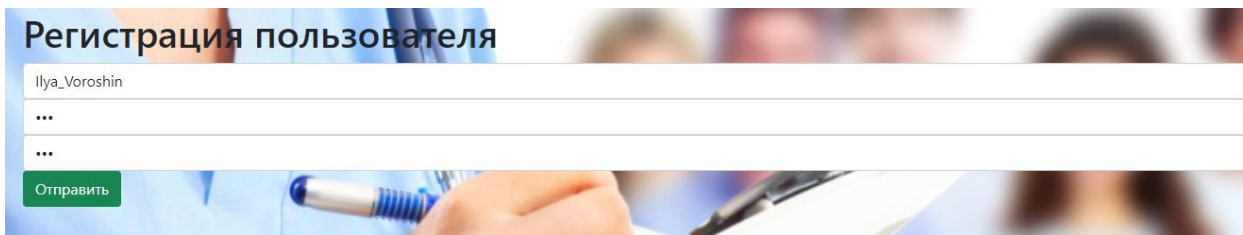


Рисунок 20. Интерфейс при вводе пользователем данных в форму регистрации

Нажатие на кнопку «Отправить», при условии отсутствия исключений, отправляет данные на сервер и добавляет в базу данных нового пользователя.

В случае, если введенный логин уже есть в базе данных, либо поле пароля и повтор пароля не совпадают, либо какое-нибудь из полей не заполнено, пользователю выводится соответствующее предупреждение.

2.7.3 Интерфейс авторизации пользователя

Страница состоит из формы с полями для ввода логина и пароля. Поле с паролем имеет тип «пароль», поэтому вводимые символы отображаются в виде кружков.



Рисунок 21. Интерфейс авторизации пользователя

Нажатие на кнопку «Войти в систему», при условии отсутствия исключений, отправляет данные на сервер и выполняет авторизацию пользователя. Нажатие на кнопку «Регистрация» переводит пользователя на страницу регистрации.

В случае, если введенные данные не совпадают с данными из базы данных, либо не все поля заполнены, пользователю выводится соответствующее предупреждение.

2.7.4 Интерфейс создания исследования

Нажатие кнопки «Создать исследование» на главной странице переводит на страницу с выбором одного из трёх предложенных типов

исследования, состоящего из трёх соответствующих радиокнопок с их описаниями, позволяющих выбрать одну опцию из предложенного набора.

The screenshot shows a web interface for selecting a randomization type. At the top, there is a navigation bar with links: Главная страница, Список пользователей, Завершённые исследования, Создание исследования, О сайте. On the right of the navigation bar are buttons: Авторизация, Регистрации, and Аккаунт. The main heading is "Выберите тип рандомизации:". Below it are three radio button options, each with a description and a diagram of red and green dots representing participants.

- ☐ Безблочная рандомизация (простая)
Простая (обычная) рандомизация создаёт последовательность случайных значений. Наглядный пример – подбрасывание монетки, определяющей, в какую группу пойдёт испытуемый.
[Diagram: A 3x4 grid of 12 dots, alternating red and green in a checkerboard pattern.]
- ☐ Блочная с фиксированным размером блоков
Формирует блоки одинаковых размеров, кратных сумме методов исследования. В этих блоках и проводится рандомизация, по одинаковому количеству пациентов с тем или иным вмешательством.
[Diagram: A 3x4 grid of 12 dots, grouped into 3 blocks of 4 dots each, alternating red and green.]
- ☐ Блочная со случайными размерами блоков
Также, как и в блочной фиксированной, рандомизация проводится в блоках, но блоки генерируются со случайным размером. Максимальный размер блока задаёт создатель исследования.
[Diagram: A 3x4 grid of 12 dots, grouped into 3 blocks of 4 dots each, alternating red and green.]

At the bottom left of the selection area is a green button labeled "Далее".

Рисунок 22. Интерфейс выбора типа рандомизации при создании исследования

Если тип не был выбран - выводится соответствующее предупреждение, а страница обновляется. Нажатие на кнопку «Далее» после выбора типа рандомизации осуществляет переход на следующую страницу создания исследования, в котором отображается выбор с предыдущей страницы и предлагается ввод данных создаваемого исследования.

The screenshot shows a web interface for creating a research study. The title is "Создание исследования" and the subtitle is "Безблочная рандомизация". Below the title are three input fields: "Название:" with a placeholder "title", "Количество вмешательств:" with a placeholder "number_of_interventions", and "Количество испытуемых:" with a placeholder "Number of participants". At the bottom left is a green button labeled "Создать". To the right of the "Создать" button is a label "Наименование вмешательства:".

Рисунок 23. Интерфейс формы создания исследования при выборе простой (безблочной) рандомизации

В зависимости от выбора типа исследования, поля интерфейса меняются в соответствии с логикой создания исследования с выбранным типом рандомизации.

Создание исследования
Блочная рандомизация с фиксированным размером блоков

Название:
title

Количество вмешательств:
number_of_interventions

Количество испытуемых:
Number of participants

Размер блока:
block_size

Создать Наименование вмешательства:

Рисунок 24. Интерфейс формы создания исследования при выборе блочной рандомизации с фиксированным размером блоков

Создание исследования
Блочная рандомизация со случайными размерами блоков

Название:
title

Количество вмешательств:
number_of_interventions

Количество испытуемых:
Number of participants

Максимальный размер блока:
max_block_size

Создать Наименование вмешательства:

Рисунок 25. Интерфейс формы создания исследования при выборе блочной рандомизации со случайными размерами блоков

Поля «Количество вмешательств», «количество испытуемых», «размер блока» и «максимальный размер блока» допускают только числовой тип ввода.

Поля для ввода названий вмешательств создаются динамически при изменении поля количества вмешательств.

Создание исследования
Блочная рандомизация со случайными размерами блоков

Название:
Исследование 2

Количество вмешательств:
3

Количество испытуемых:
15

Максимальный размер блока:
6

Создать Наименование вмешательства:

Рисунок 26. Интерфейс добавления в форму создания исследования дополнительных блоков ввода названий вмешательств при изменении поля "Количество вмешательств"

Нажатие на кнопку «Создать» создаёт исследование и перебрасывает пользователя на страницу аккаунта.

В случае, если не все поля заполнены, либо какое-то из полей названия вмешательства совпадает с любым другим, выводится соответствующее предупреждение.

2.7.5 Интерфейс аккаунта пользователя

Страница включает в себя два списка с исследованиями пользователя: активные исследования и завершённые. Каждое исследование включает в себя блок с информацией о его параметрах.



Рисунок 27. Интерфейс страницы личного кабинета пользователя

У активных исследований есть кнопки «Удалить» для полного удаления исследования до его завершения и «Редактировать исследование» - кнопка-ссылка, осуществляющая переход на страницу редактирования исследования.

Завершённые исследования имеют кнопки «Просмотр лога» для перехода на страницу с просмотром лога исследования, «Просмотр пациентов» для перехода на страницу с просмотром списка пациентов с их вмешательством, «Просмотр таблиц рандомизации» для перехода на страницу со схемой распределения пациентов на группы.

2.7.6 Интерфейс редактирования исследования

Страница начинается с дублирования информации об исследовании. Далее идут кнопки «Добавить пациента» и «Завершить исследование». Ниже имеется список уже добавленных в исследование пациентов и их вмешательств.

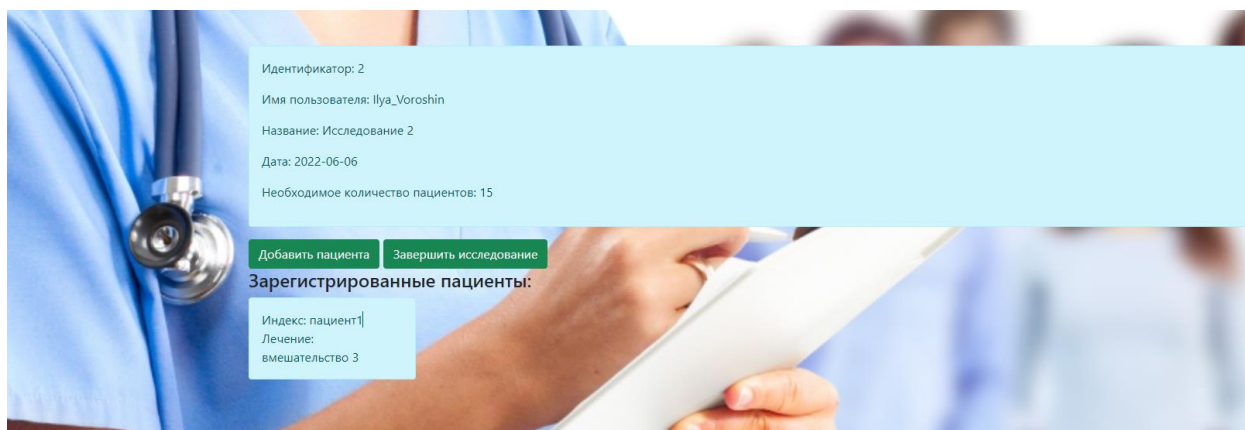


Рисунок 28. Интерфейс страницы редактирования исследования

При нажатии на кнопку «Добавить пациента» появляется всплывающая форма для ввода индекса пациента. Кнопка «Добавить» добавляет в исследование нового пациента с заданным индексом, а кнопка «Закрыть» - закрывает форму. При добавлении пациента всплывает уведомление, подтверждающее, что добавление нового пациента прошло успешно, а также обновляется список зарегистрированных пациентов.

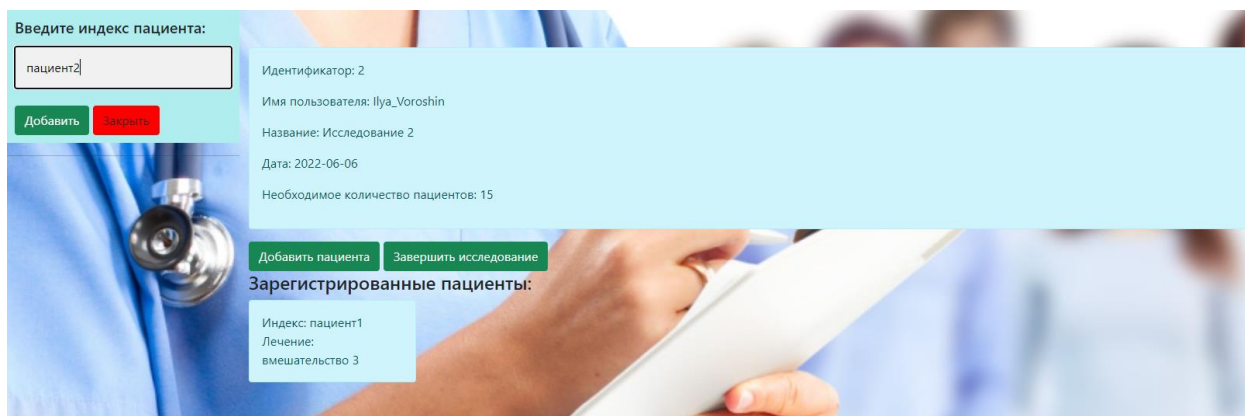


Рисунок 29. Интерфейс отображения скрытого блока ввода для добавления в исследование нового пациента при нажатии кнопки "Добавить пациента"

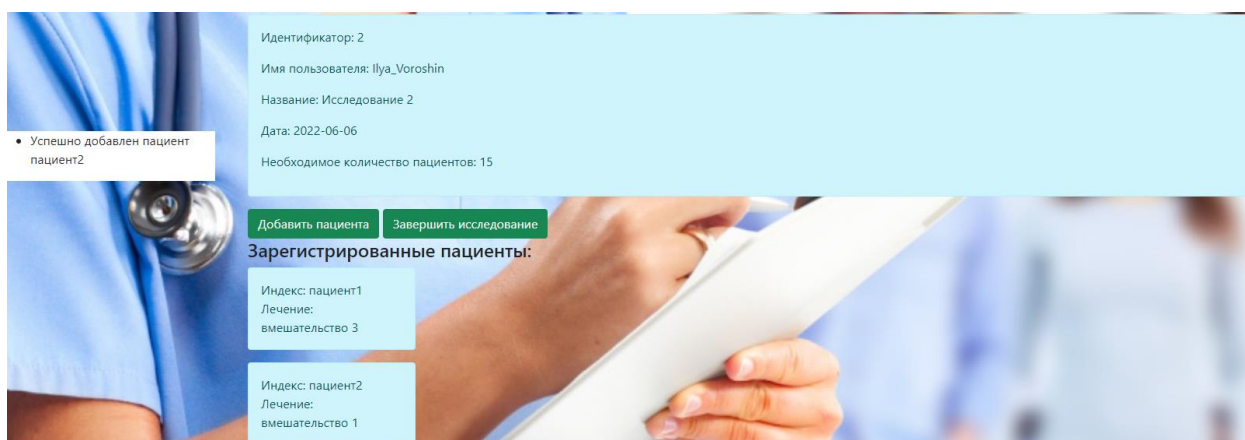


Рисунок 30. Интерфейс страницы редактирования исследования с подтверждением успешного добавления в него нового пациента

2.7.7 Интерфейс просмотра лога

Выводится список производимых с исследованием операций в виде блоков со всей записанной информацией.

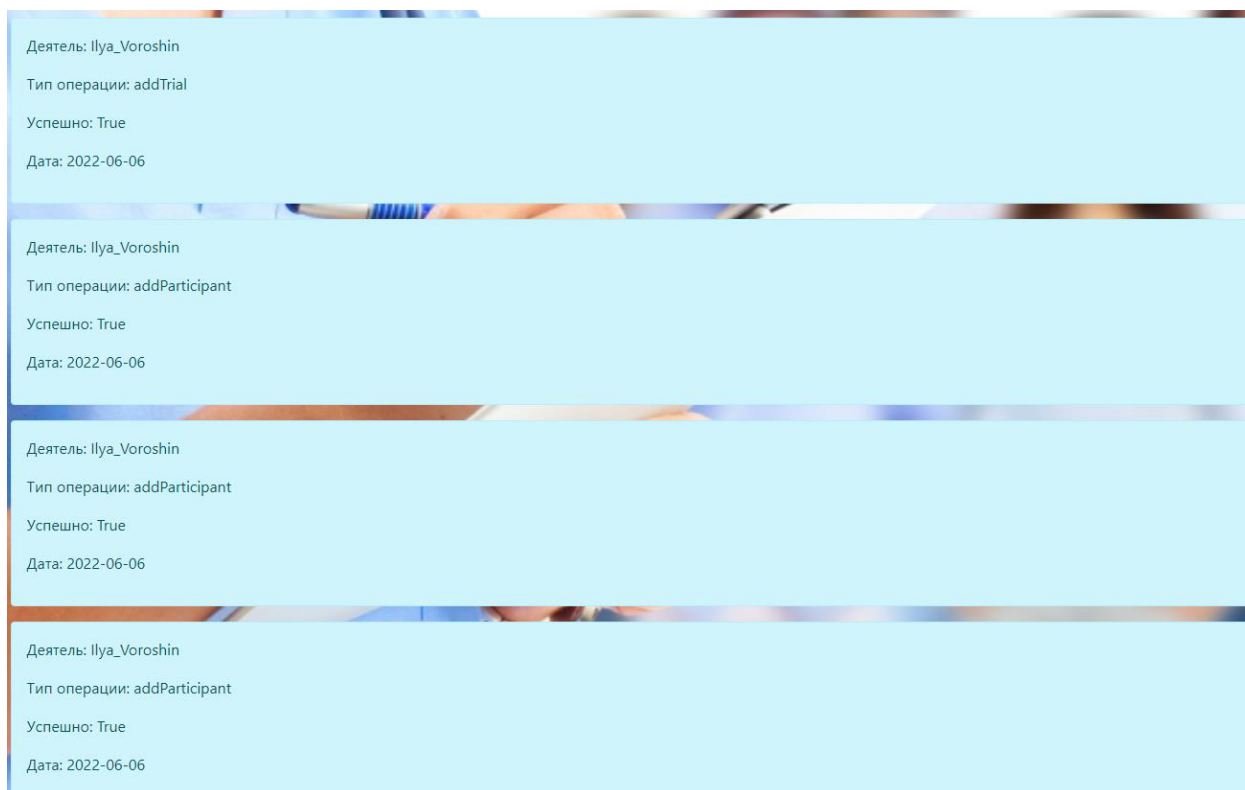


Рисунок 31. Интерфейс просмотра лога исследования

2.7.8 Интерфейс просмотра пациентов

Выводится список с включенными в исследование пациентами и их вмешательством.

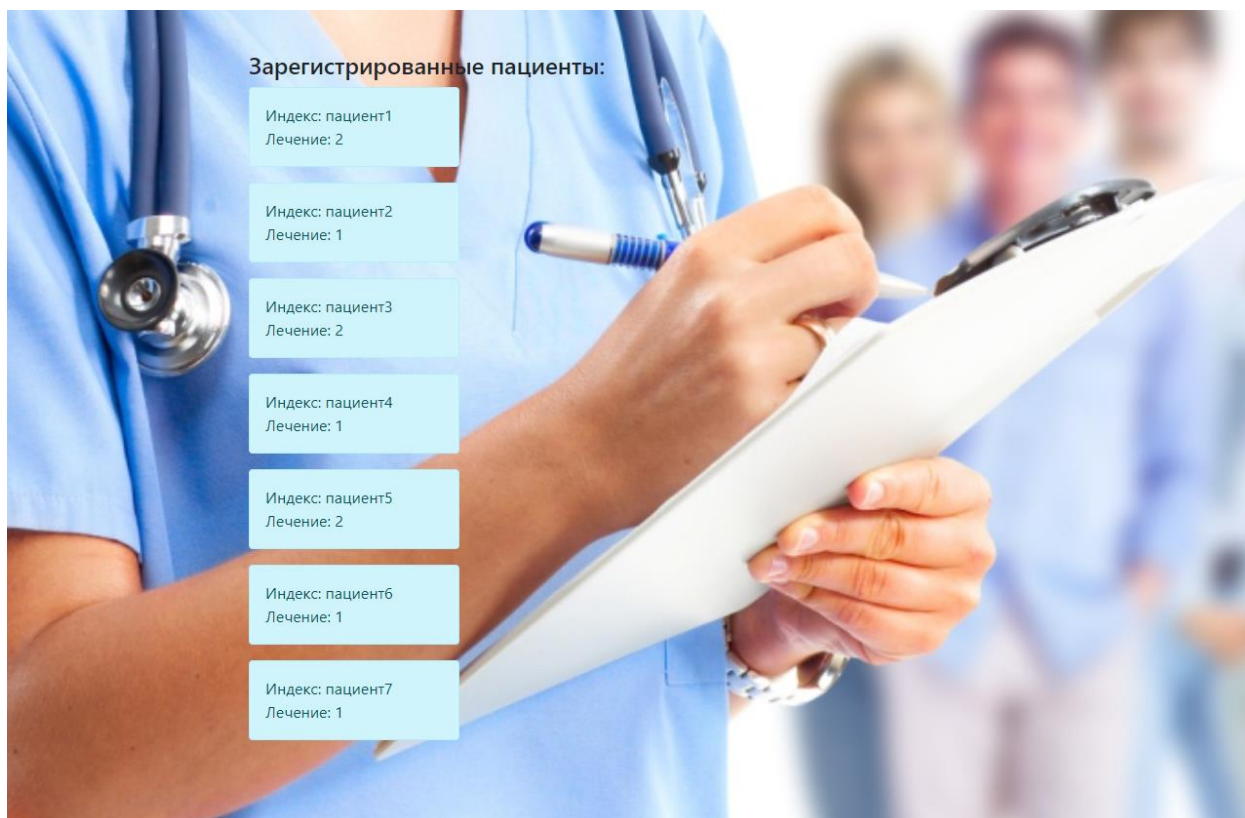


Рисунок 32. Интерфейс просмотра списка пациентов исследования

2.7.9 Интерфейс просмотра таблиц рандомизации

Выводится список со схемой распределения пациентов в группы в виде блоков со всей информацией.

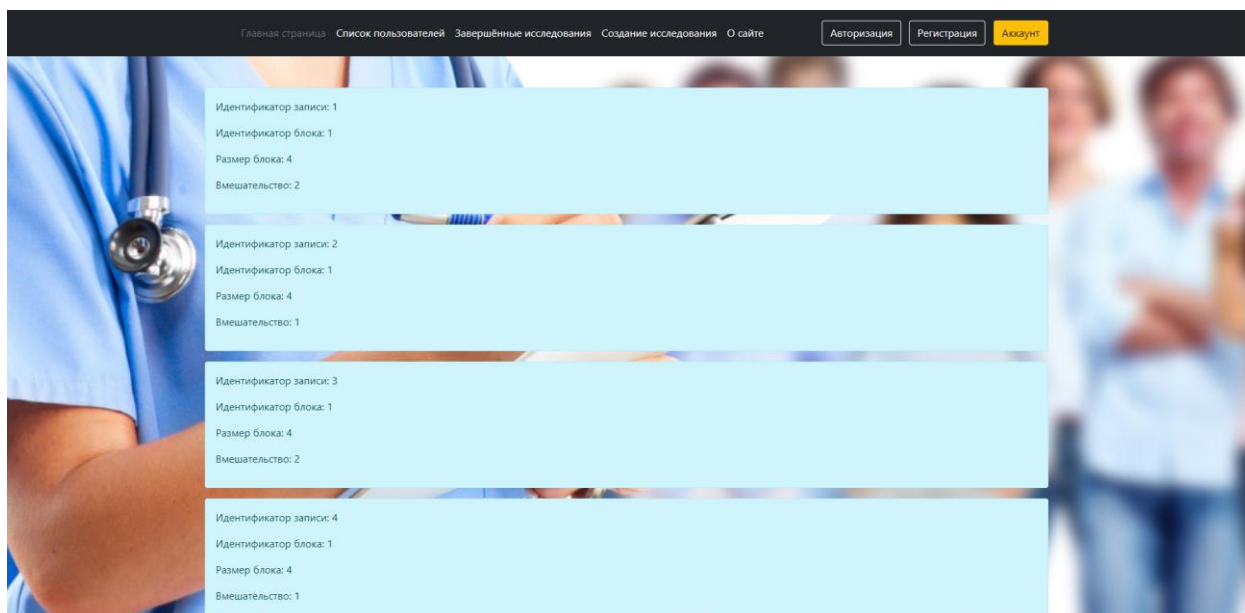


Рисунок 33. Интерфейс просмотра схемы распределения пациентов на группы исследования

2.7.10 Интерфейс просмотра завершённых исследований

Выводит список всех исследований всех пользователей. Интерфейс такой же, как и в списке завершённых исследований на странице аккаунта пользователя.

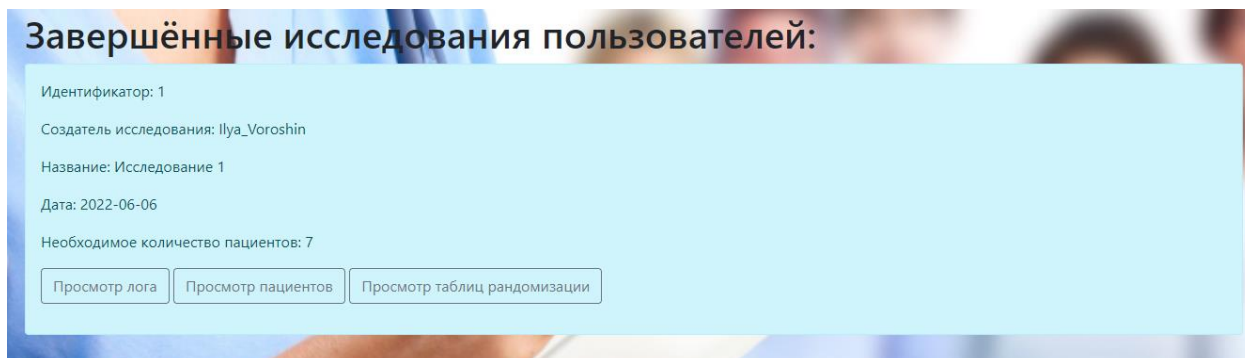


Рисунок 34. Интерфейс просмотра проведённых на сервисе исследований

2.7.11 Интерфейс просмотра списка пользователей

Выводит на экран список зарегистрированных пользователей в виде: (id пользователя) (логин пользователя).

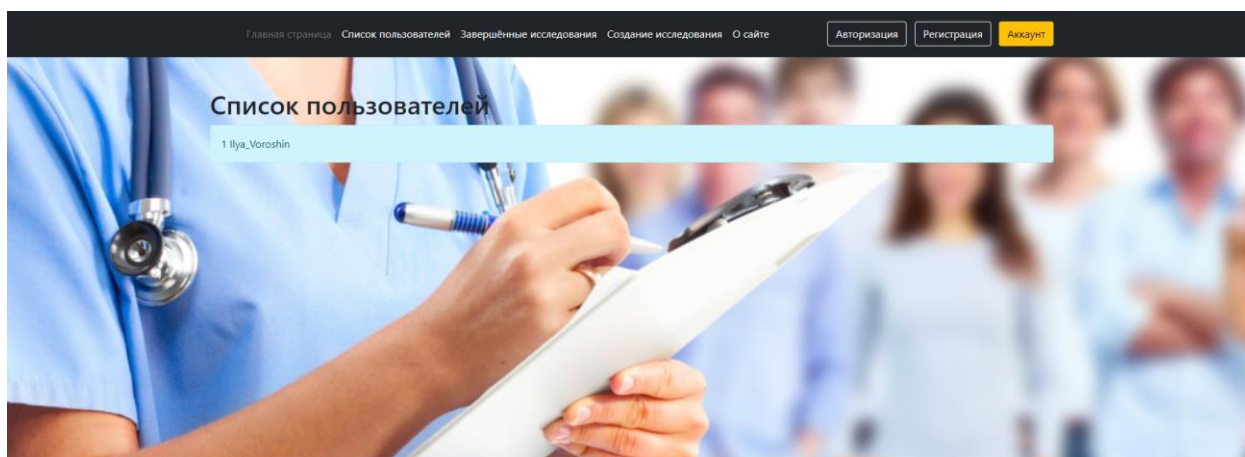


Рисунок 35. Интерфейс просмотра зарегистрированных пользователей

2.7.12 Интерфейс страницы «О сайте»

Отображает небольшой текст о предназначении сервиса.

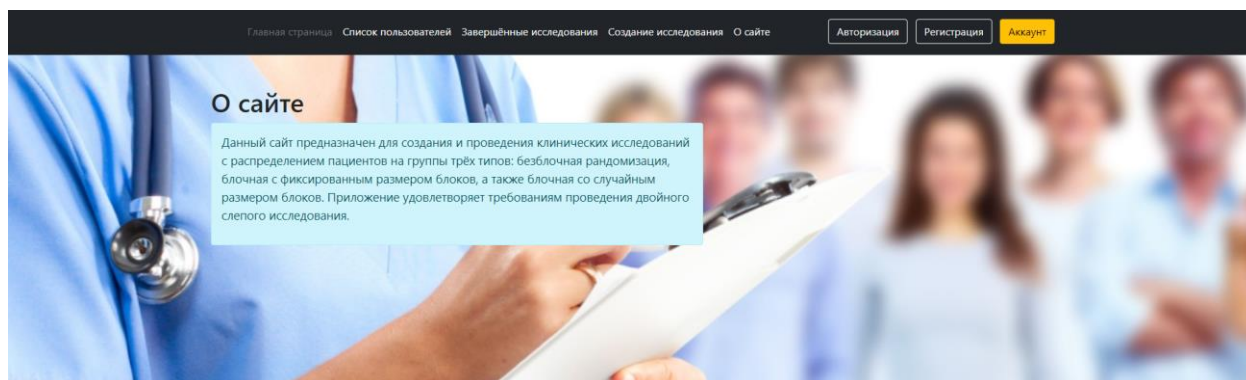


Рисунок 36.Интерфейс страницы "О сайте"

ЗАКЛЮЧЕНИЕ

В проделанной дипломной работе было создано функциональное веб-приложение по созданию клинических исследований. При создании доступны на выбор создателя три вида рандомизации распределения пациентов на группы вмешательств: простая (безблочная), блочная с фиксированным размером блоков, блочная со случайным размером блоков.

Тип вмешательства испытуемых нельзя предугадать заранее (принадлежность пациента к группе определяется сразу при его добавлении в исследование, изменить группу никак нельзя, как и исключить испытуемого из проекта), схемы распределения доступны для просмотра лишь после конечного завершения исследования.

Создание исследований доступно лишь авторизованным пользователям. У неавторизованных пользователей сохраняется возможность просмотра проведённых на сайте исследований. То есть создатель, после успешного завершения исследования, может поделиться ссылкой на исследование для просмотра его результатов, журнала действий и схемы распределения пациентов, чтобы обеспечить уверенностью, что не было никакой подтасовки испытуемых в целях фальсификации.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Рябко Б. Я., Фионов А. Н. Криптографические методы защиты информации: учебное пособие для вузов. М.: Горячая линия–Телеком, 2005. 229 с.
2. <https://trialsjournal.biomedcentral.com/articles/10.1186/1745-6215-13-90>
3. <https://bmcmmedresmethodol.biomedcentral.com/articles/10.1186/s12874-021-01303-z>
4. https://bookdown.org/pdr_higgins/rmrwr/randomization-for-clinical-trials-with-r.html
5. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3136079/>
6. Niederreiter H. Random Number Generation and Quasi-Monte Carlo Methods. – SIAM, 1992. – 247 p
https://www.ricam.oeaw.ac.at/files/people/siambook_nied.pdf
7. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6547231/>
8. Бейдер Д. Чистый Python. Тонкости программирования для профи. — СПб.: Питер, 2018. — 288 с
9. Aggarwal, Shalabh. Flask Framework Cookbook. 2nd ed. Packt Publishing, 2019. 302p.
10. <https://htmlacademy.ru/>
11. <https://getbootstrap.com/>