**Министерство науки и высшего образования Российской Федерации**

**федеральное государственное автономное образовательное учреждение высшего образования**

**“НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО” (УНИВЕРСИТЕТ ИТМО)**

**ЦЕНТР АВТОРИЗОВАННОГО ОБУЧЕНИЯ ИНФОРМАЦИОННЫМ ТЕХНОЛОГИЯМ**

**ИТОГОВАЯ АТТЕСТАЦИОННАЯ РАБОТА**

РАЗРАБОТКА ПРИЛОЖЕНИЯ АРХИВАЦИИ БИОМАТЕРИАЛА ДЛЯ МЕДИЦИНСКОЙ ЛАБОРАТОРИИ

Автор Белозеров Павел Валерьевич \_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия Имя Отчество) (Подпись)

**Центр авторизованного обучения информационным технологиям**

Наименование программы **«Python-разработчик»**

Руководитель ст. преподаватель Кузьмин Константин Михайлович \_\_\_\_\_\_\_\_\_\_\_\_

(Подпись)

**К защите допустить**

|  |  |  |
| --- | --- | --- |
| *Заместитель директора ЦАО ИТ,*  *к.т.н.* |  | / *Т.В. Зудилова*/ |

Санкт-Петербург, 2023г.

Обучающийся Белозеров Павел Валерьевич Группа 124/29

(Фамилия, И. О.)

Работа принята «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2023г.

Работа выполнена с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата защиты «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2023г.

Секретарь ИАК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия, И. О.) (подпись)

Листов хранения \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Демонстрационных материалов \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Содержание**

|  |  |
| --- | --- |
| 1. Введение | 4 |
| 1. Цель | 5 |
| 1. Описание системы | 7 |
| * 1. Технические и программные средства, используемые для реализации приложения. | 7 |
| 1. Состав и структура системы | 9 |
| * 1. Концепция. | 9 |
| * 1. Пользователи программной системы | 9 |
| * 1. Общая структура. | 10 |
| * 1. Диаграмма последовательностей | 10 |
| 1. Руководство пользователя | 12 |
| * 1. Пользователь группы Management | 13 |
| * 1. Пользователь группы Laboratory | 17 |
| * 1. Пользователь группы View | 23 |
| * 1. Пользователь группы Cleaning | 24 |
| Функции системы | 26 |
| * 1. User Story | 26 |
| * 1. Функциональные требования | 26 |
| * 1. Функциональная модель | 29 |
| Структура данных | 32 |
| 1. Тестирование | 34 |
| 1. Развертывание | 37 |
| 1. Заключение | 38 |
| 1. Список литературы | 39 |

# Введение

Темой аттестационной работы является разработка приложения архивации биоматериала для медицинской лаборатории.

Актуальность темы данной работы определяется тем, что информатизация и автоматизация производства являются на сегодняшний день одними из наиболее важных тенденций мирового развития. Развитие масштабов информатизации и автоматизации обусловлено научно-техническим прогрессом и современными потребностями предприятий.

В реалиях современного лабораторного производства очень важен процесс совершенствования существующих и создания новых программных продуктов, способных обеспечить растущие потребности бизнеса в условиях постоянного развития.

В связи с этим постоянно требуется разработка/совершенствование небольших приложений «не коробочного» типа под конкретные бизнес-задачи.

Реализация программы архивации - web приложение, что обуславливается следующими критериями:

* Экономия;
* Безопасность;
* Возможность доступа с разных устройств;
* Отсутствие клиентского ПО;
* Масштабируемость;
* Web приложение легко при поддерживать;

В условиях лабораторного производства при наличии рабочих мест в различных частях лабораторного комплекса web приложение наиболее отвечает требованиям гибкости и удобства при архивации биоматериала.

Рассмотрены программные средства, которые понадобились для разработки приложения:

* Язык программирования – Python;
* Веб-фреймворк Django;
* База данных - MySQL;
* Редактор исходного кода - Visual Studio Code;
* HTML5, CSS3, Twitter Bootstrap 5, JavaScript;

.

# Цель

Целью данной аттестационной работы является разработка приложения архивации биоматериала для медицинской лаборатории для обеспечения потребностей лабораторного производства.

Необходимо:

1. Изучить бизнес процесс архивации биоматериала

Назначение:

* Контейнер с биоматериалом после выполнения исследований необходимо хранить определенный период времени (по требованиям законодательства либо по внутренним требованиям подразделений). Необходимо обеспечить его архивацию на указанный период по заданным параметрам с возможностью нахождения в любой момент времени.
* Существующая система не удовлетворяет новым требованиям к архивации, является не гибкой, громоздкой – состоит из нескольких разрозненных модулей, не обеспечивает архивацию по необходимым параметрам. А также работает на устаревшем программном обеспечении, которое больше не поддерживается.
* Требуется разработать новый гибкий инструмент, с расширенными возможностями с использованием современных технических и программных средств.

Рисунок 2.1 Контейнеры с биоматериалом



2. Разработать приложение архивации биоматериала с учетом требований нормативных документов и Заказчика:

- Обеспечить возможность архивации по настраиваемым параметрам:

* Тип контейнера: контейнер в котором биоматериал поступает в лабораторию.
* Рабочий поток: свойство, позволяющее определять пробы в группы по различным признакам;
* Текущий статус пробы: U (unreceived), S (sorted), C (complete), A (authorized);
* Локация – место выполнения исследования;

Архивация осуществляется в штативы. Тип штатива настраивается по указанным параметрам в зависимости от требований Заказчика. Каждый тип штатива имеет свой цвет для четкой визуальной идентификации. Каждый штатив имеет индивидуальный номер для идентификации в системе. Конфигурация штатива (количество ячеек X, Y) может быть произвольной, в зависимости от требований Заказчика. Каждому типу штатива присваивается срок хранения.

- Обеспечить возможность архивации и поиска по индивидуальному штрих коду.

- Обеспечить вывод информации о необходимости утилизации биоматериала с истекшим сроком хранения.

Дополнительные требования:

- Каждый тип штатива должен являться уникальным;

- Одновременно можно архивировать только в один штатив;

- Нельзя заархивировать один и тот же номер пробы более одного раза;

- После окончания срока хранения штатив может быть очищен.

# Описание системы

* 1. **Технические и программные средства, используемые для реализации приложения.**

При разработке приложения архивации биоматериала для медицинской лаборатории были использованы следующие технические и программные средства:

**Python**  
Python — высокоуровневый язык программирования, построенный на идея императивного, объектно-ориентированного и функционального программирования.

Преимущества Python применимо к текущей задаче:

* скорость и гибкость разработки;
* множество полезных библиотек и расширений языка можно легко использовать в своих проектах благодаря унифицированному механизму импорта и программным интерфейсам;
* механизмы модульности хорошо продуманы и могут быть легко использованы;
* всё в Python является объектами, но при этом объектный подход не навязывается программисту;
* Python предлагает инструменты для простой и функциональной разработки web-приложений.

**Django**

Django — свободный фреймворк для веб-приложений на языке Python, использующий шаблон проектирования MVC. Django содержит огромное количество функциональности для решения большинства задач веб-разработки.

Для работы с базой данных Django использует собственный ORM, в котором модель данных описывается классами Python, и по ней генерируется схема базы данных.

Помимо моделей, шаблонов и представлений, Django предлагает изначально такую функциональность, как конфигурация адресов URL, автоматический интерфейс администрирования базы данных, кэширование и прочее, что значительно облегчает работу web-разработчика. Так же, как и в Python, Django предоставляет совместимость с большой стандартной библиотекой дополнительных пакетов, которые можно использовать в своих приложениях без необходимости дополнительной загрузки чего-либо.

Существенные преимущества при решении задач для данной аттестационной работы:

* Миграции базы данных;
* Аутентификация пользователя;
* Панель администратора;
* Формы;

### Стандартизированная структура, использование библиотек;

* Механизмы предотвращения распространенных атак вроде SQL-инъекций (XSS) и подделки межсайтовых запросов (CSRF);

**MySQL**

**MySQL** — свободная реляционная система управления базами данных (СУБД). Она универсальна и применяется при разработке веб-сайтов, веб-приложений и корпоративных баз данных начального уровня. Этого более чем достаточно для реализации приложения архивации.

Преимущества, которыми был обусловлен выбор базы данных для данной работы:

* Высокая скорость работы. Производительность;
* Поддержка практически всех CMS;
* Бесплатная лицензия;
* Надежная и простая система безопасности;

**Visual Studio Code**

Visual Studio Code (VS Code) — это редактор кода для разных языков программирования. Он относительно немного весит, гибкий и удобный. В нем можно писать, форматировать и редактировать код на разных языках.

Преимущества, которыми был обусловлен выбор данного редактора кода для аттестационной работы:

* Единый редактор для многих языков (Python, JavaScript, HTML, CSS, JSON и т.д.);
* Разделение на окна;
* Контроль версий;
* Малый вес;
* Гибкость;

**Bootstrap 5**

Bootstrap - один из самых популярных инструментов, который используется при создании сайтов и веб-приложений.

Bootstrap включает в себя множество разных компонентов для веб-сайтов: типографику, веб-формы, кнопки, блоки навигации и другое.

Преимущества Bootstrap, которыми было обусловлено его использование в работе:

* Уменьшение количества времени, затрачиваемого на разработку
* **Адаптивность**
* **Кросс-браузерность**
* **Легкость в использовании и быстрота в освоении**
* **Единство стилей**

1. **Состав и структура системы**
   1. **Концепция.**

**Для** сотрудников лаборатории, **которым** необходимо архивировать биоматериал и при необходимости осуществлять поиск, **данная** программа архивации **является** комплексной системой, **которая** позволит производить архивацию биоматериала в зависимости от гибко настраиваемых параметров: рабочий поток, текущий статус пробы, тип контейнера, локация. Данная система так же позволит создавать штативы необходимой конфигурации, устанавливать сроки хранения. Будет знать местоположение каждого контейнера с биоматериалом. Так же система будет выводить информацию о необходимости сброса (утилизации) штативов с истекшим сроком хранения. **В отличие** от действующей (более не поддерживаемой) системы архивации **наш продукт** будет обладать более гибкими настройками параметров архивации, объединит все модули в одно приложение, позволит устанавливать сроки архивации для конкретных штативов, администрировать существующие конфигурации.

* 1. **Пользователи программной системы.**

Таблица 4.2 Пользователи программной системы

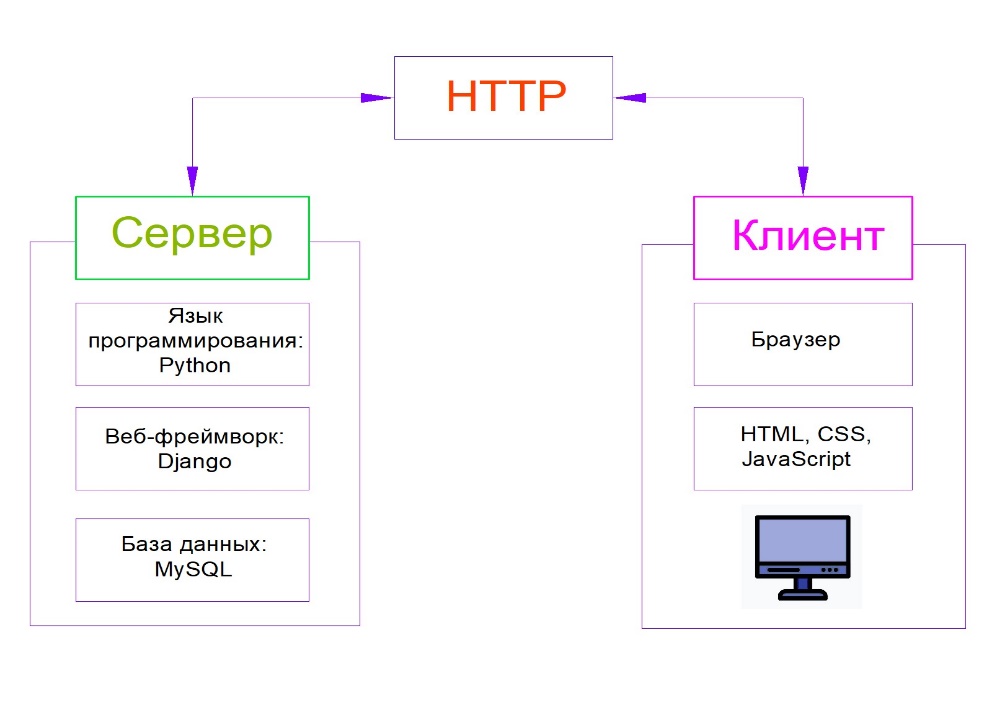
|  |  |  |
| --- | --- | --- |
| Имя | Доступ | Описание |
| Администратор | Привилегированный. Модуль администрирования. | Конфигурации архивных штативов, создание типов штативов на основании настраиваемых параметров по заявкам.  Управление количеством штативов конкретного типа, параметрами штативов.  Установка сроков хранения для типов штативов. |
| Лаборант | Модуль архивации | Непосредственная архивация контейнеров с биоматериалом на местах.  Пользователь последовательно сканирует индивидуальные штрих коды контейнеров с биоматериалом и устанавливает в текущий штатив согласно «сетки» на мониторе приложения. |
| Санитарка | Модуль сброса (утилизации) | Сброс (утилизация) штативов с биоматериалом из архива (места хранение) с истекшим сроком хранения. |
| Сотрудник лаборатории, другой персонал | Модуль поиска | Поиск контейнера с биоматериалом по индивидуальному штрих коду. |

**Практическая значимость** данной работы заключается в следующем:

* Непосредственно, реализация процесса архивации биоматериала с учетом современных тенденций развития технических и программных средств;
* Создание гибкого инструмента с расширенными возможностями;
* Обеспечении выполнения требований нормативных документов;
* Создании базы для дальнейшей оптимизации процесса архивации;
  1. **Общая структура.**

В общем виде структуру web-приложения можно представить следующим образом:

Рисунок 4.3 Структура web-приложения

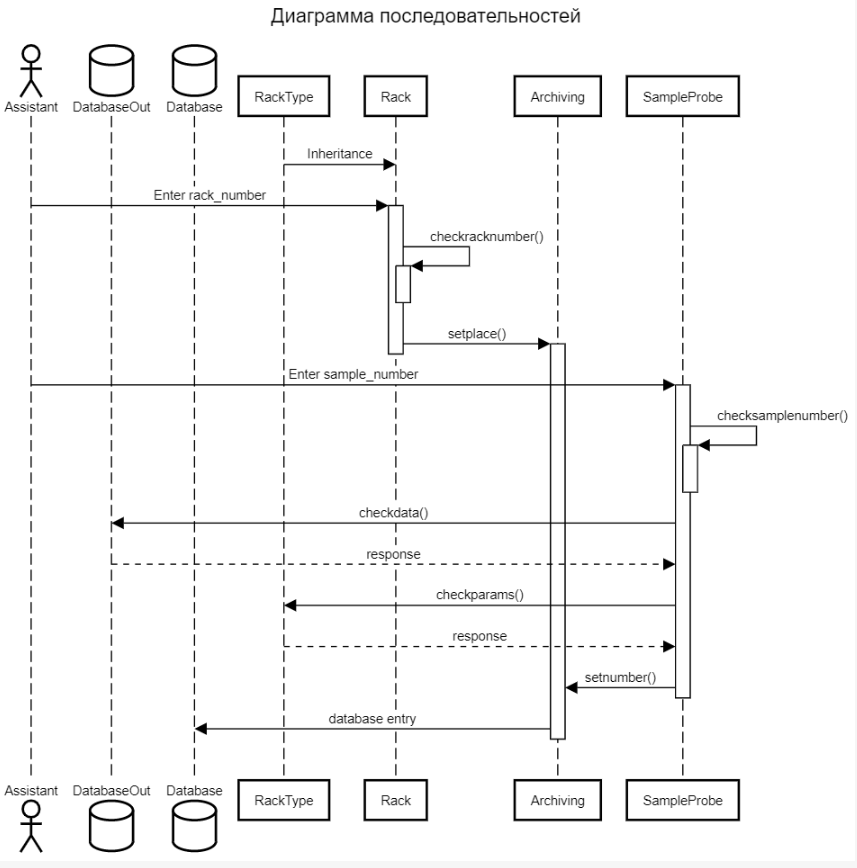
****

**4.4. Диаграмма последовательности** позволяет изобразить поведение нескольких объектов в рамках одного прецедента.

В данном примере демонстрируется диаграмма последовательности при непосредственной архивации:

* обмен сообщениями между объектами;
* ограничения, накладываемые на взаимодействие объектов;
* события, инициирующие взаимодействия объектов;

Рисунок 4.4. Диаграмма последовательностей



1. **Руководство пользователя**

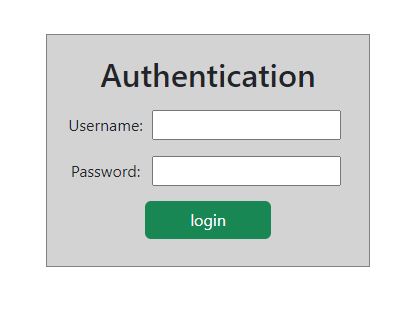
Реализовано 4 группы пользователей в зависимости от выполняемых задач:

* Management
* Laboratory
* View
* Cleaning

Управление группами и добавление новых пользователей осуществляется через панель администрирования Django с использованием Superuser.

Для авторизации необходимо ввести логин и пароль пользователя из соответствующей группы.

Рисунок 5. Страница авторизации

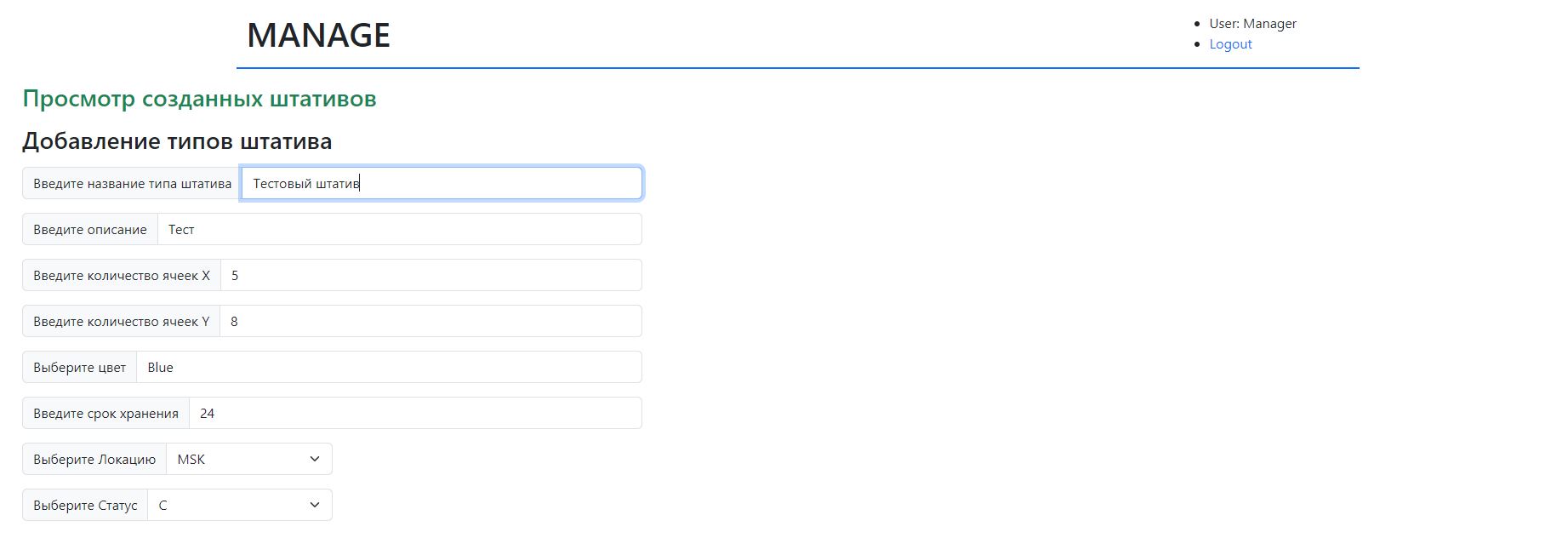


В зависимости от группы, которой принадлежит пользователь, в дальнейшем откроется соответствующий интерфейс.

* 1. **Пользователь группы Management**

При авторизации пользователя, который принадлежит группе Management откроется страница создания и управления типами штативов

Рисунок 5.1.1 Страница создания и управления типами штативов



На данной странице присутствует информация о авторизованном пользователе, а также возможность разавторизации (Logout).

Для создания типа штатива необходимо ввести/выбрать необходимую информацию в указанных полях.

- Название типа штатива

- Описание

- Геометрию штатива: количество ячеек по горизонтали и вертикали

- Цвет штатива

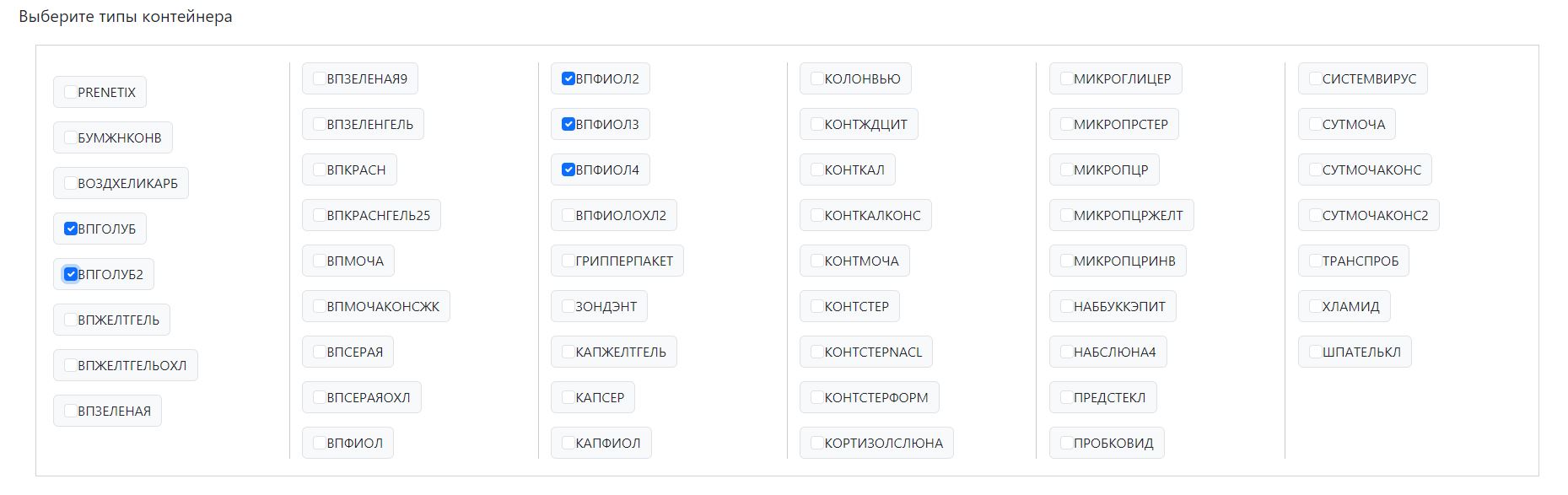
- Срок хранения биоматериала

- Выбрать Локацию из перечня

- Выбрать Статус из перечня

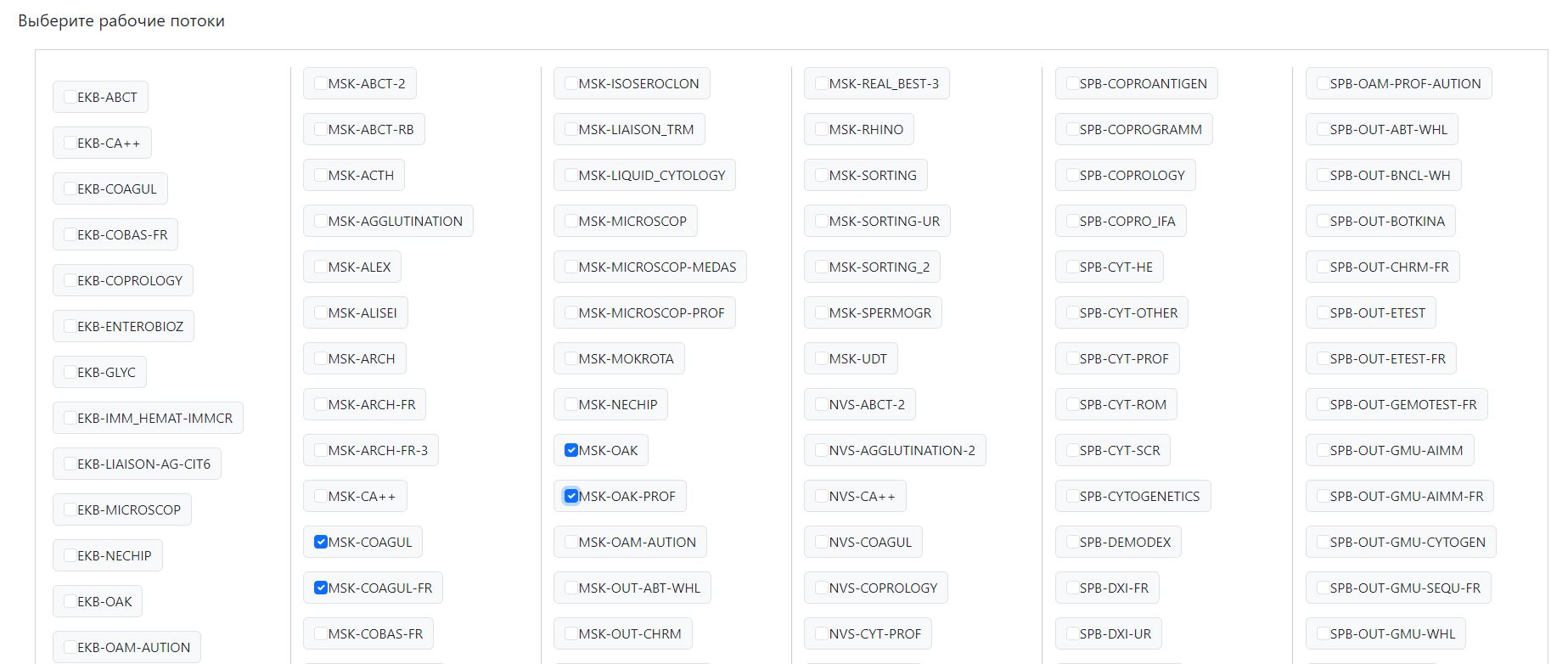
- Выбрать тип контейнера (один или несколько)

Рисунок 5.1.2 Выбор типов контейнера



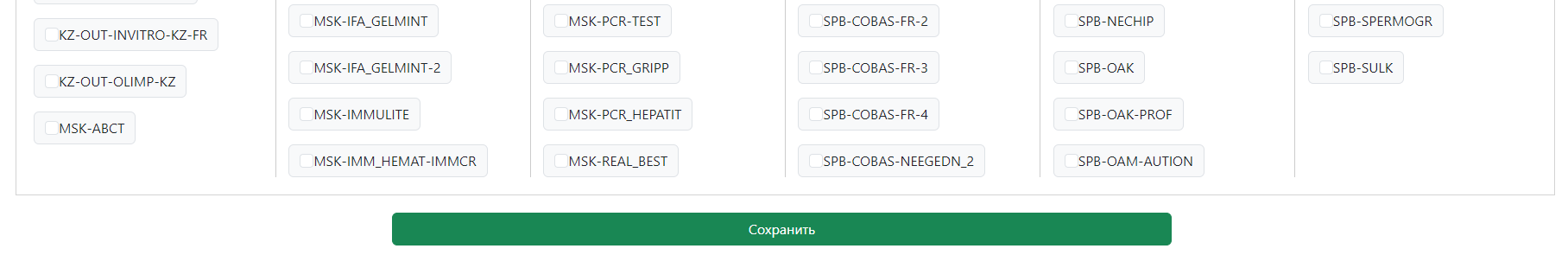
- Выбрать поток (один или несколько)

Рисунок 5.1.3 Выбор рабочих потоков контейнера



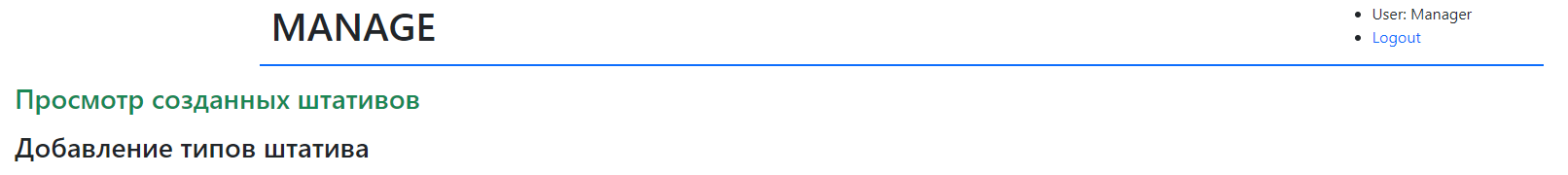
После нажатия кнопки «Сохранить» происходит перемещение на страницу создания штативов

Рисунок 5.1.4 Кнопка «Сохранить»



На страницу создания штативов можно так же переместится в любой момент посредством ссылки «Просмотр созданных штативов»:

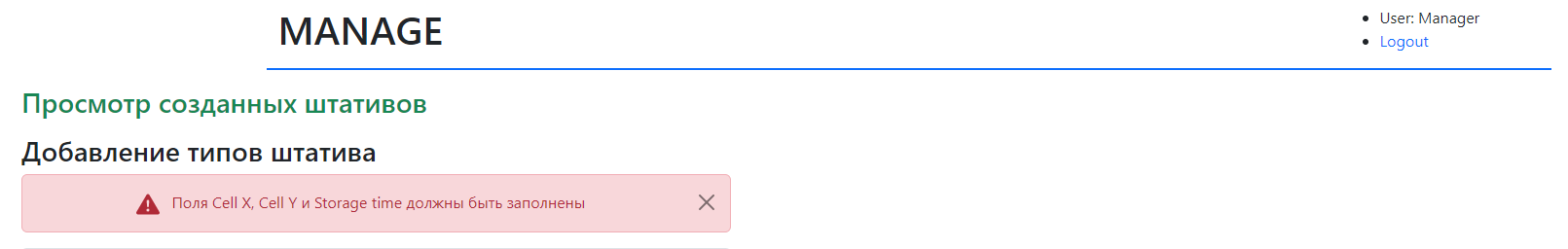
Рисунок 5.1.5 Просмотр созданных штативов



Возможные ошибки:

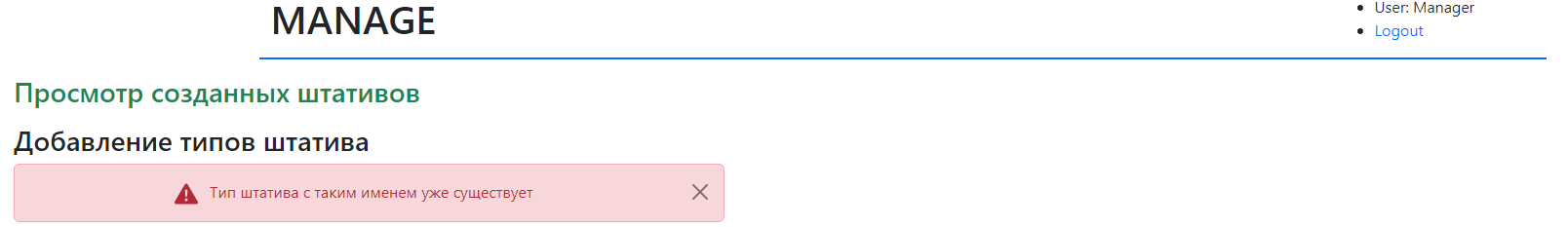
- Если не заполнены некоторые обязательные поля, то при попытке сохранении выйдет сообщение типа alert:

Рисунок 5.1.6 Всплывающее сообщение



- Так же, имя штатива должно быть уникальным, при несоблюдении этого условия при попытке сохранения так же выйдет сообщения типа alert:

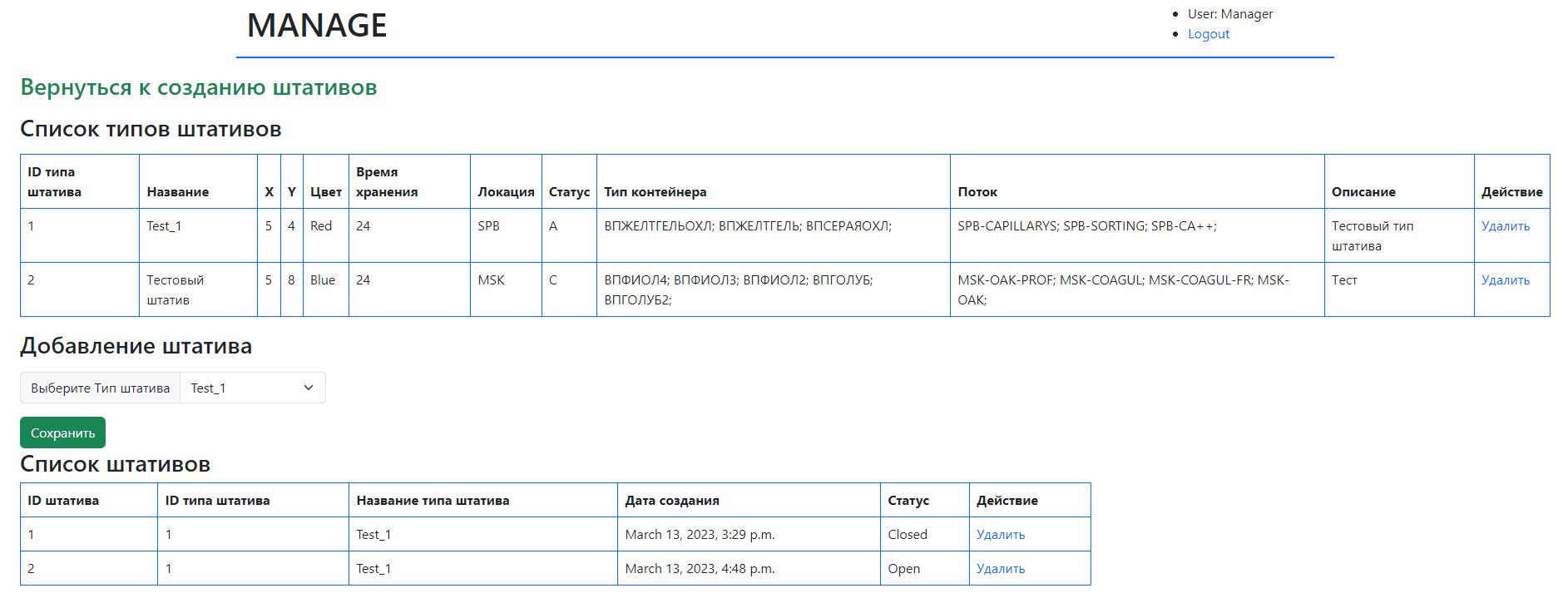
Рисунок 5.1.7 Всплывающее сообщение



**Страница создания штативов**

После создания типа штатива пользователь автоматически переходит на страницу создания штативов

Рисунок 5.1.8 Страница создания штативов



На этой странице выведена информация о созданных типах штативов, а также о созданных штативах.

Добавление штативов осуществляется посредством выбора типа штатива, для которого мы хотим создать штатив и нажатием кнопки «Сохранить»

После этого новый штатив появится в списке «Список штативов»

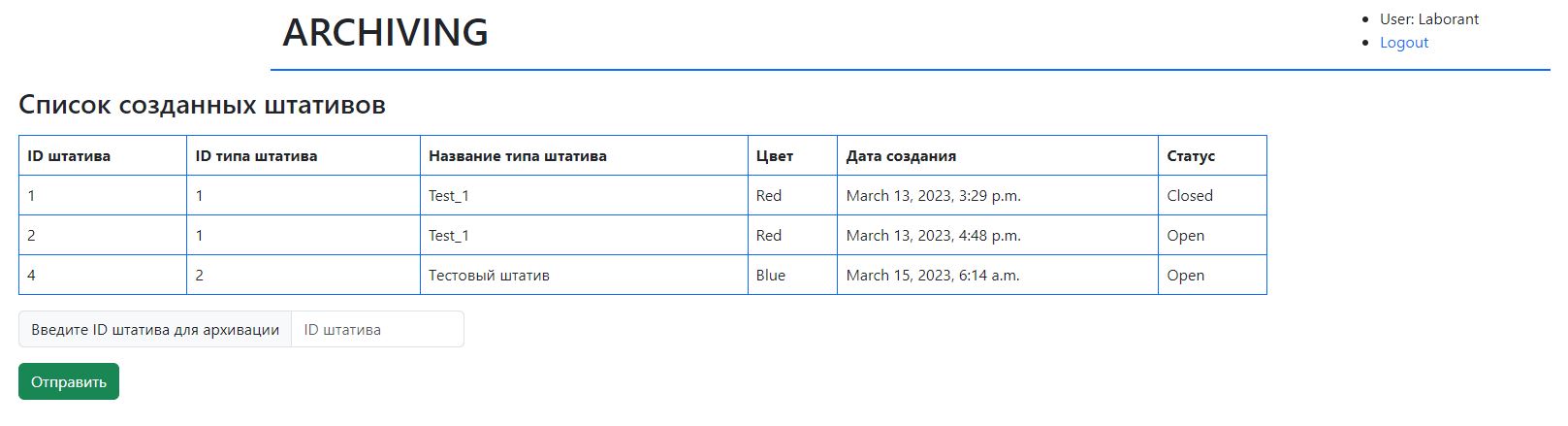
Так же реализована функция удаления конкретного типа штатива или непосредственно штатива (колонка «Действие»)

Посредством ссылки «Вернуться к созданию штативов» пользователь может перейти на предыдущую вкладку.

* 1. **Пользователь группы Laboratory**

При авторизации пользователя, который принадлежит группе Laboratory откроется страница выбора штативов для архивации

Рисунок 5.2.1. Страница выбора штативов для архивации

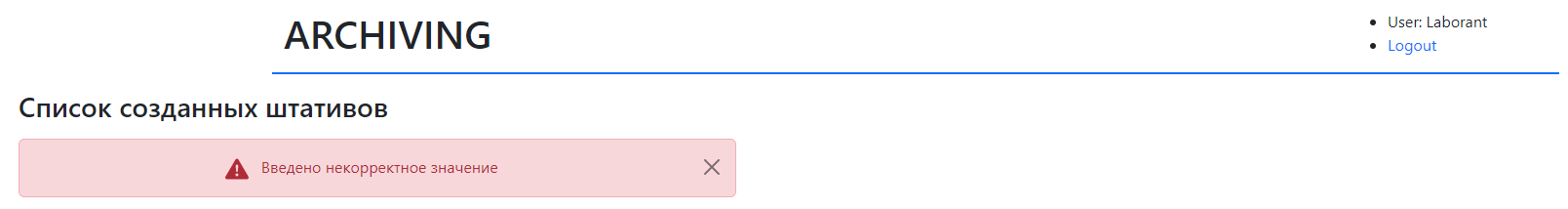


На этой странице присутствует перечень созданных штативов с указанием текущего статуса (Open, In Work или Closed)

Для выбора штатива и перехода к странице непосредственной архивации необходимо ввести ID нужного штатива и нажать кнопку «Отправить»

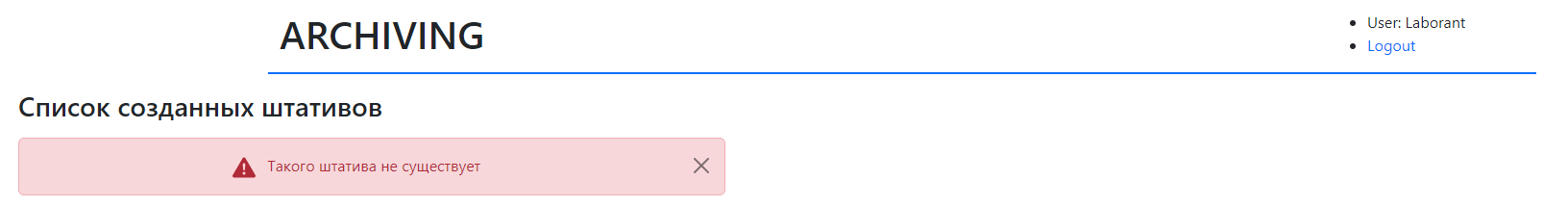
- Если введено некорректное значение, будет вызвано сообщения типа alert:

Рисунок 5.2.2 Всплывающее сообщение



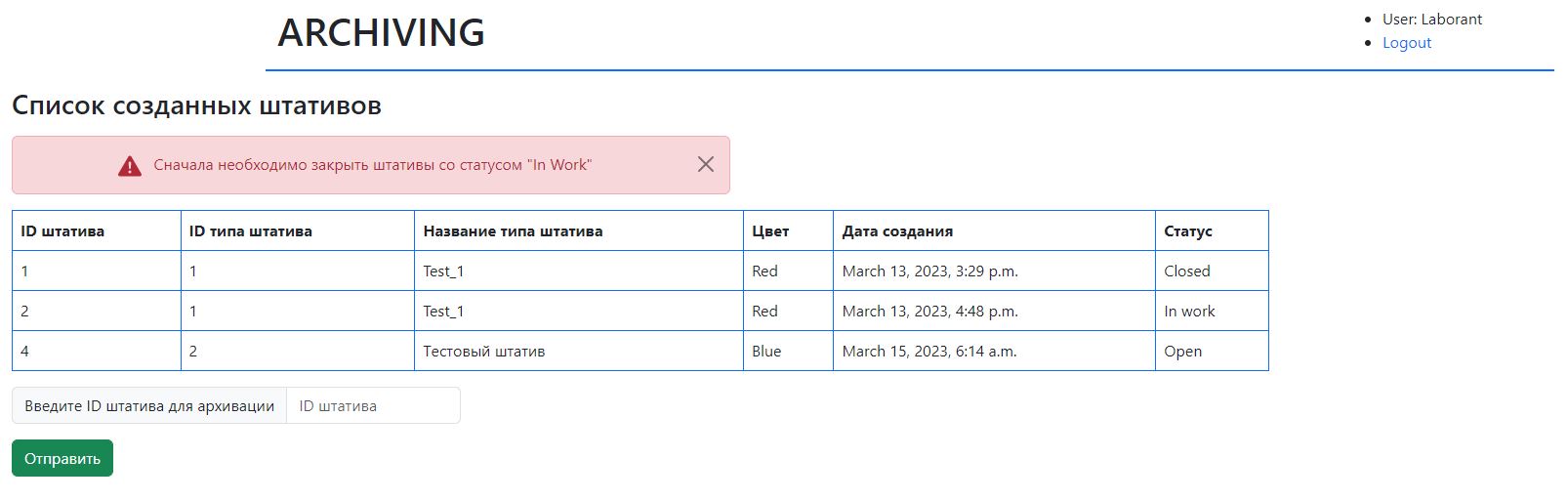
- Если введен ID несуществующего штатива, будет вызвано сообщения типа alert:

Рисунок 5.2.3 Всплывающее сообщение



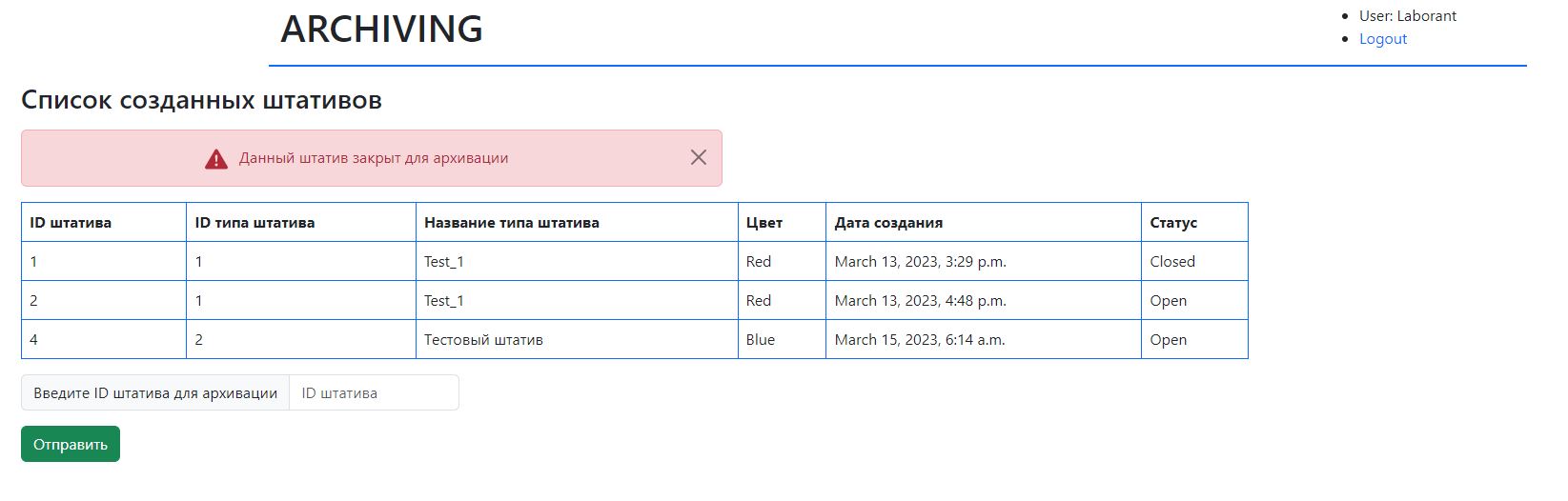
- При попытке ввести ID штатива, со статусом «In Work», будет вызвано сообщение типа alert:

Рисунок 5.2.4 Всплывающее сообщение



– При попытке ввести ID штатива, со статусом «Closed», будет вызвано сообщение типа alert:

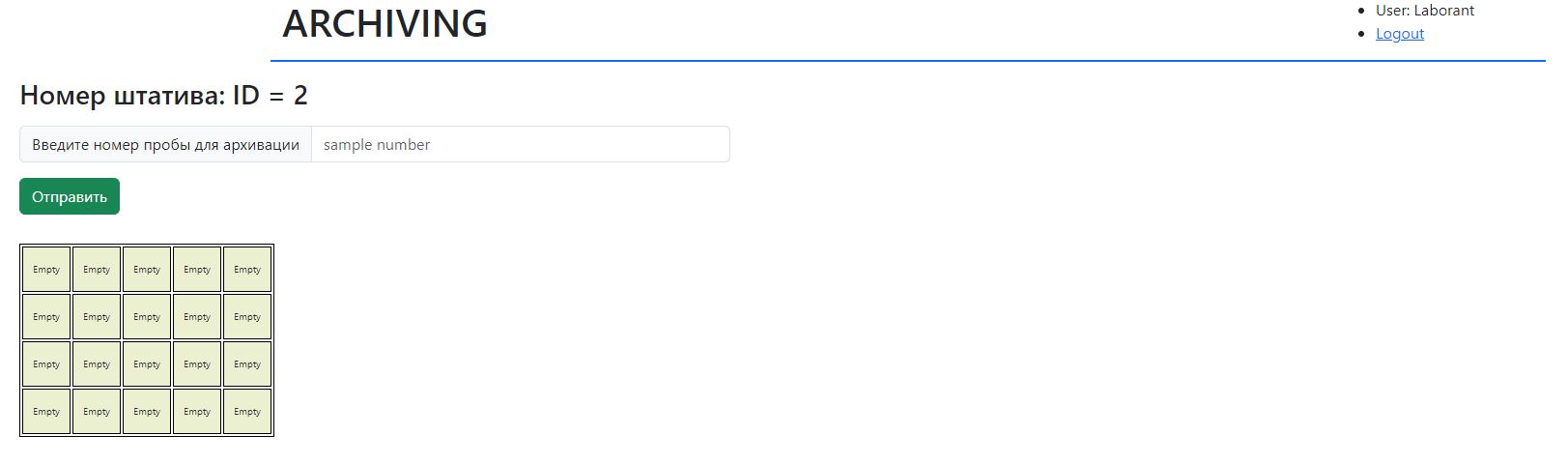
Рисунок 5.2.5 Всплывающее сообщение



После введения ID штатива пользователь перенаправляет на страницу непосредственной архивации. Где представлено поле для ввода номера архивации и условная сетка штатива, которая имитирует реальный «физический» штатив на столе у лаборанта.

Количество ячеек соответствует заданному на этапе создания типа штатива.

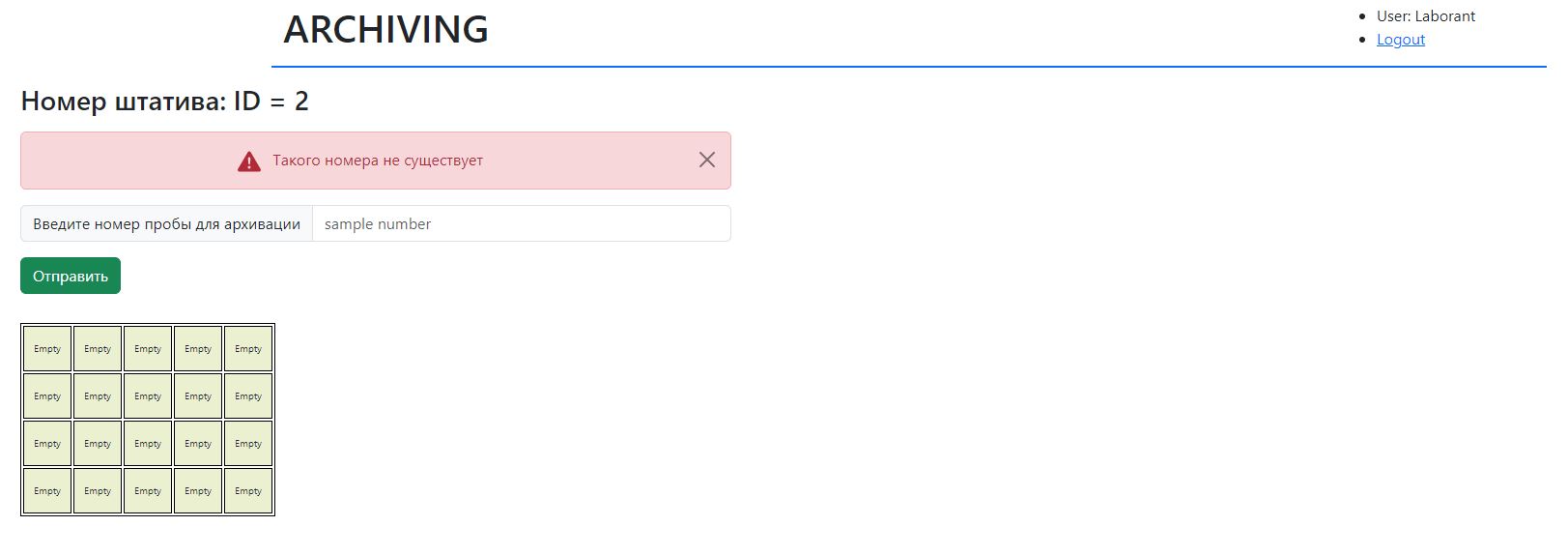
Рисунок 5.2.6 Страница архивации



После ввода происходит поиск по номеру в основной базе данных образцов.

- Если номер не найдет, то будет вызвано сообщение типа alert:

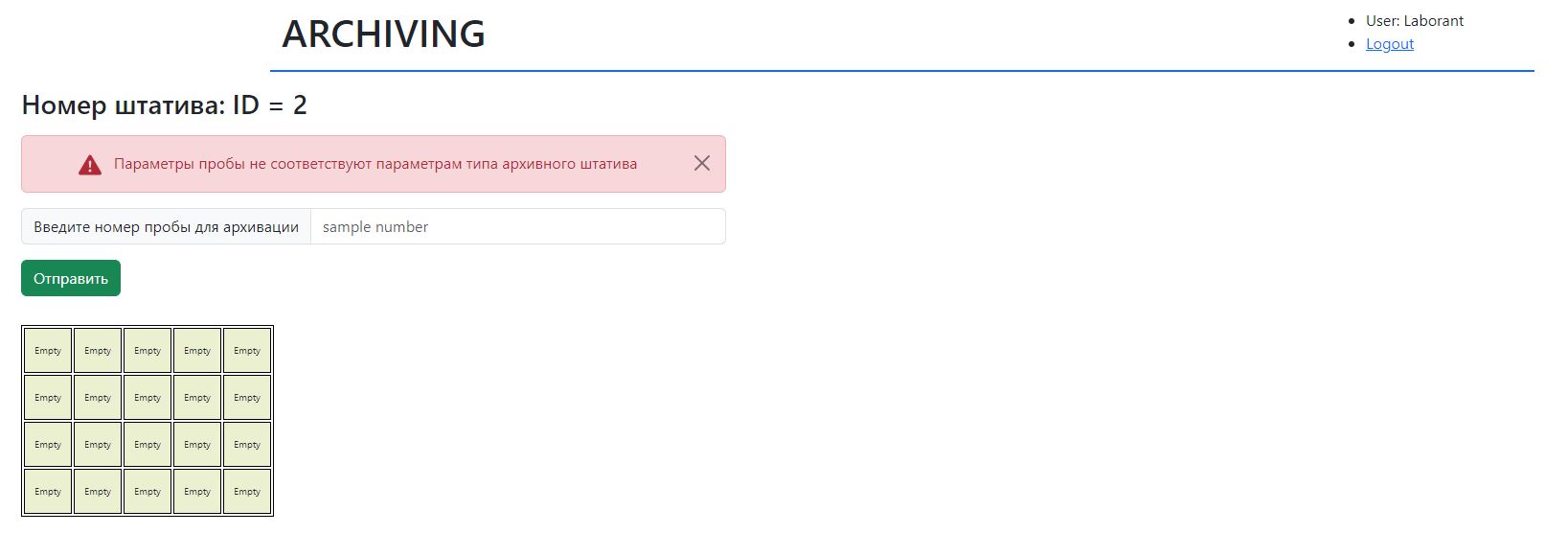
Рисунок 5.2.7 Всплывающее сообщение



При нахождении номера в базе то сравниваются текущие параметры пробы с установленными параметрами типа штатива (location, status, workflow, container type)

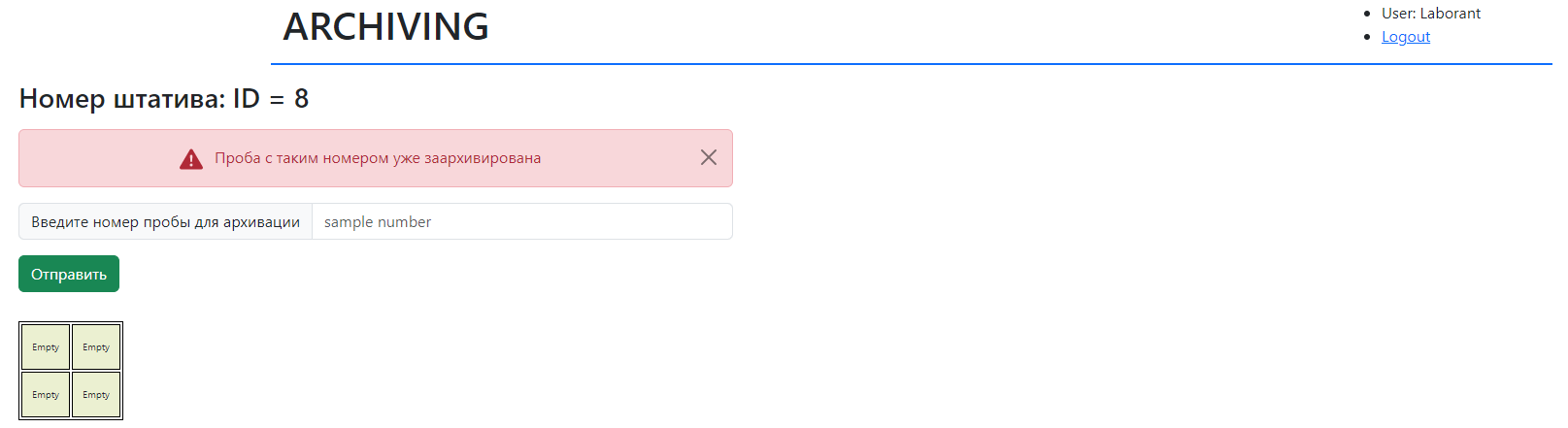
- Если параметры не совпадают, то будет вызвано сообщение типа alert:

Рисунок 5.2.8 Всплывающее сообщение



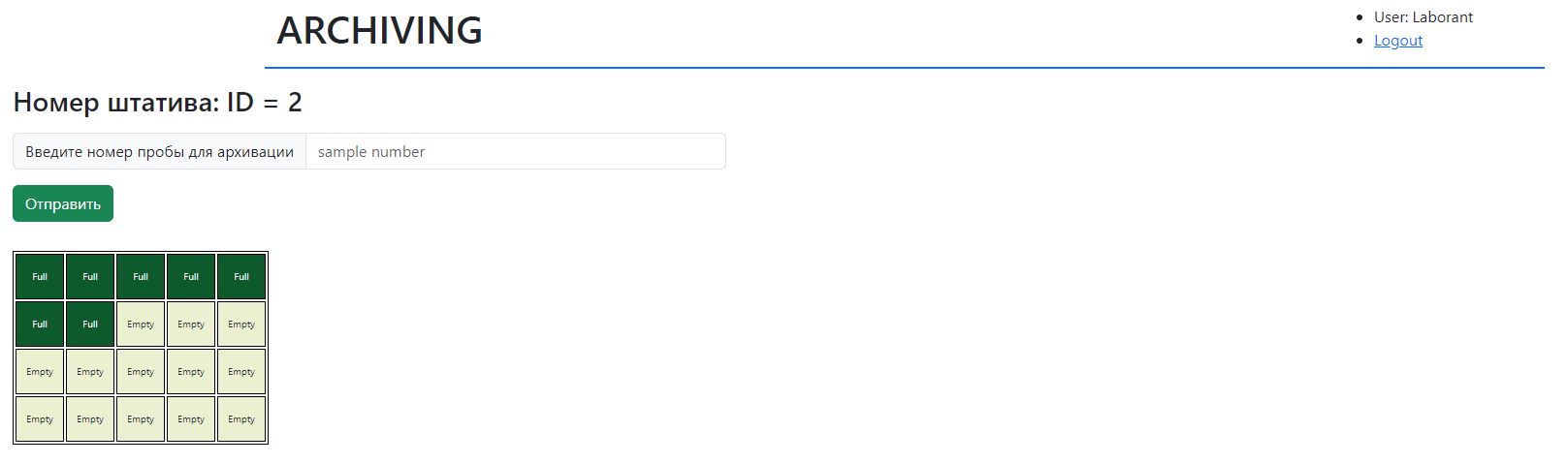
По техническому заданию, номера проб индивидуальны – нельзя заархивировать пробу несколько раз. В связи с этим, если проба с указанным номером уже присутствует в архиве, то будет вызвано сообщение типа alert:

Рисунок 5.2.9 Всплывающее сообщение



- В случае успешного сравнения происходит архивация пробы с привязкой к штативу и координатной сетке (X, Y). При этом пользователю показывается куда необходимо «физически» поставить пробу в штативе:

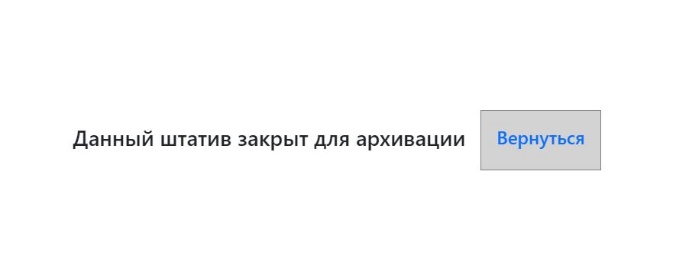
Рисунок 5.2.10 Архивация пробы



Штатив последовательно «заполняется» пробами.

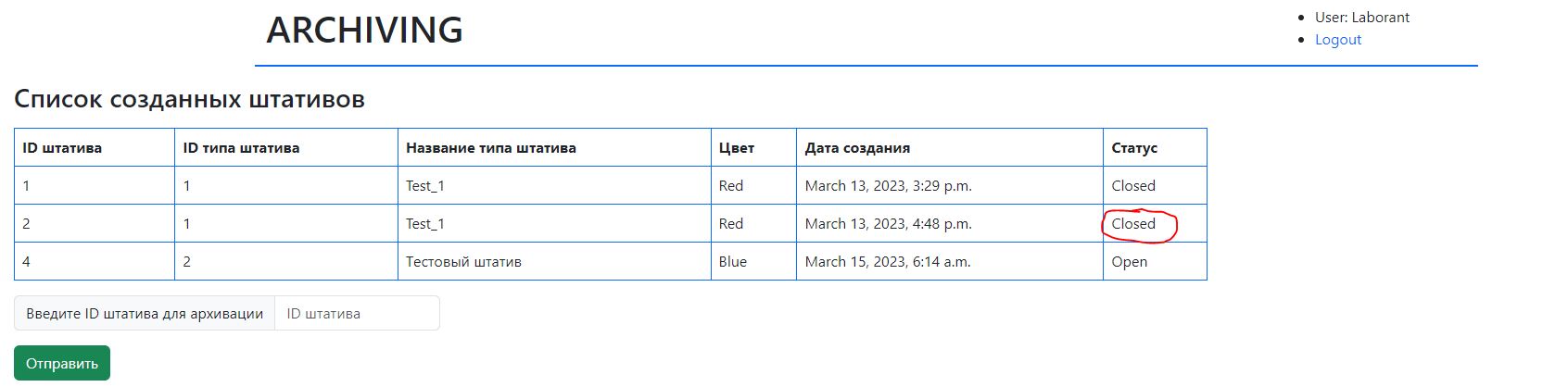
После заполнения штатива его статус меняется на «Closed», а пользователь перенаправляется на страницу выбора штатива для архивации:

Рисунок 5.2.11 Штатив закрыт для архивации



При этом можно увидеть изменившийся статус заполненного штатива

Рисунок 5.2.12 Статус штатива



* 1. **. Пользователь группы View**

При авторизации пользователя, который принадлежит группе View откроется страница поиска пробы в архиве

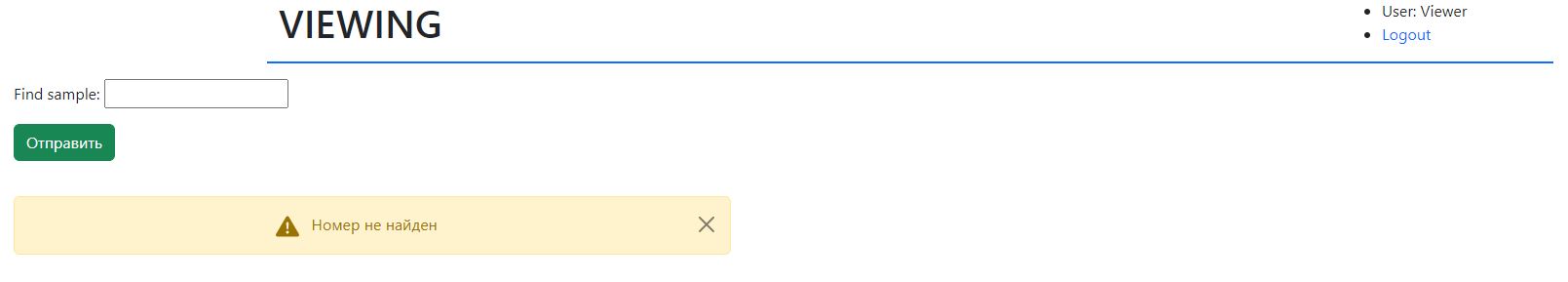
Рисунок 5.3.1 Страница поиска пробы в архиве



Для осуществления поиска необходимо ввести искомый номер пробы в поле и нажать кнопку «Отправить». При этом будет осуществлен поиск по указанному номеру в баз данных архива

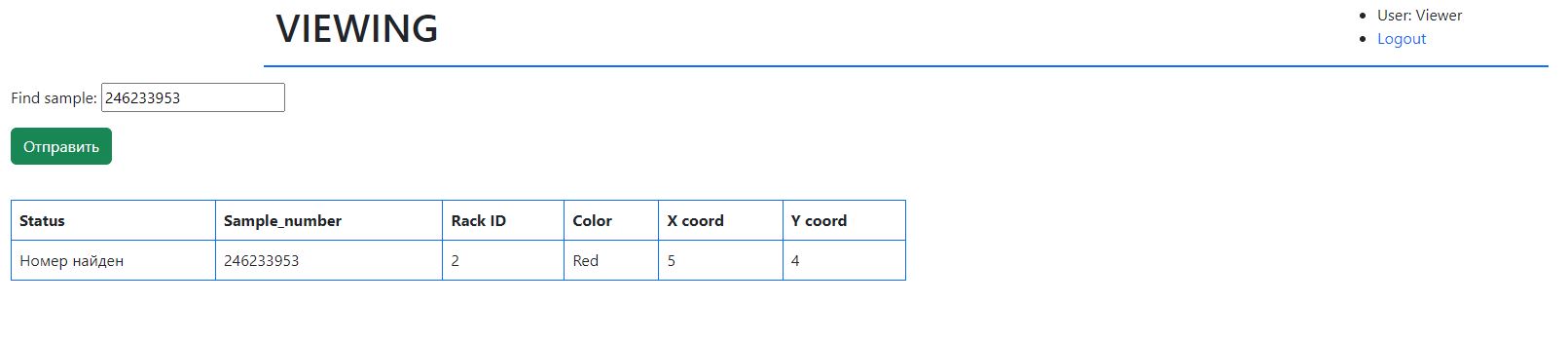
- В случае не нахождения номера в архиве будет вызвано сообщение типа alert:

Рисунок 5.3.2 Всплывающее сообщение



- В случае успешного нахождения номера на экран будет выведена информация о заархивированной пробе с указанием ID штатива и координат архивации

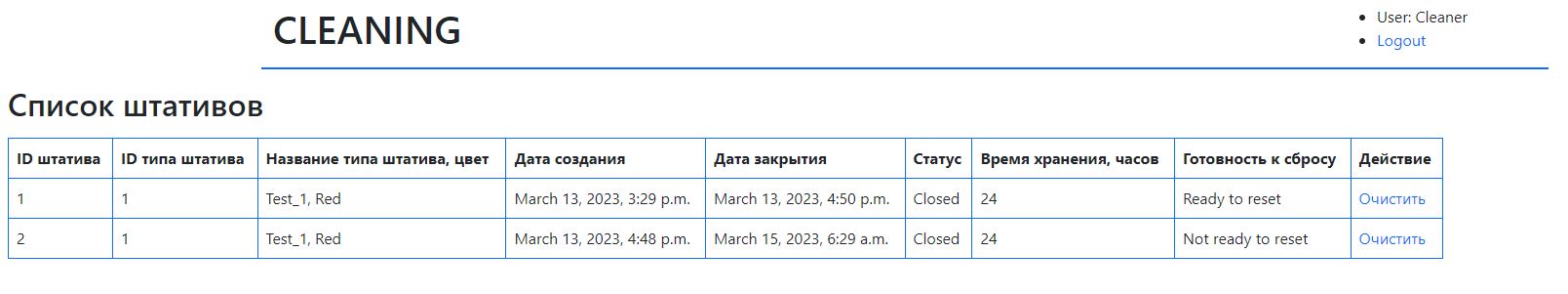
Рисунок 5.3.3 Всплывающее сообщение



* 1. **Пользователь группы Cleaning**

При авторизации пользователя, который принадлежит группе Cleaning откроется страница очистки штативов

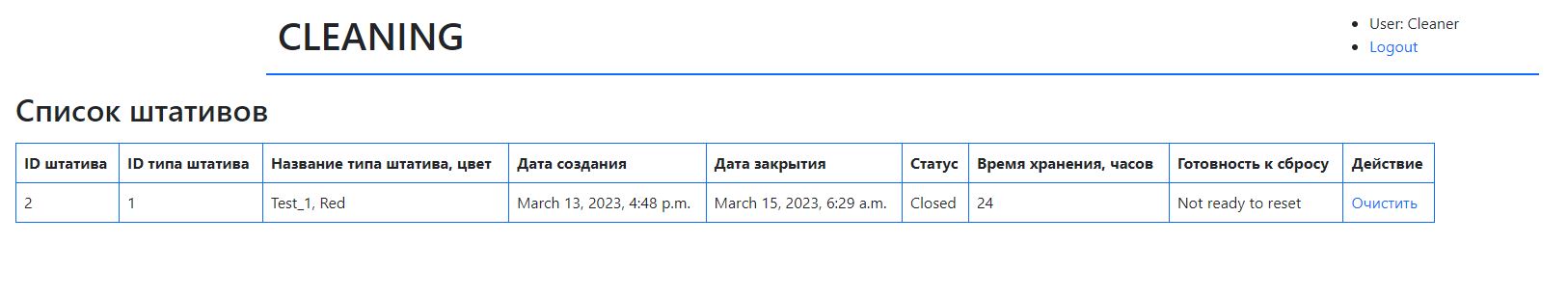
Рисунок 5.4.1 Страница очистки штативов



Пользователю выводится список штативов со статусом Closed и сообщением о готовности к сбросу в зависимости от того, сколько времени прошло с момента архивации (закрытия штатива) с учетом времени хранения, указанного при создании типа штатива.

При этом пользователь может совершить действие «Очистить», после которого все заархивированные пробы данного штатива будут «сброшены» (удалены из базы), а штатив будет переведет в статус «Open» и снова открыт к архивации. При этом из данного перечня он пропадет.

Рисунок 5.4.1 Страница очистки штативов



# Функции системы

**6.1. User Story**

Пользовательская история — это описание функциональной возможности ПО простыми, общими словами, составленное с точки зрения конечного пользователя. Она пишется с целью разъяснить, как именно функциональная возможность принесет пользу Заказчику.

В рамках аттестационной работы была составлена User story:

**Как** администратор **я хочу** создавать/удалять типы штативов **для того чтобы** обеспечить потребность лаборатории.

**Как** администратор **я хочу** настраивать типы штативов по указанным параметрам **для того чтобы** выполнить требования лаборатории к гибкости архивации.

**Как** администратор **я хочу** добавлять/удалять нужное количество штативов конкретных типов **для того чтобы** обеспечить наполнение архива.

**Как** администратор **я хочу** устанавливать срок хранения для типа штатива **для того чтобы** соблюсти необходимы сроки хранения биоматериала.

**Как** лаборант **я хочу** видеть в программе перечень типов штативов с порядковыми номерами **для того чтобы** понять какой пустой штатив нужно взять для дальнейшей архивации.

**Как** лаборант **я хочу** сканировать/вести id штатива **для того чтобы** на экране отобразился интерфейс заполнения штатива.

**Как** лаборант **я хочу** сканировать/ввести штрих код контейнера с биоматериалом **для того чтобы** понять, что с ним делать в рамках архивации.

**Как** санитарка **я хочу** видеть перечень штативов с истекшим сроком **для того чтобы** понять, какие штативы взять для сброса.

**Как** санитарка **я хочу** иметь возможность воспользоваться инструментом очистки для **того чтобы** сбросить штатив.

**Как** сотрудник лаборатории **я хочу** сканировать/ввести штрих код контейнера с биоматериалом **для того чтобы** увидеть в каком штативе и в какой ячейке находится указанный контейнер, либо получить сообщение, о том, что контейнера в архиве нет с указанием причины отсутствия.

**6.2. Функциональные требования**

Создаваемая система должна обеспечить процесса архивации биоматериала с учетом современных тенденций развития технических и программных средств, а также требований Заказчика.

**Основные функции системы**, подлежащие реализации:

* Создание, настройка штативов, типов штативов;
* Архивация;
* Очистка (сброс);
* Поиск;

**Функциональные требования** к системе представлены в таблице 6.2.

Таблица 6.2. Функциональные требования

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Описание «рамки решения»** | **ID** | **Описание «рамки проекта»** |
| **Модуль администрирования – реализация интерфейса (Management)** | | | |
| 1.1 | Администратор должен иметь возможность создавать/удалять типы штативов | 111  112 | Реализация возможности добавлять/удалять типы штативов;  Реализация базы данных для типов штативов; |
| 1.2 | Администратор должен иметь возможность настраивать типы штативов по возможным параметрам:   * Типы контейнеров; * Количество ячеек (X, Y); * Рабочие потоки * Текущий статус пробы; * Локация; * Название; * Цвет штатива; | 121 | Реализация возможности добавления указанных параметров архивации для типа штатива. |
| 1.3 | Администратор должен иметь возможность устанавливать срок хранения для типа штатива | 131 | Реализация возможности установки срока хранения для типов штатива |
| 1.3 | Система должна обеспечить возможность корректной настройки параметров типов штативов с учетом существующих данных. | 131 | Реализация обращения к базе данных с существующими параметрами:  - Location (один ко многим);  - Status (один ко многим);  - Workflows (многие ко многим);  - Container types (многие ко многим); |
| 1.4 | Система должна обеспечить проверку уникальности типа штатива | 141  142 | Реализация условий для проверки на уникальность;  Реализация вывода сообщения; |
| 1.5 | Администратор должен иметь возможность добавлять/удалять нужное количество штативов | 151  152 | Реализация возможность добавления/удаления необходимого количества штативов для указанного типа;  Реализация базы данных для штативов (один ко многим); |
| 1.6 | Если администратор ввел некорректное значение, система должна выдавать сообщение об ошибке | 161  162 | Реализация условий для проверки на корректность;  Реализация вывода сообщения; |
| **Модуль архивации – реализация интерфейса (Archiving)** | | | |
| 2.1 | Лаборант должен иметь возможность выбрать штатив, в который будет осуществлена архивация. | 211  212 | Реализация вывода на экран перечня доступных штативов;  Реализация обновления информации на экране после изменения статуса какого-либо штатива; |
| 2.2 | Система должна открывать интерфейс архивации после сканирования/ввода id штатива | 221  222 | Реализация поля для сканирования/ручного ввода штрих кода штатива;  Реализация перехода к системе архивации контейнеров с биоматериалом; |
| 2.3 | Если введен некорректный id, система должна выдавать сообщение об ошибке | 231  232  233 | Реализация условий для проверки id на корректность;  Реализация сравнения, введенного id с условиями;  Реализация вывода сообщения; |
| 2.4 | Если введен id «закрытого» штатива, система должна выдавать сообщение об ошибке | 241  242  243 | Реализация условий для проверки очередности использования штативов;  Реализация сравнения, введенного штрих кода с условиями;  Реализация вывода сообщения; |
| 2.5 | Лаборант должен иметь возможность отсканировать/ввести штрих код контейнера с биоматериалом для архивации. | 251 | Реализация поля для сканирования/ручного ввода штрих кода штатива |
| 2.6 | Система должна обеспечить архивацию по номеру отсканированного/введенного штрих кода контейнера | 261 | Реализация добавления, отсканированного штрих кода в базу данных в связке с ячейкой (X, Y), номером штатива, типом штатива. |
| 2.7 | Система должна показывать, куда в сетке штатива необходимо поставить контейнер, штрих код которого был отсканирован. | 271 | Реализация визуальной составляющей архивации, понятной для пользователя. |
| 2.8 | Если отсканирован штрих код контейнера неправильного типа, система должна выдавать сообщение об ошибке | 281  282  283 | Реализация условий для проверки штрих кода на корректность;  Реализация сравнения, введенного штрих кода с условиями;  Реализация вывода сообщения; |
| 2.9 | Если отсутствуют данные по отсканированному штрих коду контейнера, система должна выдавать сообщение об ошибке | 291  292 | Реализация поиска в базе по номеру штрих кода;  Реализация вывода сообщения; |
| 2.10 | Если отсканирован штрих код контейнера, свойства которого не подходят под параметры настроенного типа штатива, система должна выдавать сообщение об ошибке | 2101  2102  2103  2104 | Реализация поиска в базе по номеру штрих кода;  Реализация условий для проверки текущих свойств отсканированного контейнера с установленными параметрами типа штатива;  Реализация сравнения, введенного штрих кода с условиями;  Реализация вывода сообщения; |
| 2.11 | Если отсканирован штрих код контейнера, который уже присутствует в архиве | 2111  2112 | Реализация поиска в базе по номеру штрих кода;  Реализация вывода сообщения; |
| 2.12 | Если штатив заполнен система должна автоматически «закрыть» его и вернуться на экран выбора штативов. | 2121  2122  2123 | Реализация «закрытия» штатива с переходом на основной экран;  Добавление «закрытого» штатива в базу данных;  Реализация начала отсчета срока хранения для конкретного штатива; |
| **Модуль «сброса» – реализация интерфейса (Cleaning)** | | | |
| 3.1 | Система должна выводить на экран перечень штативов (id) и их статусов на текущий момент | 311  312 | Реализация вывода на экран перечня штативов с необходимым статусом (Closed);  Синхронизация с базой данных для обновления текущего статуса; |
| 3.2 | Санитарка должна иметь возможность «очистить» необходимые штативы | 321  322 | Реализация удаления всех номеров пробирок из очищаемого штатива  Реализация изменения статуса штатива |
| **Модуль поиска – реализация интерфейса (Viewing)** | | | |
| 4.1 | Система должна поддерживать поиск в базе данных архива по штрих коду контейнера | 411  412  413 | Реализация окна поиска;  Реализация поиска в базе данных по номеру штрих кода;  Реализация вывода результатов; |
| 4.2 | Если отсканирован некорректный штрих код, систем должна выдавать сообщение об ошибке | 421  422  423 | Реализация условий для проверки штрих кода на корректность;  Реализация сравнения, введенного штрих кода с условиями;  Реализация вывода сообщения; |
| 4.3 | Если отсканирован/введен штрих код контейнера, отсутствующий в базе данных архива, система должна выдавать сообщение об ошибке | 431 | Реализация вывода сообщения; |

**6.3. Функциональная модель**

Функциональная модель - это модель инфологического уровня представления, в которой акцентируется функциональный аспект моделирования предметной области. Функциональное моделирование позволяет рассмотреть бизнес процесс в виде набора функции, которые взаимодействуют между собой.

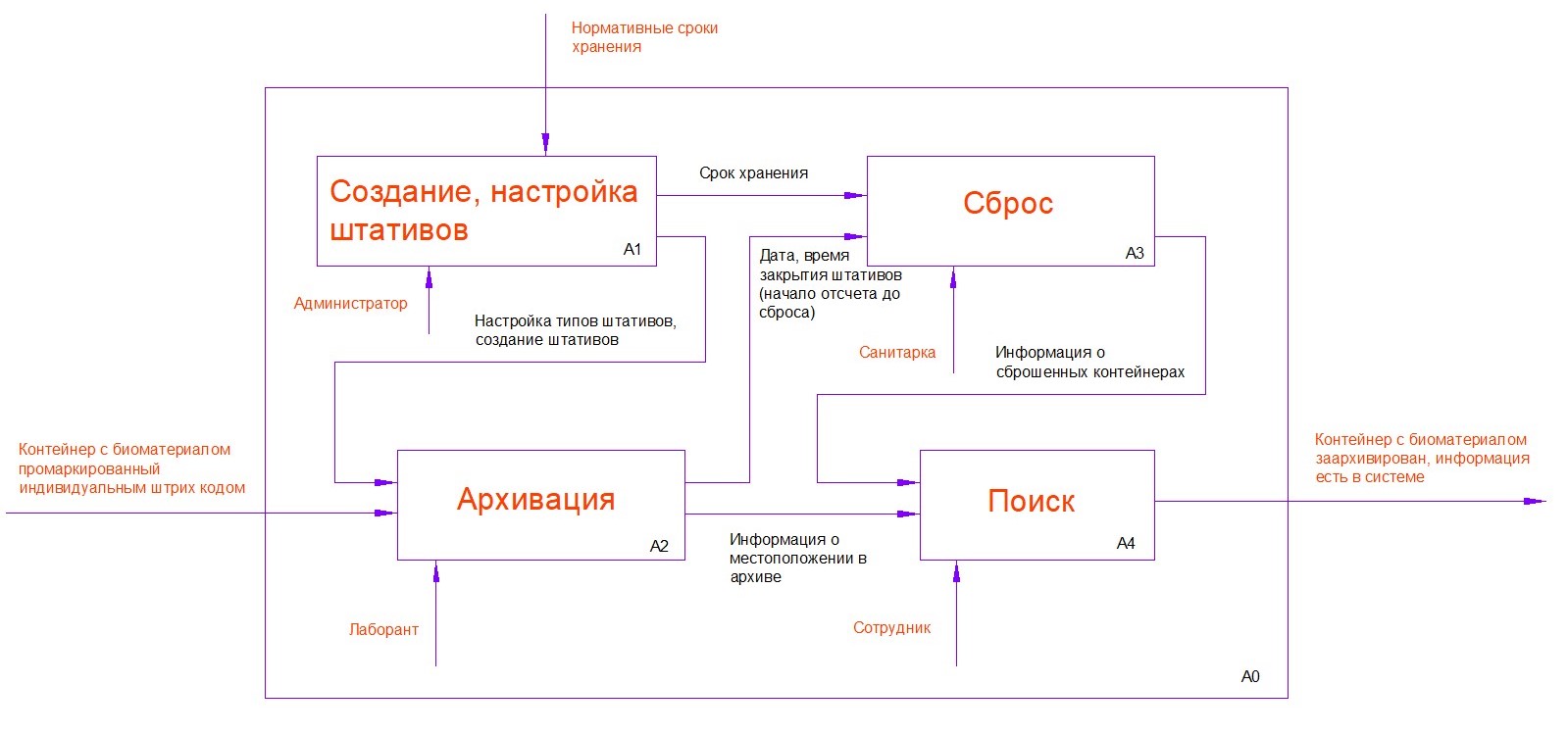
В рамках аттестационной работы была составлена функциональная модель

Рисунок 6.3.1 Контекстная диаграмма



На рисунке изображена декомпозитная диаграмма функциональной модели. На этой диаграмме показана работа приложений в целом, разбитая на более мелкие функции, взаимодействующие между собой.

Рисунок 6.3.2. Диаграмма декомпозиции 1-го уровня



На втором уровне модели отражены основные функции (тематически сгруппированные бизнес-процессы) системы и их взаимосвязи.

Рисунок 6.3.3. Диаграмма декомпозиции 2-го уровня А1



Рисунок 6.3.4. Диаграмма декомпозиции 2-го уровня А2

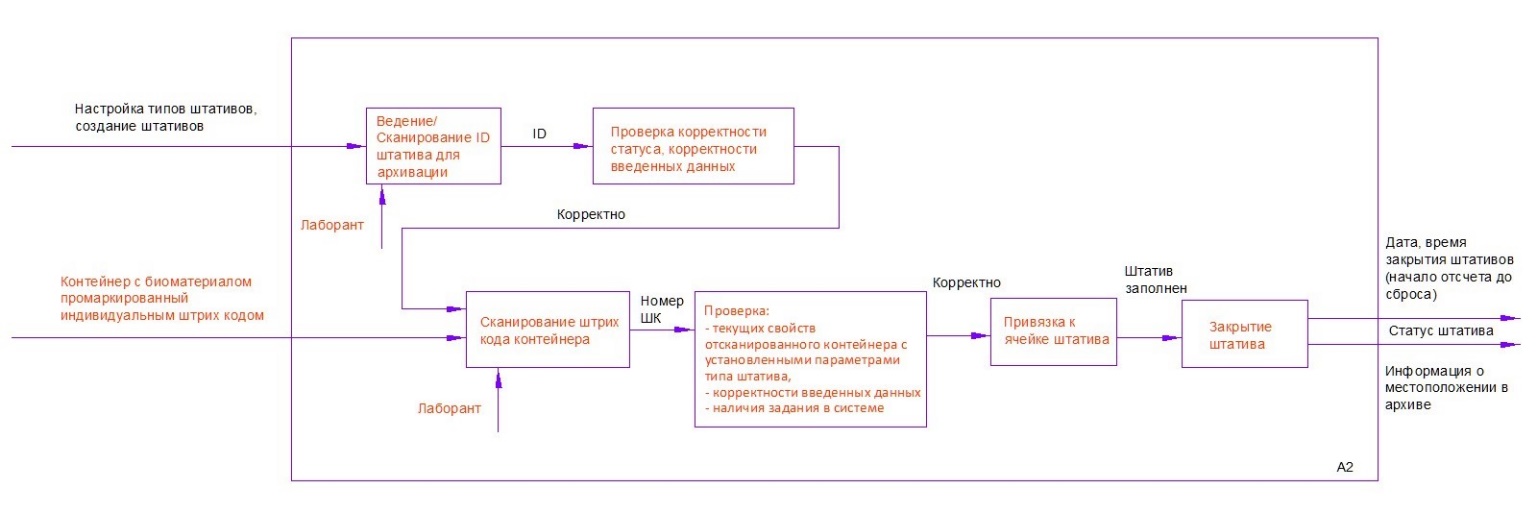


Рисунок 6.3.5. Диаграмма декомпозиции 2-го уровня А3

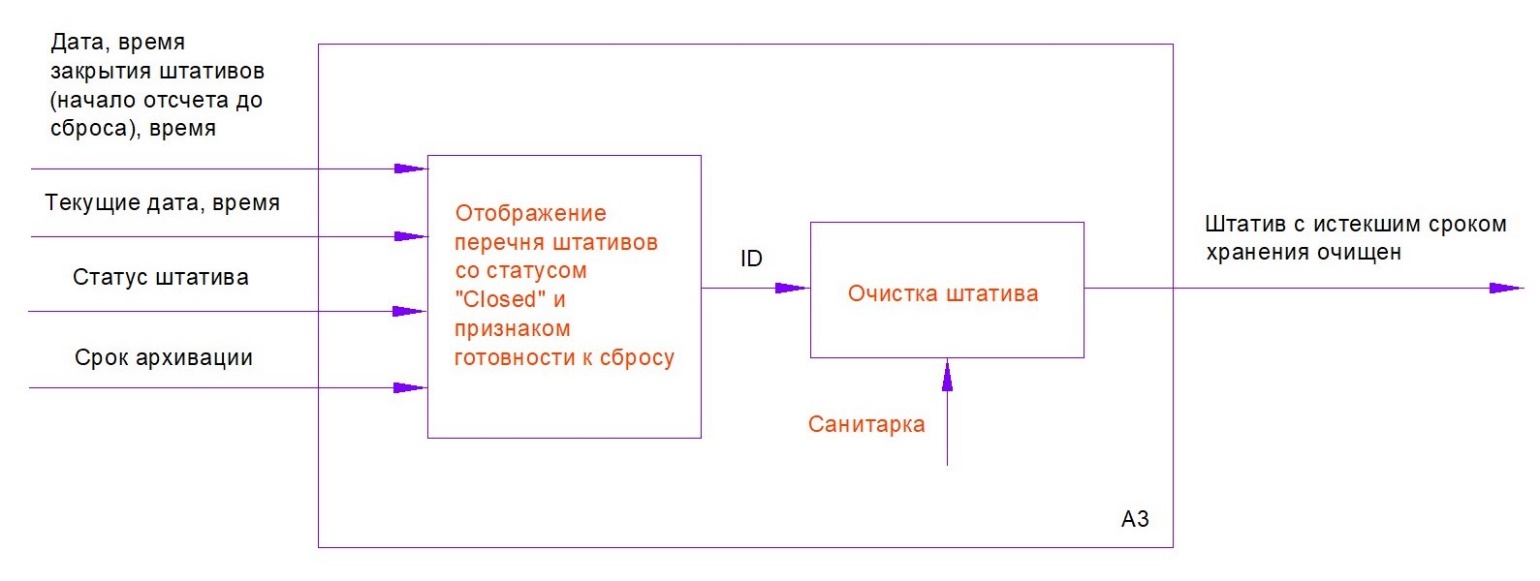
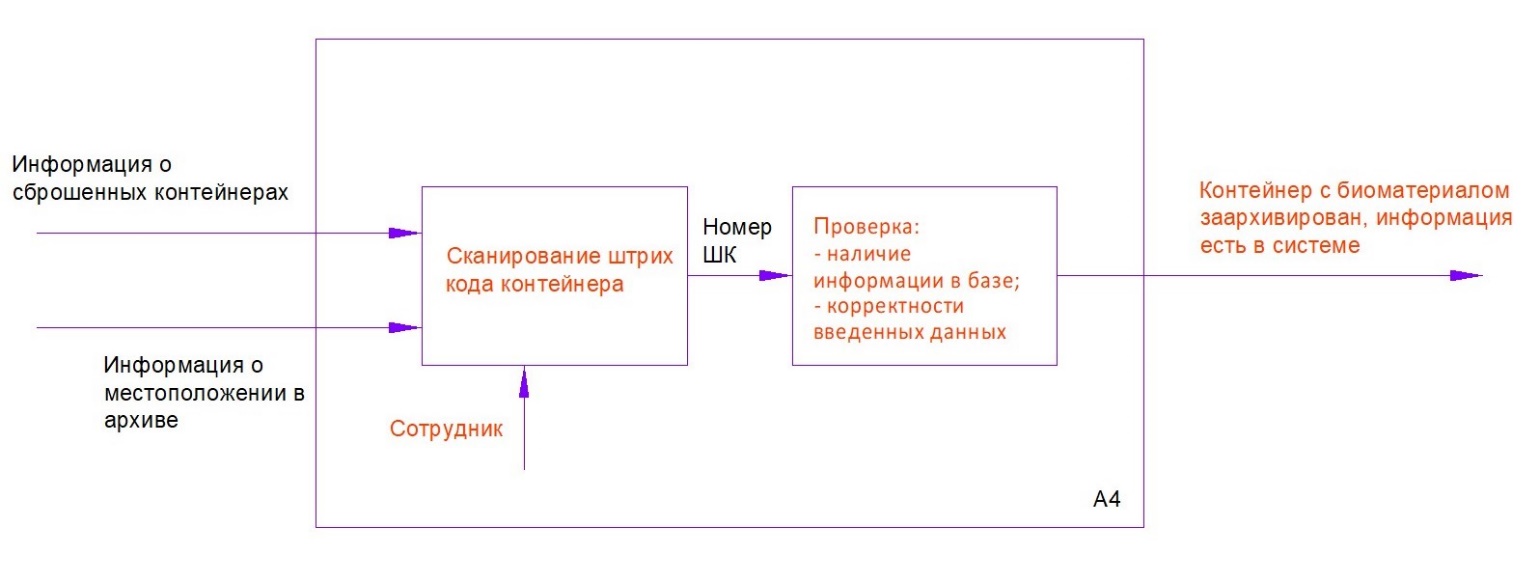


Рисунок 6.3.6 Диаграмма декомпозиции 2-го уровня А4



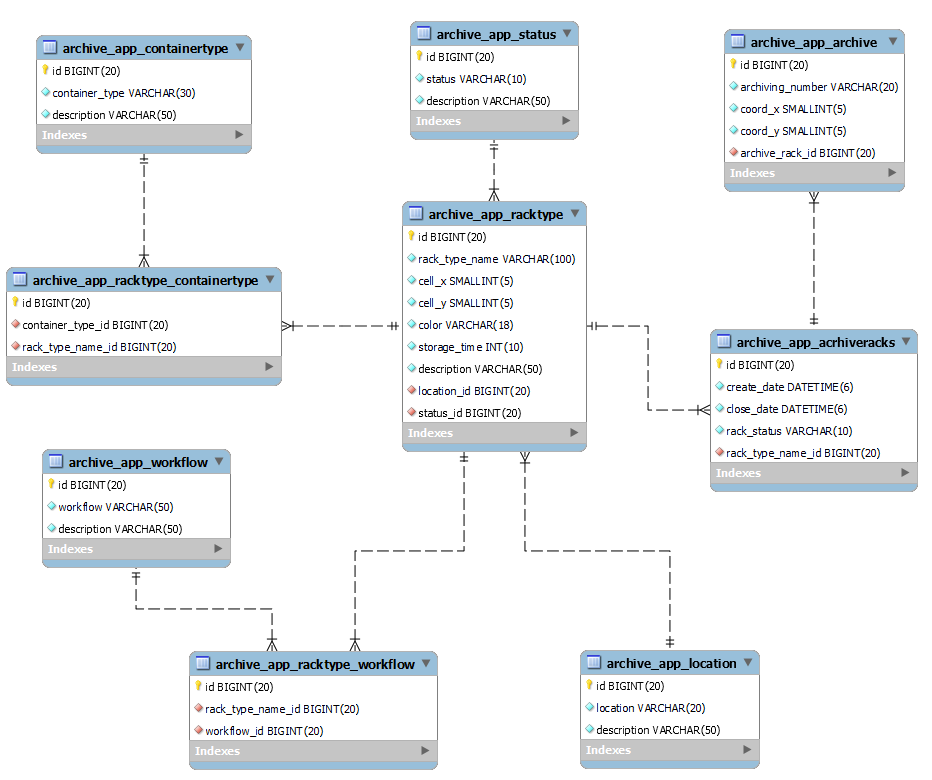
В процессе разработки приложения для архивации биоматериала были реализованы основные функции системы согласно функциональным требованиям.

# Структура данных

В процессе реализации приложения для архивации биоматериала была реализована собственная база данных (реляционная модель).

**Структура базы данных -** принцип или порядок организации записей в **базе данных** и связей между ними.

Рисунок 7.1. Модель базы данных приложения



Таблицы:

* archive\_app\_containertype:

- container\_type - тип контейнера;

- description – описание;

* archive\_app\_status:

- status – статус пробы;

- description – описание;

* archive\_app\_workflow:

- workflow – рабочий поток;

- description – описание;

* archive\_app\_location:

- location - локация выполнения

- description – описание;

* archive\_app\_racktype - типы штативов:

- rack\_type\_name – название типа штатива;

- cell\_x – количество ячеек по горизонтали;

- cell\_y – количество ячеек по вертикали;

- storage\_time – время хранения проб;

- description – описание;

* archive\_app\_archiveracks – штативы:

- create\_date – дата создания;

- close\_date – дата закрытия;

- rack\_status – статус штатива;

* archive\_app\_archive: архив биоматериала:

- archiving\_number – номер заархивированной пробы;

- coord\_x – координата пробы по оси X;

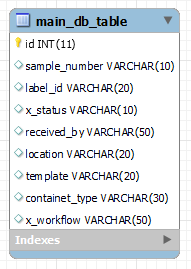
- coord\_y – координата пробы по оси Y;

* archive\_app\_racktype\_workflow, archive\_app\_racktype\_ containertype – связующие таблицы (многие ко многим)

База данных приведена к третьей нормальной форме нормализации (3NF)

Так же в проекте используется реальная база данных номеров пробирок с соответствующими параметрами.

Рисунок 7.2. Реальная база данных



* main\_db\_table: реальная база данных

Данная база данных не администрируется, используется только для чтения данных (сравнение параметров пробирки с параметрами штатива при архивации).

1. **Тестирование**

**Тестирование программного обеспечения -** проверка соответствия реальных и ожидаемых результатов поведения программы, проводимая на конечном наборе тестов, выбранном определённым образом.

**Цель тестирования** — проверка соответствия ПО предъявляемым требованиям, обеспечение уверенности в качестве ПО, поиск очевидных ошибок в программном обеспечении, которые должны быть выявлены до того, как их обнаружат пользователи программы.

При тестировании приложения для архивации биоматериала был сформирован следующий перечень групп:

**Функциональное тестирование** – необходимо убедиться, что разрабатываемое приложение соответствует нужной функциональной спецификации, упомянутой в документации по разработке (см. User Story, п. 6.1)

* Работоспособность основного функционала приложения;
* Авторизация (Логин/Пароль);

- пользователь существует в системе с введенным логином и паролем;

- пользователь с введенным логином/паролем не существует в системе;

- пользователю конкретной группы доступен только определенный функционал в соответствии с техническим заданием;

* Валидация полей ввода.

- числовые поля не принимают буквы;

- отрицательные значения не используются;

- текст не выезжает за границы поля;

* Валидация обязательных полей;

- максимальная и минимальная длина;

- обязательность к заполнению;

* Навигация между разделами приложения;
* Корректное отображение ошибок;
* Всплывающие сообщения в соответствии с техническим заданием;
* Сообщения об ошибках ввода в соответствии с техническим заданием;
* Функциональность доступных кнопок;
* Тестирование WebView;
* Все стили загружаются;
* Картинки загружаются;

**Тестирование безопасности**: Данная проверка нацелена на поиск недостатков и пробелов с точки зрения безопасности приложения.

* Пользователь не может авторизоваться с неверным логином/паролем;
* Пароль скрыт астерисками на странице авторизации.
* Доступ к закрытым разделам сайта (admin).
* SQL-инъекции.
* HTML-инъекции.
* Роли пользователей и доступ к контенту.

**Тестирование удобства использования** - проверка удобства пользования, обнаружение изъянов в навигации и интерфейсе, а также избыточной или недостаточной информативности:

* Отсутствие орфографических и грамматических ошибок, все страницы имеют корректные заголовки.
* Выравнивание картинок, шрифтов, текстов.
* Подсказки существуют для всех полей.
* Отступы между полями, колонками, рядами и сообщениями об ошибках.
* Кнопки имеют стандартный размер, цвет.
* На сайте нет битых ссылок и изображений.
* Отображение чекбоксов и радио-кнопок, кнопки должны быть доступны с клавиатуры, и пользователь должен быть в состоянии пользоваться сайтом, используя только клавиатуру.
* Отображение выпадающих списков.
* Наличие плейсхолдеров в полях.
* Переходы между страницами.

**Тестирование веб-сервисов -** проверка корректности вызываемых веб-приложением сервисов на предмет корректной обработки данных, изменения статусов объектов, возвращение информации из БД и проч.

**Кроссплатформенное тестирование** проводится, чтобы убедиться, что ваше приложение совместимо с другими браузерами, различными оболочками, аппаратным обеспечением устройства.

1. Тестирование в различных браузерах (Firefox, Chrome, Safari): анимация, верстка, шрифты, уведомления и т.д.
2. Java Script код работает в разных браузерах.

Ввиду ограниченного срока выполнения аттестационной работы, при разработке приложения архивации биоматериала для медицинской лаборатории было выполнено (в основном) ручное тестирование по заявленным группам тестирования.

Однако использование автоматических тестов несет в себе очевидные преимущества:

- меньшие временные затраты на проведение тестов;

- подробное выполнение;

- постоянную функциональность;

- последовательность действий;

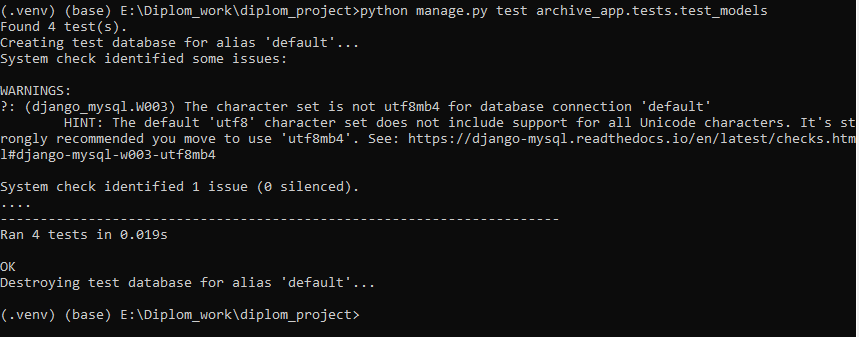
С пониманием этого, предполагается в дальнейшем передать приложение сторонней организации для полного автоматического тестирования.

Django предоставляет фреймворк для создания тестов, построенного на основе иерархии классов, которые, в свою очередь, зависят от стандартной библиотеки Python [unittest](https://docs.python.org/3/library/unittest.html#module-unittest). Несмотря на название, данный фреймворк подходит и для юнит-, и для интеграционного тестирования. Фреймворк Django добавляет методы API и инструменты, которые помогают тестировать как веб так и, специфическое для Django, поведение. Это позволяет вам имитировать URL-запросы, добавление тестовых данных, а также проводить проверку выходных данных ваших приложений.

В рамках разработки приложения архивации биоматериала для медицинской лаборатории были разработаны некоторые тестовые кейсы для наглядности автоматизации тестирования.

При тестировании был использован базовый класс - [django.test.TestCase](https://docs.djangoproject.com/en/1.10/topics/testing/tools/#testcase). Этот класс создаёт чистую базу данных перед запуском своих методов, а также запускает каждую функцию тестирования в его собственной транзакции. У данного класса также имеется тестовый [клиент](https://docs.djangoproject.com/en/1.10/topics/testing/tools/#django.test.Client), который вы можете использовать для имитации взаимодействия пользователя с кодом на уровне отображения.

Рисунок 8.1 пример выполнения автоматического тестирования в среде Django



1. **Развертывание**

Окружение развёртывания - это среда, которое предоставляет сервер, на котором будет размещаться веб-приложение для публичного запуска и доступа. Данное окружение включает в себя:

* Физический сервер, на котором будет запускаться приложение
* Операционную систему;
* Языки программирования времени выполнения (скриптовые) и библиотеки, которые использует приложение;
* Веб-сервер, используемый для обслуживания страниц и другого контента;
* Сервер приложений, который передаёт "динамические" запросы между сайтом Django и веб-сервером;
* Базу данных, от которой зависит приложение;

В условиях лабораторного производства планируется, что разработанное приложение будет размещено на существующем сервере Заказчика с возможностью локального распространения внутри Организации.

Для аттестационной работы, разработанное приложение было развернуто на ресурсах облачной платформы PythonAnywhere.

PythonAnywhere - это онлайн интегрированная среда разработки (IDE) и служба веб-хостинга (Платформа как услуга) на основе языка программирования Python.

Ссылка на приложение: <http://whiter777.pythonanywhere.com/>

# Заключение

В рамках аттестационной работы было разработано приложение архивации биоматериала для медицинской лаборатории.

При этом:

* проведен анализ исходного материала, предметной области;
* сформулированы цели и задачи работы;
* использованы современных пакеты компьютерных программ и технологий;
* в процессе разработки решены технические задачи;
* изучено современное программное обеспечение;

При решении задачи были изучены требования к программному обеспечению, пользовательскому интерфейсу. Получен практический навык работы с веб-фреймворком Django, СУБД MySQL. Было разработано приложение, которое соответствует предъявленным требованиями аттестационной работы, и функционирует согласно описанным требованиям. Также были закреплены и получены новые навыки работы с использованием языка программирования Python.

# Список литературы

* Вигерс К. Разработка требований к программному обеспечению/пер. с англ. – М.: Русская Редакция, 2014. – 736 с.
* Алистер Коберн. Современные методы описания функциональных требований к системам. М.: Лори, 2017. – 288 с
* Руководство по веб-фреймворку Django [Электронный ресурс]: Информационный ресурс METANIT.COM. - Режим доступа: <https://metanit.com/python/django/>.
* Django documentation [Электронный ресурс]: Информационный ресурс Django. - Режим доступа: <https://docs.djangoproject.com/en/4.1/>.
* Bootstrap documentation [Электронный ресурс]: Информационный ресурс Bootstrap. - Режим доступа: <https://getbootstrap.com/docs/5.3/getting-started/introduction/>.
* MySQL documentation [Электронный ресурс]: Информационный ресурс MySQL. - Режим доступа: <https://dev.mysql.com/doc>.