|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ЦИФРОВАЯ КАФЕДРА 01

КУРС «WEB-РАЗРАБОТЧИК»

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

***К КОМАНДНОМУ ПРОЕКТУ***

***НА ТЕМУ:***

***«РАЗРАБОТКА МНОГОПОЛЬЗОВАТЕЛЬСКОГО  
ВЕБ-ПРИЛОЖЕНИЯ С ОТОБРАЖЕНИЕМ   
3D-МОДЕЛЕЙ ГИДРОТЕХНИЧЕСКИХ ОБЪЕКТОВ И ДОПОЛНИТЕЛЬНОЙ ИНФОРМАЦИИ»***

Участники проекта:

Студент МТ1-61 **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  С.А. Сандер

(Группа) (Подпись, дата) (И.О.Фамилия)

Студент РКТ2-121 **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** О.А. Серебрянников

(Группа) (Подпись, дата) (И.О.Фамилия)

Студент СМ9-101 **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** А.И. Солдаткин

(Группа) (Подпись, дата) (И.О.Фамилия)

Студент Э2-82Б **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** Д.В. Фирсов

(Группа) (Подпись, дата) (И.О.Фамилия)

Студент МТ5-101 **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** А.Е. Чибизов

(Группа) (Подпись, дата) (И.О.Фамилия)

СОДЕРЖАНИЕ

[1. Техническое задание на реализацию проекта по курсу веб-разработки 3](#_Toc140969192)

[1.1. Цель 3](#_Toc140969193)

[1.2. Задачи 3](#_Toc140969194)

[1.3. Требования к оформлению пояснительной записки 3](#_Toc140969195)

[1.4. Организационные требования 4](#_Toc140969196)

[1.5. Технические требования 4](#_Toc140969197)

[1.6. Требования к наименованию переменных 6](#_Toc140969198)

[1.7. Требования к публикации и развертыванию веб-приложения 6](#_Toc140969199)

[2 Проект «Hydroscan» 7](#_Toc140969200)

[3 Создание веб-приложения для проекта «Hydroscan» 9](#_Toc140969201)

[3.1. Техническое задание от «Hydroscan» 9](#_Toc140969202)

[3.2. Общие требования 9](#_Toc140969203)

[3.3. Главная страница веб-приложения 10](#_Toc140969204)

[3.4. Личный кабинет-дашборд (со стороны клиента) 10](#_Toc140969205)

[3.4.1. Отображение заказа 10](#_Toc140969206)

[3.4.2 Создание заказа 11](#_Toc140969207)

[3.5. Административное управление через Django (со стороны компании) 12](#_Toc140969208)

[3.6. База данных 12](#_Toc140969209)

[4. Структура и управление веб-приложением 13](#_Toc140969210)

[5. Работа веб-приложения 18](#_Toc140969211)

[5.1. Главная страница 18](#_Toc140969212)

[5.2. Личный кабинет 22](#_Toc140969213)

[5.2.1. Регистрация в личном кабинете 22](#_Toc140969214)

[5.3. Страница заказа 24](#_Toc140969215)

[5.4. Страница администратора сайта 30](#_Toc140969216)

# 1. Техническое задание на реализацию проекта по курсу веб-разработки

## Цель

• Разработка веб-приложения с использованием фреймворка Django.

## Задачи

* Проектирование и прототипирование веб-приложения с использованием BPMN – диаграмм и дизайн-макетов (Можно использовать Figma / FigJam / Miro);
* Разработка серверной логики и моделей данных с использованием Django;
* Разработка шаблонов отображения данных с помощью HTML / CSS / JS;
* Разработка сопроводительной документации по установке и запуску
* веб-приложения;
* Публикация и развертывание веб-приложения;
* Оформление пояснительной записки к проекту.

## Требования к оформлению пояснительной записки

• Титульный лист с наименованием проекта, ФИО исполнителя;

• Содержание;

• Нумерация страниц;

• Шрифт: Times New Roman, 14 пт;

• Выравнивание: по ширине страницы;

• Межстрочный интервал: 1,5;

• Вставки исходного кода оформляются в виде листингов, с подписями;

• Блок-схемы и изображения оформляются в виде рисунков, с подписями.

## Организационные требования

* Проект выполняется в командах по 4-5 человек;
* Командное взаимодействие происходит через Gitlab, развернутый на сервере МГТУ;
* Вклад каждого студента в проект оценивается по коммитам;

## Технические требования

• Проект реализуется с помощью фреймворка Django, версия 4.2;

• UI-компоненты выполнены с помощью фреймворка Bootstrap / Material / Taiga;

• Проект многопользовательский, поддерживается регистрация и авторизация пользователей, реализована выдача уникального контента для каждого пользователя

• Реализовано меню, которое позволяет перемещаться между разделами веб-приложения;

• Реализовано меню, которое позволяет перемещаться между разделами веб-приложения;

• В проекте используется минимум 3 связанных между собой сущности (например, факультет-группа-студент):

• Связь сущностей должна быть не только логической, но и физической, на уровне модели данных;

• Реализовано минимум 3 компонента:

• Минимум 1 из компонентов отображает данные в виде списка;

• В проекте используются формы;

• Реализован CRUD подход:

• Для GET методов реализована передача параметров через query;

• Для PUT и DELETE методов реализована передача параметров через URL;

• Для POST методов реализована передача параметров через form-data.

• В случае, если запрашиваемой страницы не существует, пользователь должен быть перенаправлен на страницу с ошибкой 404;

• Реализована проверка входных параметров на корректность на стороне серверной логики. В случае, если входные параметры некорректны, пользователь должен получать осмысленное описание ошибки;

• Используется валидация данных в формах;

• Реализован процесс авторизации / регистрации с использованием нескольких пользовательских групп:

• Для неавторизованного пользователя часть функционала, связанная с добавлением, обновлением, удалением информации, должна быть недоступна.

• В части HTML / CSS кода:

• Не используется атрибут style;

• Не используется !important;

• Идентификаторы используются строго по своему назначению;

• В классах, в случае необходимости, реализован подход наследования;

• Используется селектор лоботомированной совы;

• Форматирование исходного кода выполнено с использованием отступов;

• Исходный код проходит проверку линтерами PYLINT-DJANGO и DJLINT;

• При развертывании приложения режим отладки должен быть деактивирован;

• В консоли веб-браузера не должны отображаться ошибки, вызванные работой веб-приложения.

## Требования к наименованию переменных

* Наименования переменных должны коротко и точно отражать их суть;
* Наименования переменных не должны быть написаны с использованием транслитерации;
* Наименования переменных не должны содержать орфографических ошибок;
* Наименования переменных должны быть оформлены в едином стиле.

## Требования к публикации и развертыванию веб-приложения

* Приложение должно разрабатываться с использованием системы
* контроля версий:
  + - Наименование коммитов должны отражать суть внесенных
* изменений;
* Репозиторий с приложением должен быть опубликован и иметь
* публичный режим доступа к моменту проверки работы;
* Приложение, развернутое на сервере, не должно содержать каталог .git и
* другие системные файлы питона или вспомогательных утилит;
* Необходимо реализовать CI \ CD процесс, который будет автоматически
* публиковать приложение на сервере при добавлении коммитапр в
* master-ветку.

# 2 Проект «Hydroscan»

«Hydroscan» – это проект, предназначенный для взаимодействия с клиентами, которые занимаются анализом гидротехнических сооружений. Это помогает им в использовании дронов и сканеров для сбора информации. «Hydroscan» помогает пользователям в сохранении и управлении полученной информации, что повышает эффективность их работы.

Данный проект был создан с целью улучшения процесса взаимодействия между аналитиками гидротехнических сооружений и их клиентами. Часто общение между ними проходит в формате электронных писем, что не всегда удобно и эффективно. Поэтому «Hydroscan» предоставляет возможность создания заказов и управления ими через административную панель на Django. Это позволяет пользователям сохранять и управлять полученной информацией, что повышает эффективность их работы.

«Hydroscan» был создан в рамках выпускного проекта на университетском курсе. Однако, идея проекта возникла из необходимости создания сайта для стартапа знакомого. Вдохновением для команды послужила идея о том, что существующие решения для взаимодействия с клиентами не удовлетворяли потребностям аналитиков гидротехнических сооружений.

MVP проекта включает в себя возможность регистрации, создания заказов и управление ими через административную панель на Django. Команда решила использовать Django на бэкенде и Quasar на фронтенде, потому что они уже изучали эти технологии.

Компания занимается визуальным осмотром гидротехнических сооружений (пирсы, причалы, опоры мостов, плотины, шлюзы, дамбы), а также плавучих средств (корабли, лодки, баржи) с возможностью создать 3D-модель объекта и выдавать отчет о дефектах. Для этого мы используем подводный аппарат (ПА) и беспилотный летательный аппарат (БПЛА) для осмотра объекта. С помощью подводного робота мы осматриваем и сканируем подводную часть гидротехнического объекта, а с помощью БПЛА — надводную часть. Компания выезжает на объект, осматривает его и делает 3D-модель с отчетом о дефектах.

Проект еще не запущен, поэтому первые пользователи еще не искались. Команда планирует провести маркетинговые исследования и разработать стратегию для привлечения клиентов после того, как проект будет готов к запуску. Кроме того, команда рассматривает возможность проведения тестового запуска проекта, чтобы убедиться в его эффективности и узнать, какие еще изменения нужно внести.

Hydrocan - это проект, который может изменить способ работы аналитиков гидротехнических сооружений. Это поможет им сэкономить время и увеличить эффективность своей работы. Команда готова продолжать работу над проектом и улучшать его, чтобы он стал еще лучше и более востребованным.

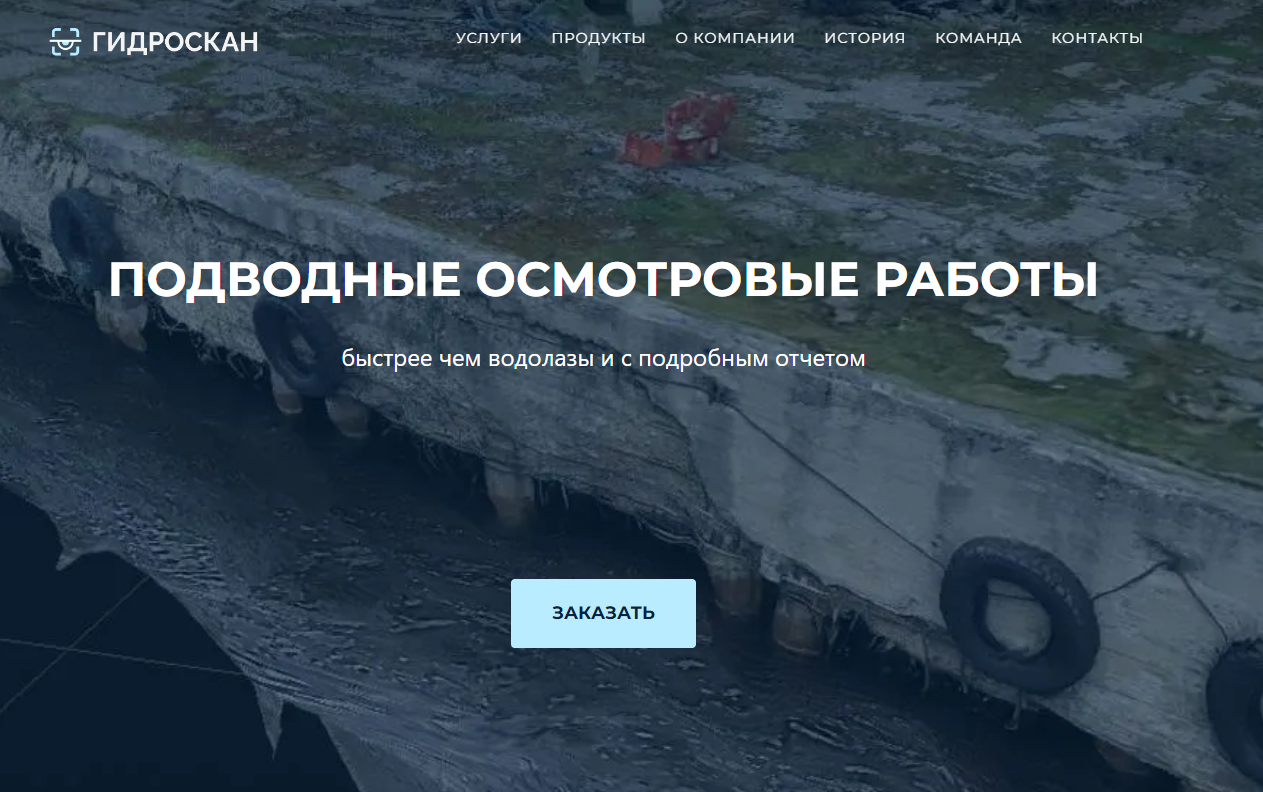


Рисунок 1 – Официальная страница проекта «Hydroscan»

# 3 Создание веб-приложения для проекта «Hydroscan»

## 3.1. Техническое задание от «Hydroscan»

Требуется разработать многопользовательский веб-интерфейс, на который можно загружать полученные 3D-модели и отчеты о дефектах. Клиенты могут зайти в свой личный кабинет, где они смогут просмотреть 3D-модель объекта, получить краткую информацию о нем, скачать отчет о дефектах и заказать дополнительные услуги, например, более детальную обработку 3D-модели на наличие дефектов.

## 3.2. Общие требования

* Использовать современные технологии и фреймворки;
* Бэкенд – Django, Python;
* Адаптивность под разные устройства (компьютер, смартфон);
* Базовые требования по безопасности – средства защиты от несанкционированного доступа к данным и от атак типа SQL Injection,  
  XSS и CSRF;
* Страницы:
  + Главная страница;
  + Личный кабинет / дашборд:
    - Отображение текущего заказа и истории заказов, если они были ранее;
    - Создание нового заказа с указанием;
      * + Административное управление (управление заказами, добавление  
          3D-модели и отчета в заказы, изменение статуса заказа, управление главной страницей – добавить ещё один выполненный проект);

CI / CD и развертывание в контейнерах;

По возможности использовать asyncio для блокирующих тасков.

## 3.3. Главная страница веб-приложения

Хедер:

1. Логотип;
2. Навигация по главной странице;
3. Кнопка входа в аккаунт.

Контент:

1. Фон с текстом и кнопка «сделать заказ»;
2. Услуги (будет реализовано 3-4 услуги);
3. Выполненные проекты;
4. История компании (в виде таймлайна);
5. Команда (фотографии разработчиков);
6. Форма обратной связи и контакты.

Футер: стандартный футер с информацией.

## Личный кабинет-дашборд (со стороны клиента)

У каждого заказчика будет личный кабинет-дашборд, где он сможет подробно посмотреть информацию по каждому заказу. С главной страницы клиент нажимает кнопку войти, или авторизоваться, или подключиться и попадает в дашборд. По умолчанию открывается или последний заказ, или пустой дашборд. Слева будет список заказов, а справа окно отображения заказа. Внизу под списком заказов будет кнопка создать заказ, при нажатии на которую остальное содержимое справа перезагружается в окно создания заказа.

### Отображение заказа

* **Слева сверху** список заказов - имя объекта + дата создания (если несколько раз заказывали осмотр одного объекта). Можно добавить индикаторы по каждому заказу (желтый - выполняется, зеленый - выполнен, серый - отменен, красный - ошибка оплаты или еще ошибка другого рода, синий - ожидает оплаты);
* **Слева снизу** - кнопка создать новый заказ (переход к странице создания заказа);
* **Посередине** большое окно для 3D-модели. для каждого заказа загружается моделька и на ней в разных местах отмечаются точки, на которые можно нажать и приблизить это место. будет полезно чтобы приближать дефекты;
* **Справа** можно отобразить следующую информацию:
  + **Статус заказа** – «в процессе», «выполнен», «отменен» и т.д.;
  + **Детали заказа** – дата заказа, название объекта, описание объекта, список услуг;
  + **Скачать отчет** – кнопка скачать отчет;
  + **Скачать 3D-модель** – кнопка;
  + **Дополнительные услуги** - список дополнительных услуг – те, которые не отметили при создании заказа и какие-то дополнительные.

### 3.4.2 Создание заказа

* Выбор типа услуги, кнопки-чекбоксы — осмотр подводной части объекта, осмотр надводной части объекта, создание 3D-модели объекта, составление отчета об осмотре, автоматический поиск дефектов и создание отчета о дефектах (возможно в будущем появятся другие услуги);
* Указать местонахождение объекта, карта или текстовое поле — было бы классно сделать карту, на которой клиент может отметить точкой местонахождение;
* Характеристики объекта, селект — выбор гидротехнического сооружения (пирс, причал, опоры мостов) или плавучего средства (корабль, баржа, катер), или указание своего варианта;
* Площади обследования — в квадратных метрах;
* Мутность воды — в метрах видимости;
* Течение воды в области обследования — в м/с,
* Характера дна — селект (илистое, щебень, песок, растительность или свой вариант);
* Желаемые сроки проведения работ — текстовое поле или календарь, в котором можно указать дни, в которые нужно провести работы;
* Количество объектов;
* Дополнительное описание объекта, текстовое поле.

Где-то внизу рядом с кнопкой «Оформить» нужен подсчет примерной суммы заказа по нажатии на кнопку оформить — вывод сообщения на весь экран что с вами свяжутся менеджеры.

## Административное управление через Django (со стороны компании)

Эта страница нужна для того, чтобы управлять заказами клиентов (менять статус выполнения, указывать планируемую дату проведения работ), редактировать параметры и стоимость, добавлять 3D-модели и отчеты. Также данная страница нужна для того, чтобы добавлять выполненные проекты в портфолио.

## База данных

В базе данных надо хранить информацию о клиентах и их заказах: требования к заказу, процесс создания 3D-модели,

# Структура и управление веб-приложением

Для упрощения создания веб-приложения была создана блок-схема её работы, изображенная на рисунке 2.

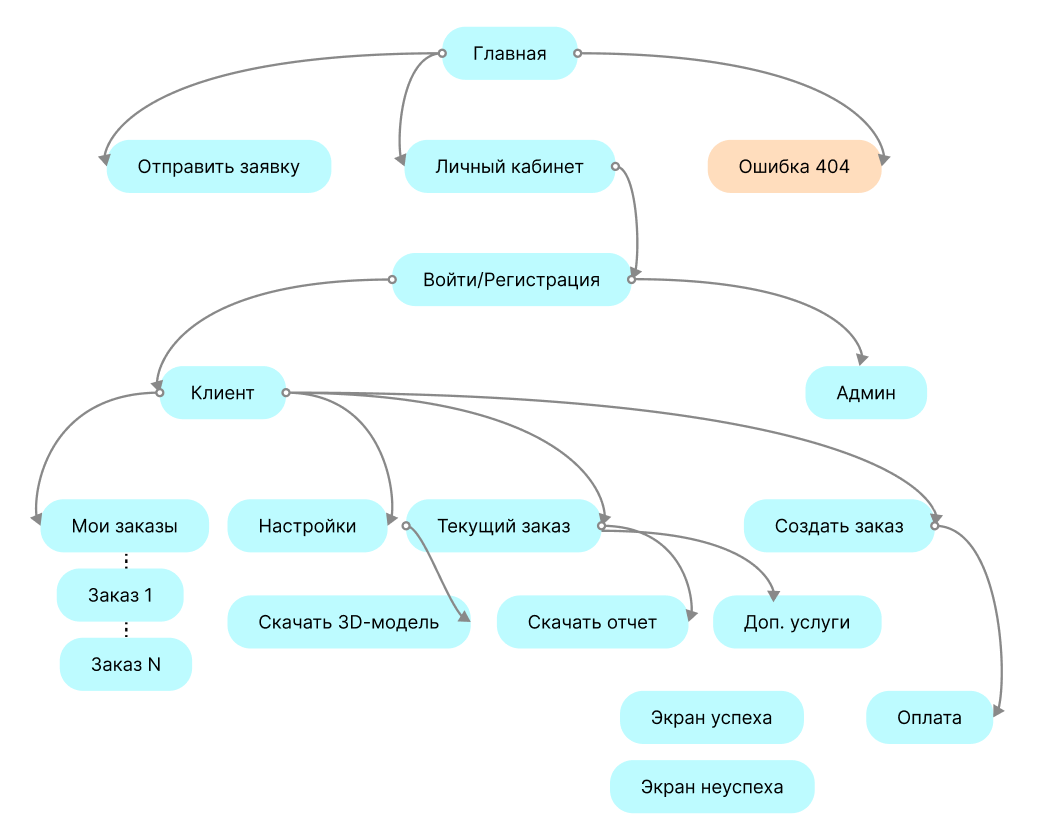


Рисунок 2 – Блок схема работы веб-приложения

На рисунке 3 изображен прототип отображения главной страницы.

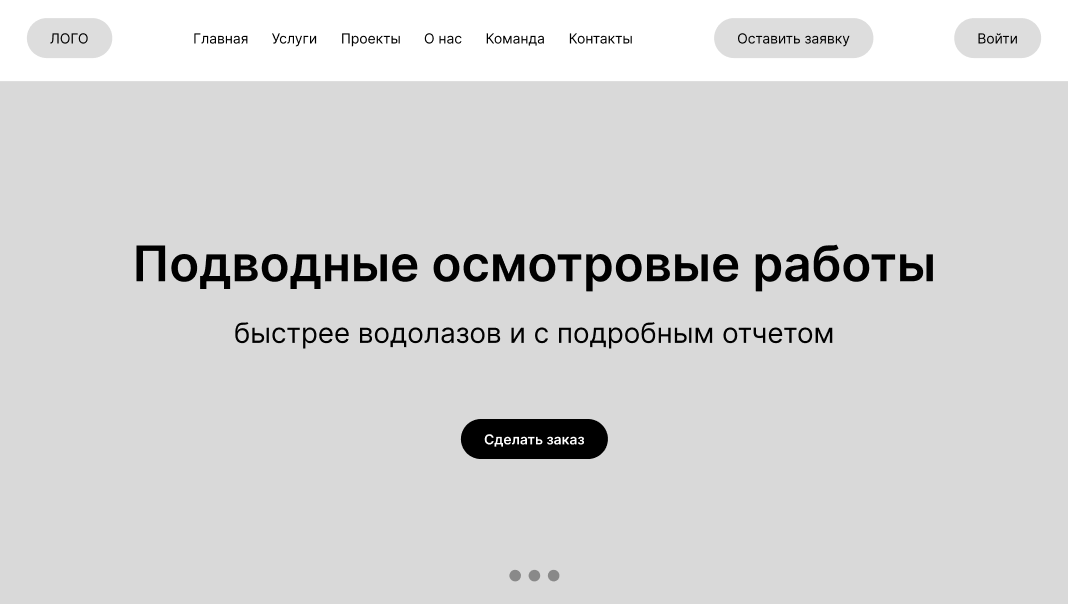


Рисунок 3 – Прототип главной страницы

На рисунке 4 изображен прототип отображения личного кабинета.

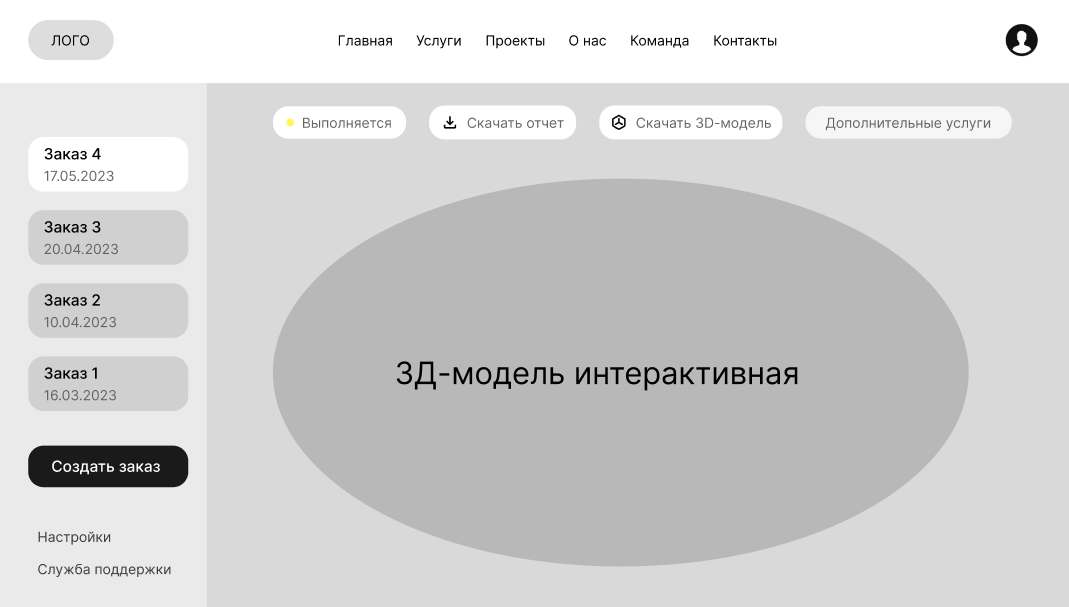


Рисунок 4 – Прототип личного кабинета

На рисунках 5-7 изображен прототип отображения создания заказа в веб-приложении.

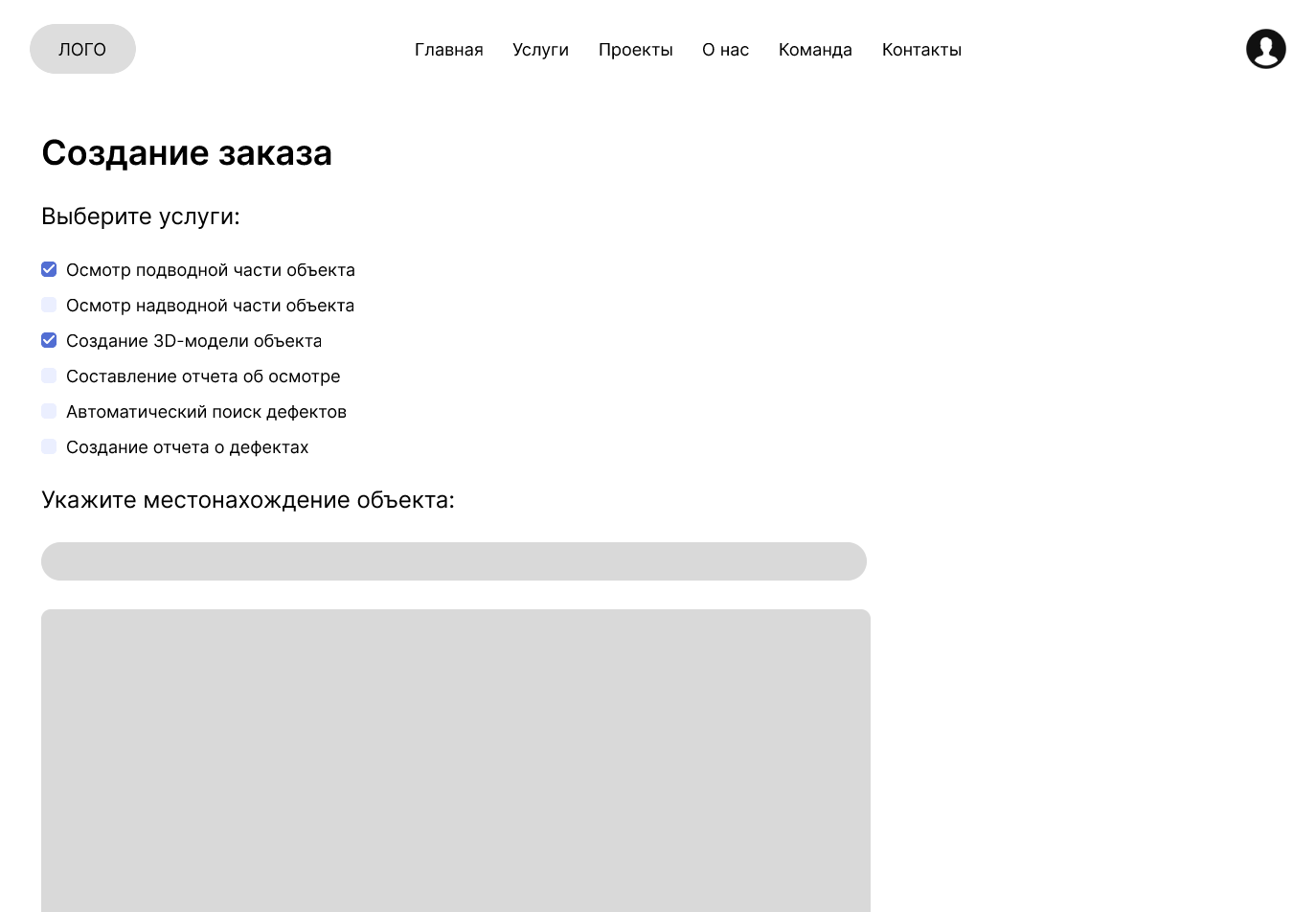


Рисунок 5 – Прототип создания заказа

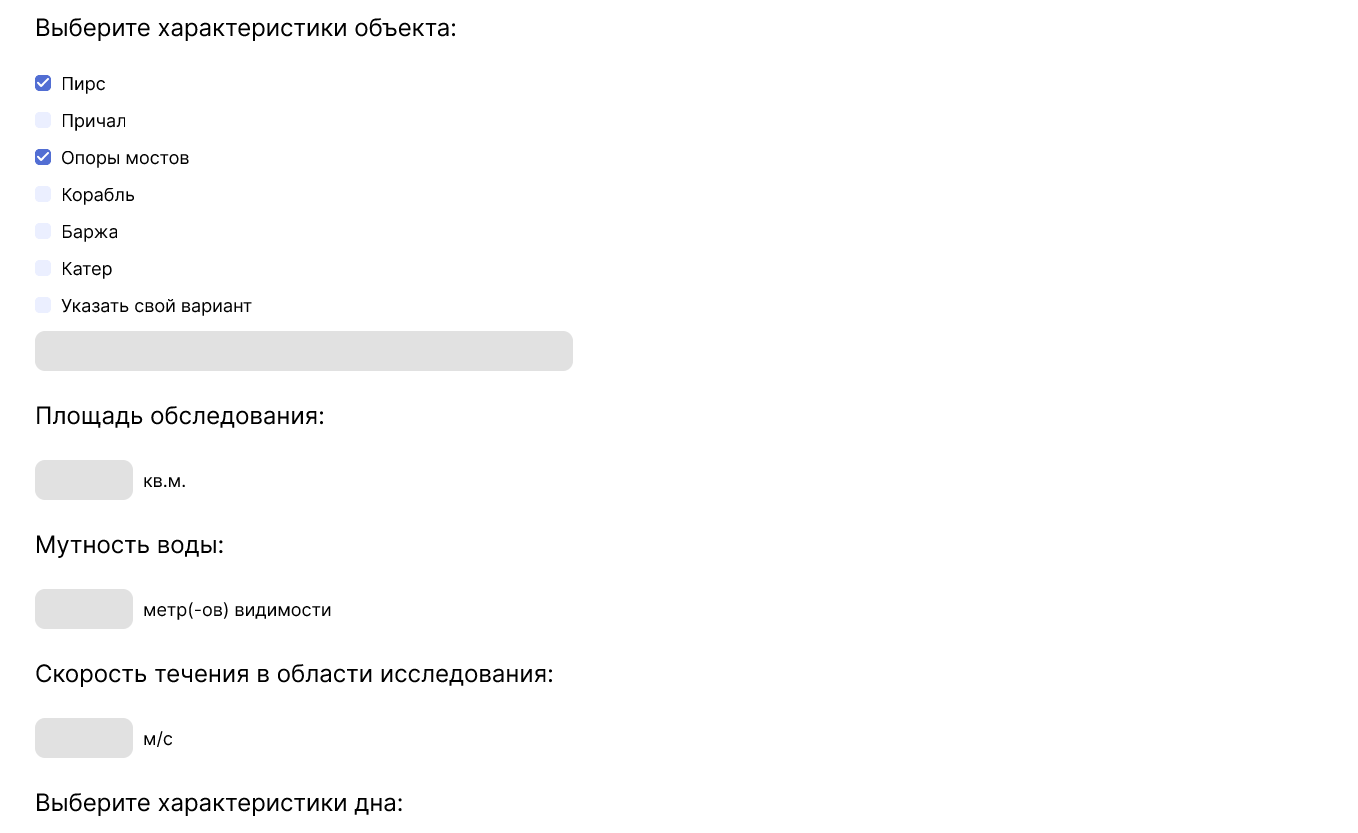


Рисунок 6 – Прототип создания сайта

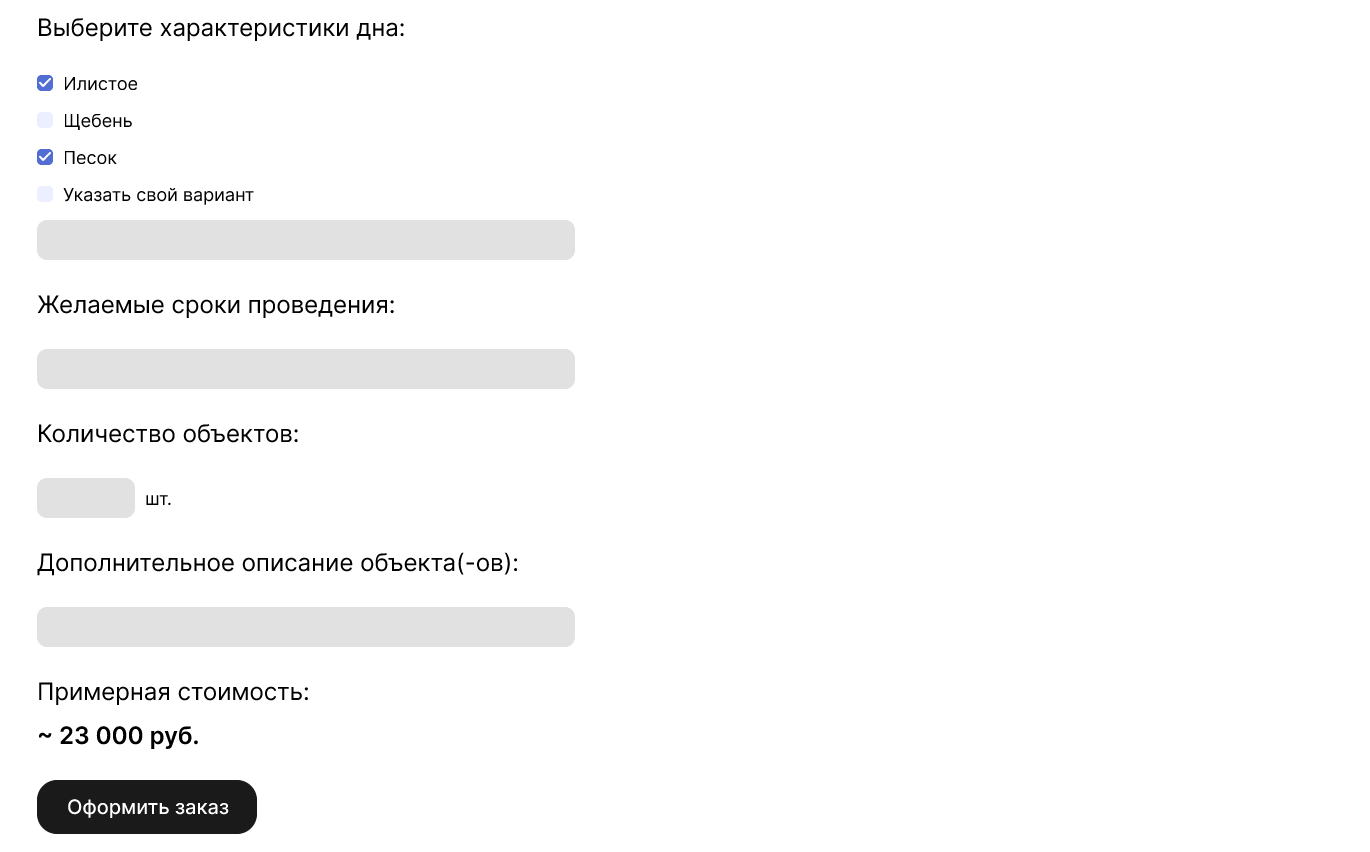


Рисунок 7 – Прототип создания сайта

Блок схема структура веб-приложения изображена на рисунках 8-10.



Рисунок 8 – Структура веб-приложения

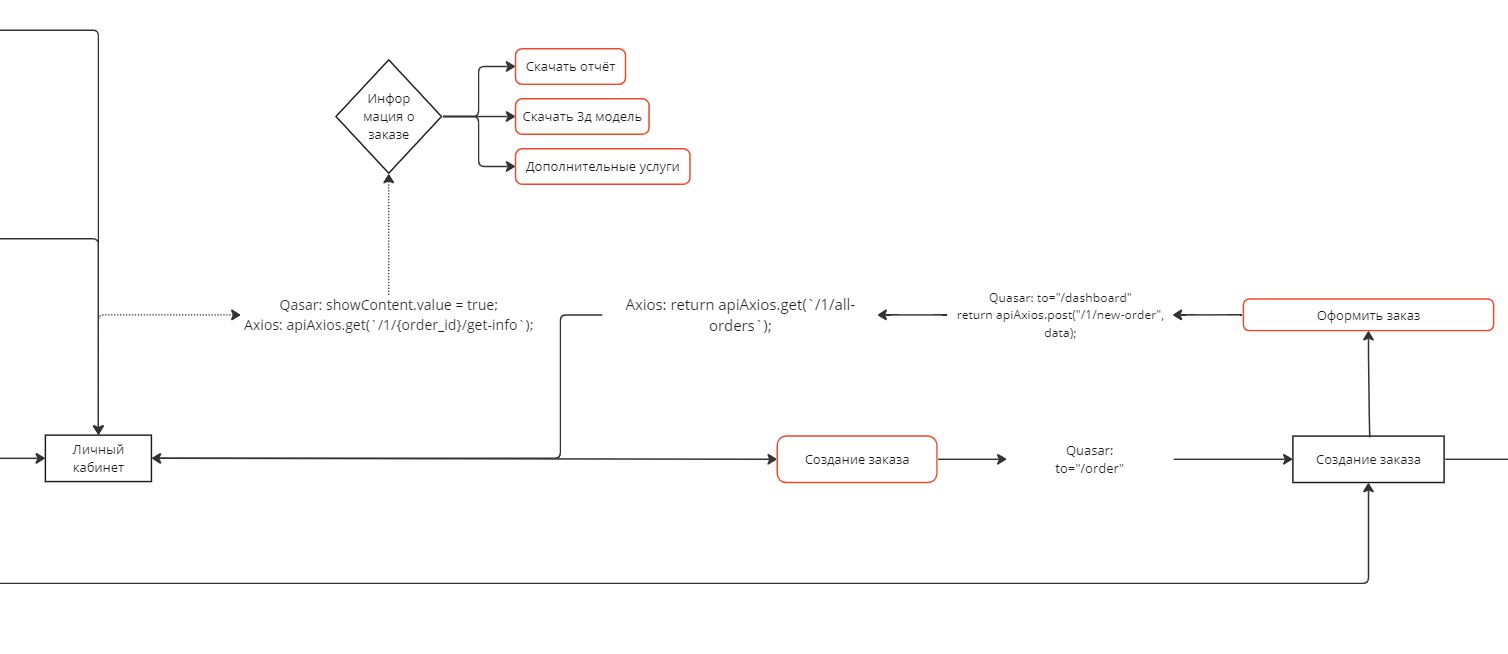


Рисунок 9 – Структура веб-приложения

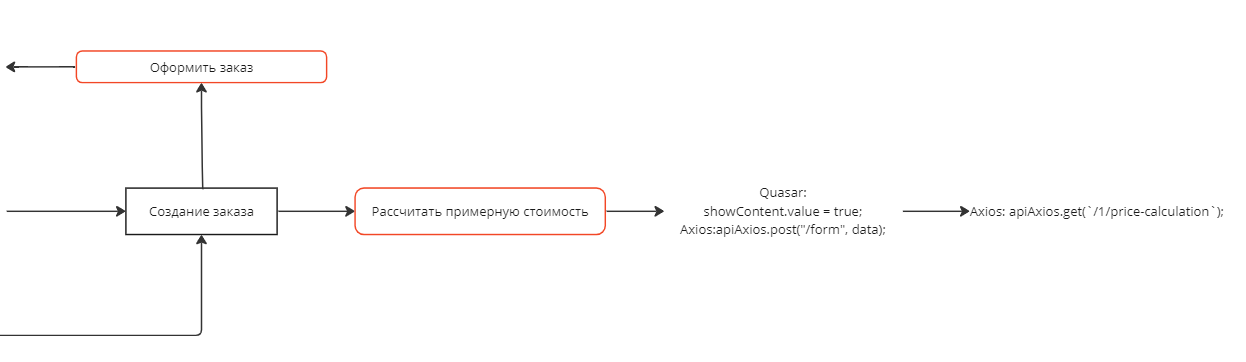


Рисунок 10 – Структура веб-приложения

# Работа веб-приложения

## 5.1. Главная страница

На главной странице сайте есть четыре важные кнопки для перехода, изображенные на рисунке 11:

* Личный кабинет;
* Оставить заявку;
* Войти;
* Сделать заказ.

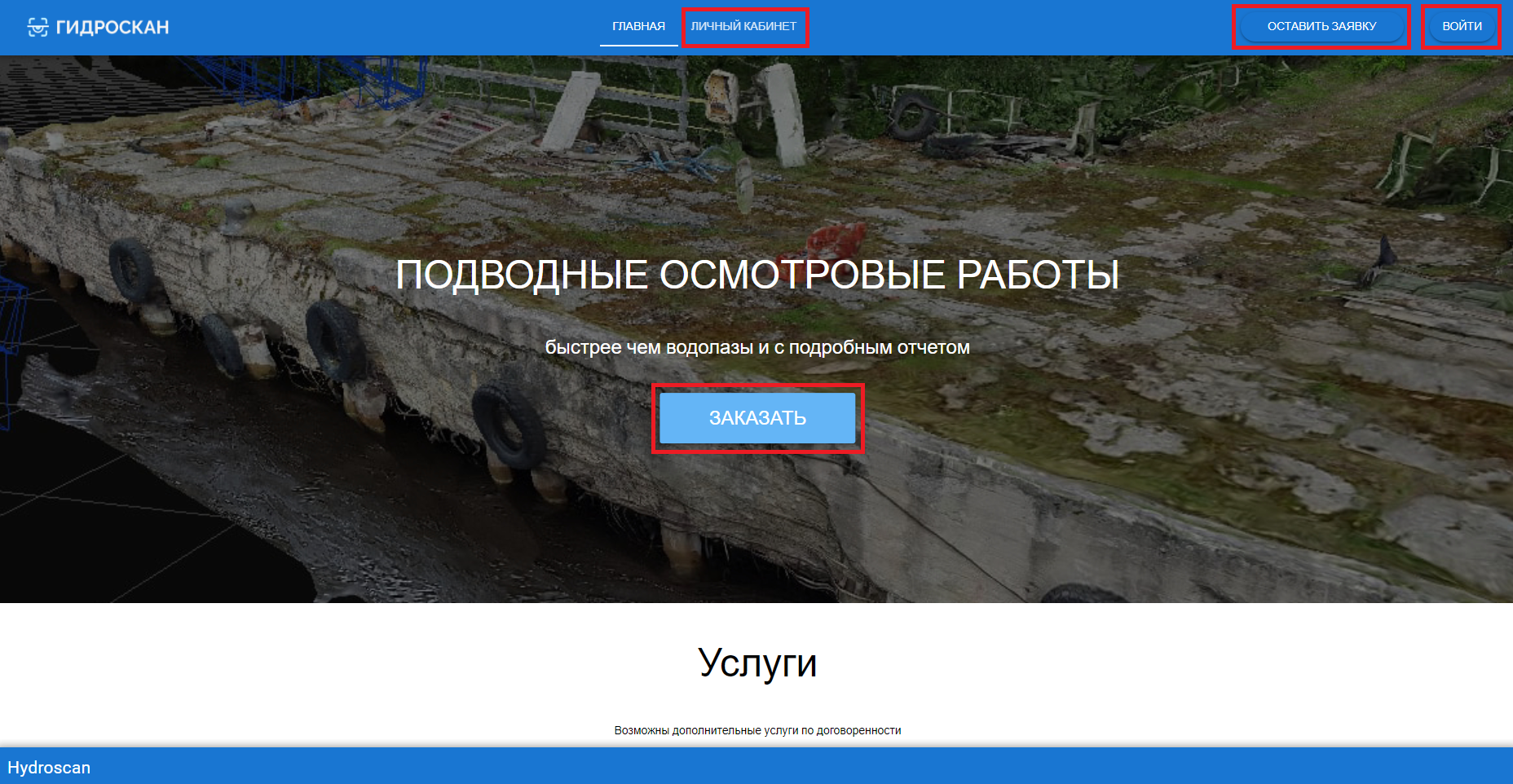


Рисунок 11 – Главная страница

Также на главной странице есть список предоставляемых услуг, продуктов компании, информация, история компании и контактные данные её членов.

Список предоставляемых услуг:

* 3D сканирование: с помощью подводного аппарата, оснащенного камерой высокого разрешения, производится съемка объекта. После обработки данных, получается 3D модель объекта;
* Дефектоскопия: специальная нейросеть анализирует полученную 3D модель и выдает отчет о наличии дефектов;
* Координаты: 3D модель будет привязана к предпочитаемой системе координат;
* Отчеты по ГОСТ: отчеты о проведении работ и результаты их выполнения подготавливаются в соответствии с ГОСТ.

Список предоставляемых продуктов:

* «Кусто 3»: автономный подводный аппарат;
* «Гидроскан»: телеуправляемый подводный аппарат.

Картинки, демонстрирующие другие составляющие главной страницы сайта представлены на рисунках 12-15.



Рисунок 12 – Предоставляемые услуги



Рисунок 13 – Предоставляемые продукты

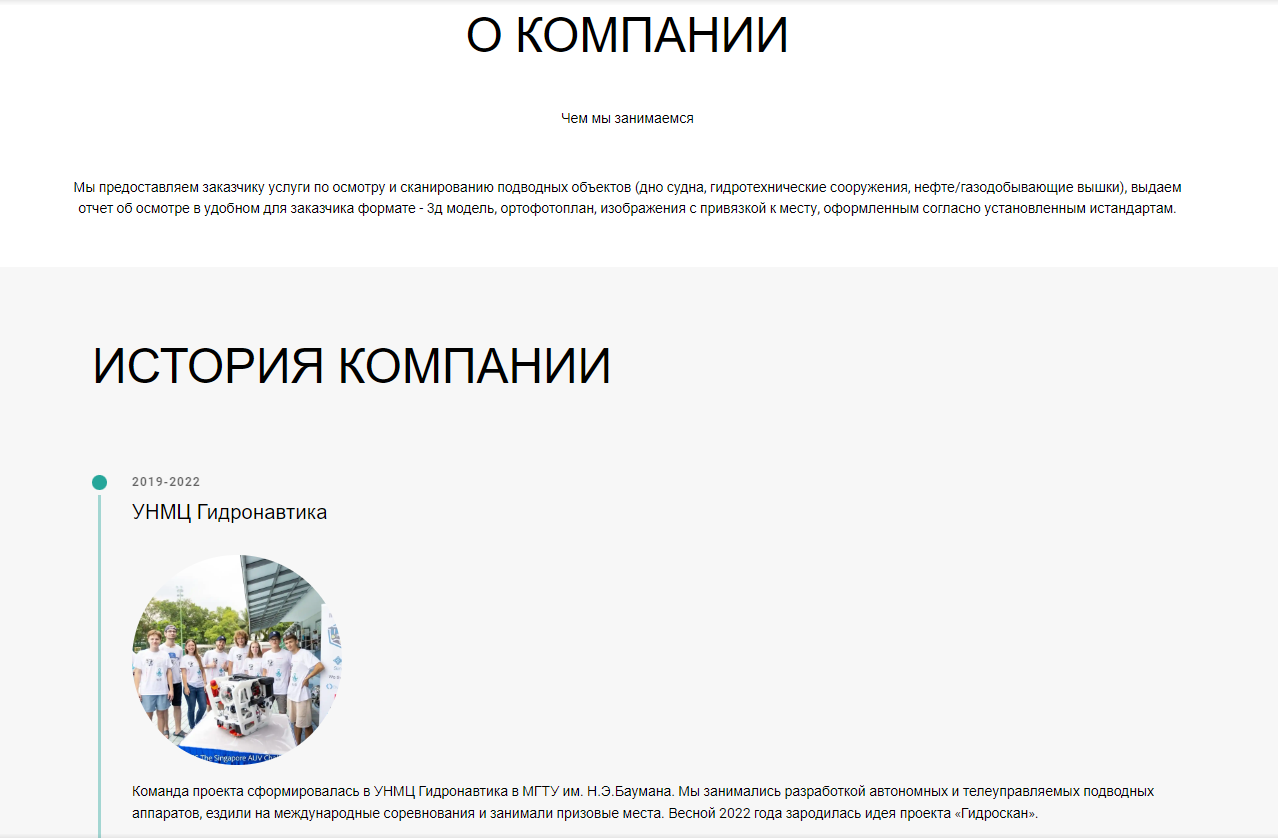


Рисунок 14 – Информация о компании «Hydrocan»

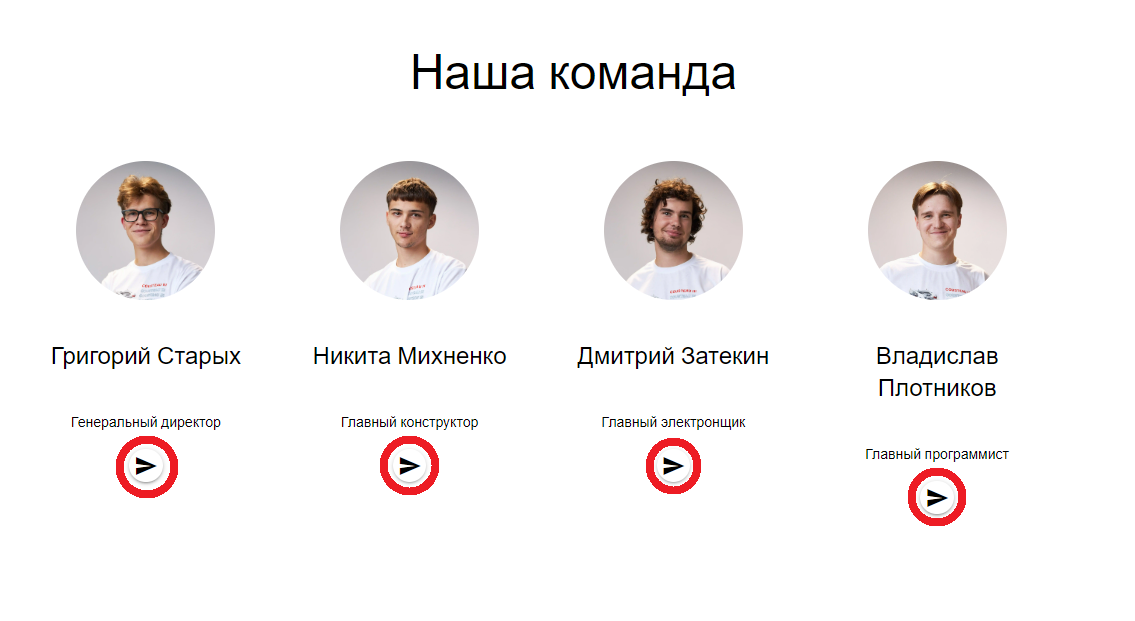


Рисунок 15 – Контактные данные команды (нажатие кнопок переносит на страницу в Telegram конкретного члена команды)

## 5.2. Личный кабинет

Чтобы войти на страницу личного кабинета, изображенного на рисунке 16, нужно пройти регистрацию или ввести уже зарегистрированные email и пароль.

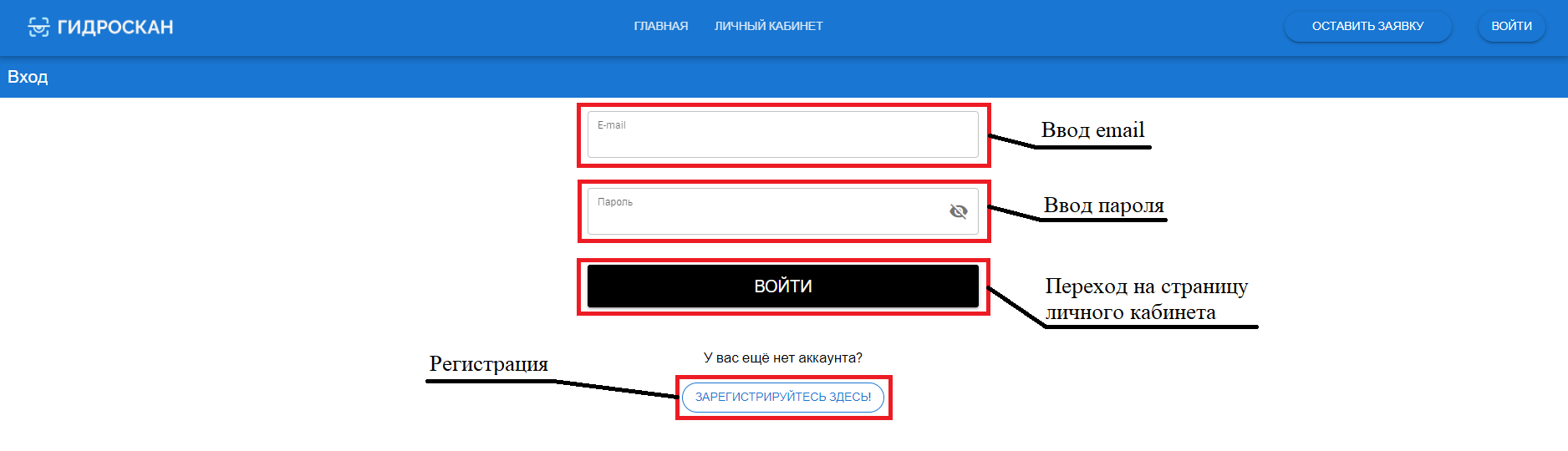


Рисунок 16 – Личный кабинет и ввод данных пользователя

### 5.2.1. Регистрация в личном кабинете

Для регистрации в личном кабинете нужно ввести следующие данные пользователя:

* Имя пользователя (Username);
* Фамилия;
* Имя;
* Email;
* Пароль.

На рисунке 17 изображена страница регистрации.

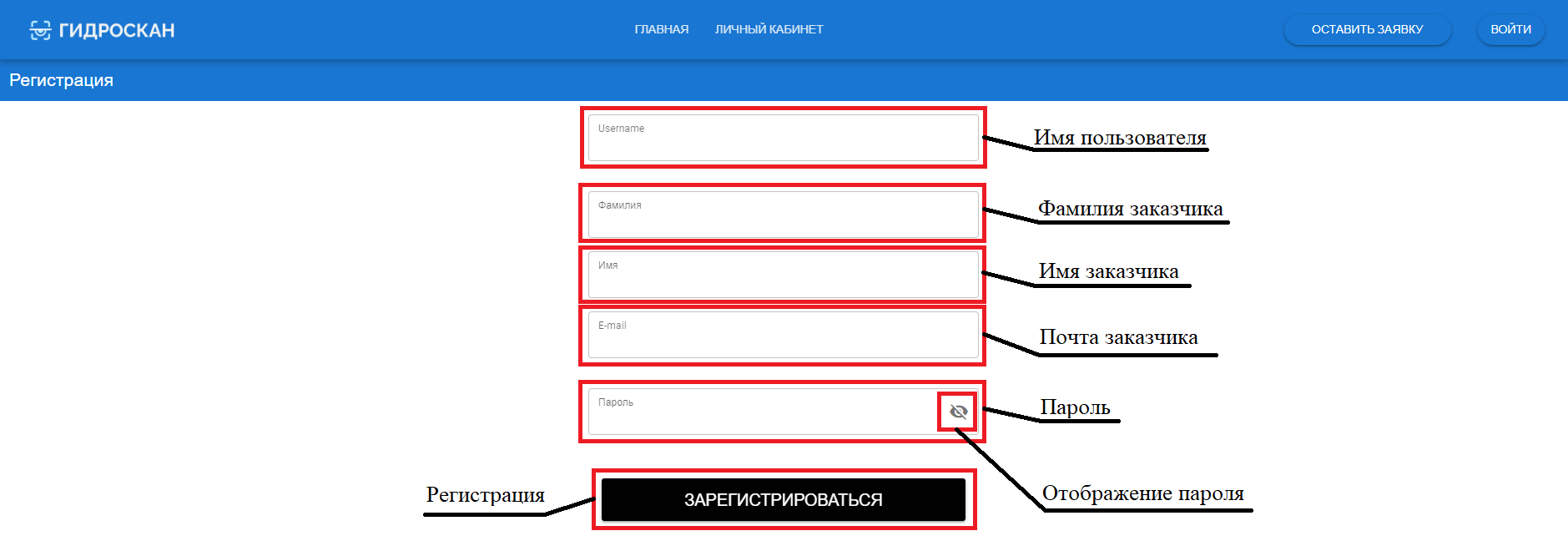


Рисунок 17 – Страница регистрации пользователя

После регистрации пользователь может сделать заказ на подводные осмотровые работы.

## 5.3. Страница заказа

После входа на страницу личного кабинета, пользователь может сделать следующие действия, изображенные на рисунке 18:

* Перейти на главную страницу сайта;
* Просмотреть уже сделанные заказы;
* Создать заказ;
* Настройки;
* Оставить заявку
* Выйти (покинуть страницу личного кабинета).

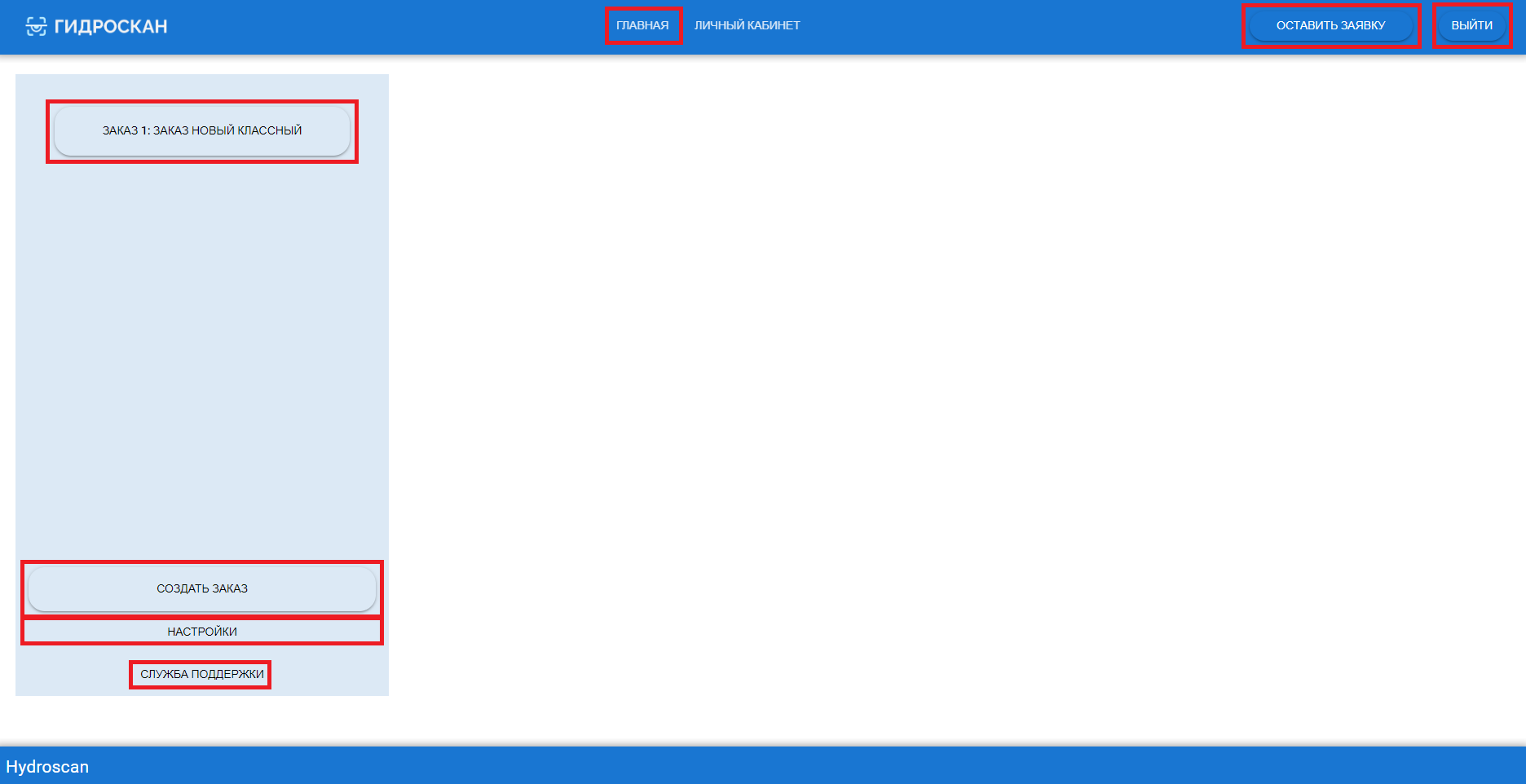


Рисунок 18 – Действия в личном кабинете

Пользователь должен задать следующие параметры для заказа на странице, изображенной на рисунках 19-22:

* 1. Название заказа;
  2. Выбор услуги:
* Осмотр подводной части объекта;
* Осмотр надводной части объекта;
* Создание 3D-модели объекта;
* Составление отчета об осмотре;
* Автоматический поиск дефектов;
* Создание отчета о дефектах;
  1. Указание местоположения объекта;
  2. Выбор характеристик объекта:
* Пирс;
* Причал;
* Опоры мостов;
* Корабль;
* Баржа;
* Катер;
* Другое;
  1. Площадь обследования (в квадратных метрах);
  2. Мутность воды (метров видимости);
  3. Скорость течения области исследования (в м/с);
  4. Выберите характеристики дна:
* Иглистое;
* Щебень;
* Песок;
* Другое;

9) Желаемы сроки проведения;

10) Дополнительное описание объекта(-ов).

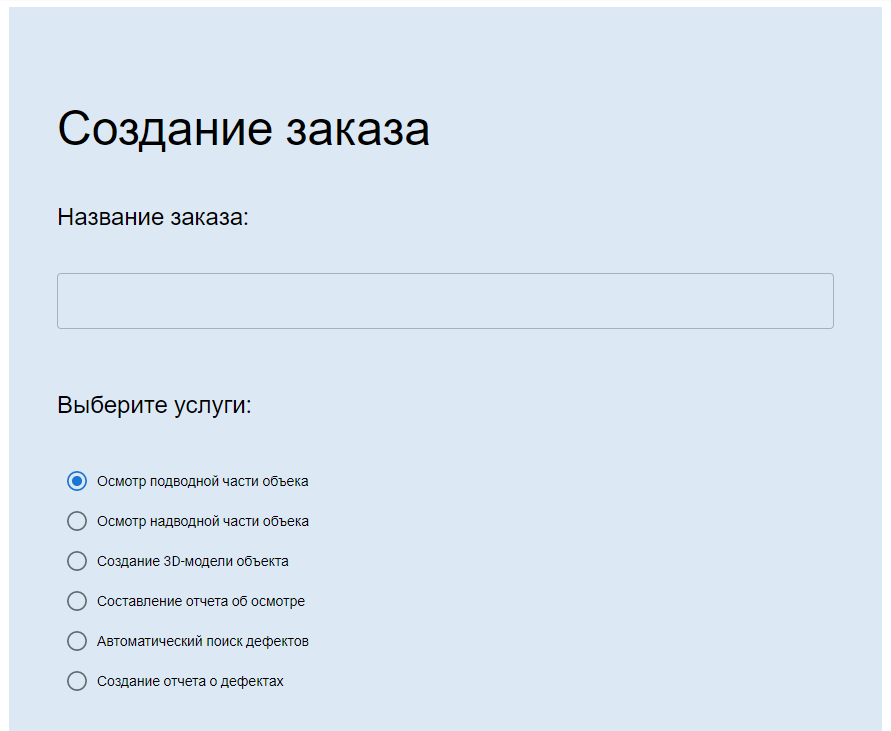


Рисунок 19 – Параметры заказа

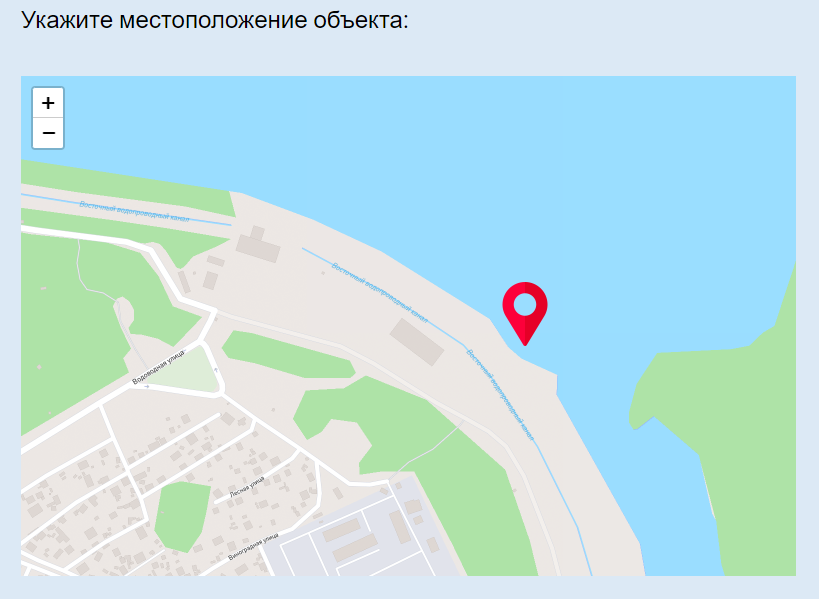


Рисунок 20 – Параметры заказа

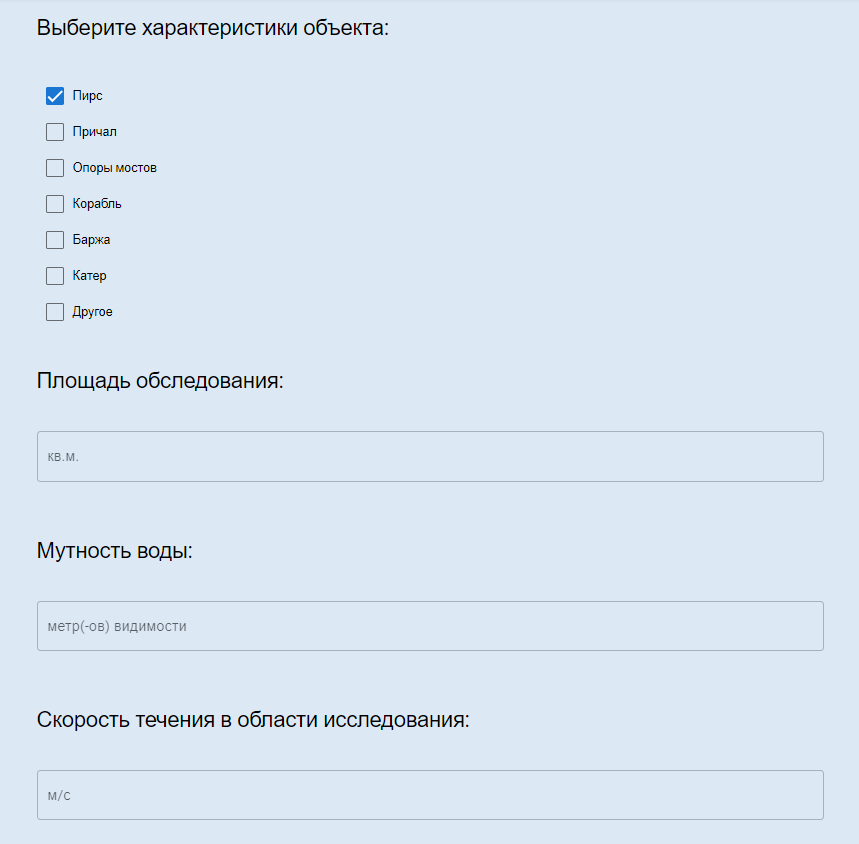


Рисунок 21 – Параметры заказа

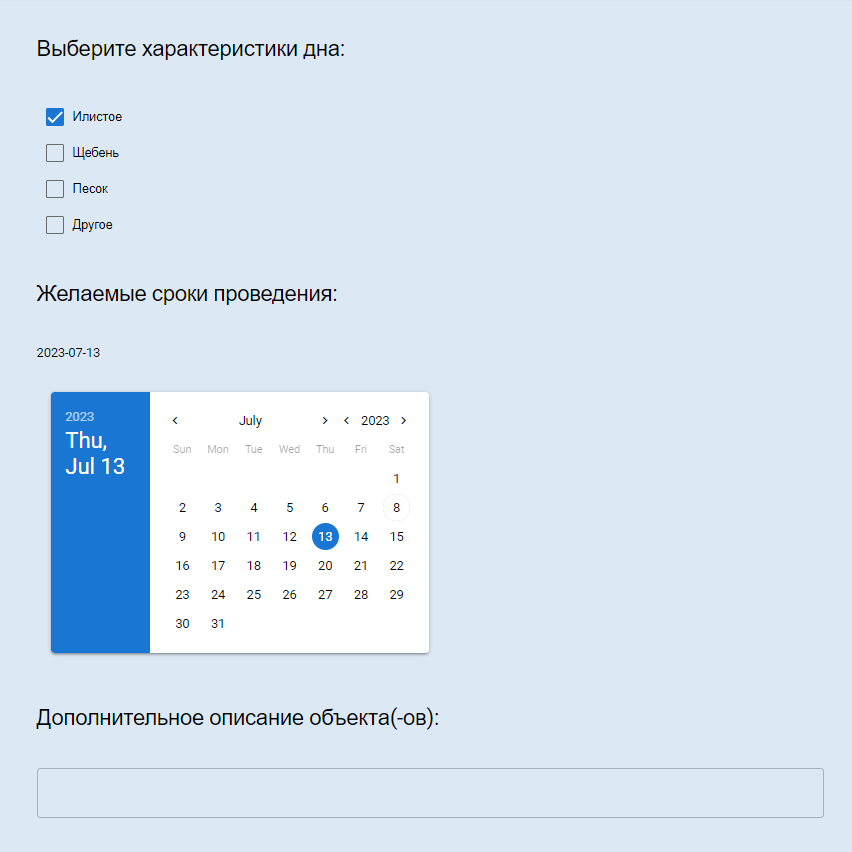


Рисунок 22 – Параметры заказа

После ввода параметров заказа его можно оформить и рассчитать примерную стоимость услуги (Рисунок 23).

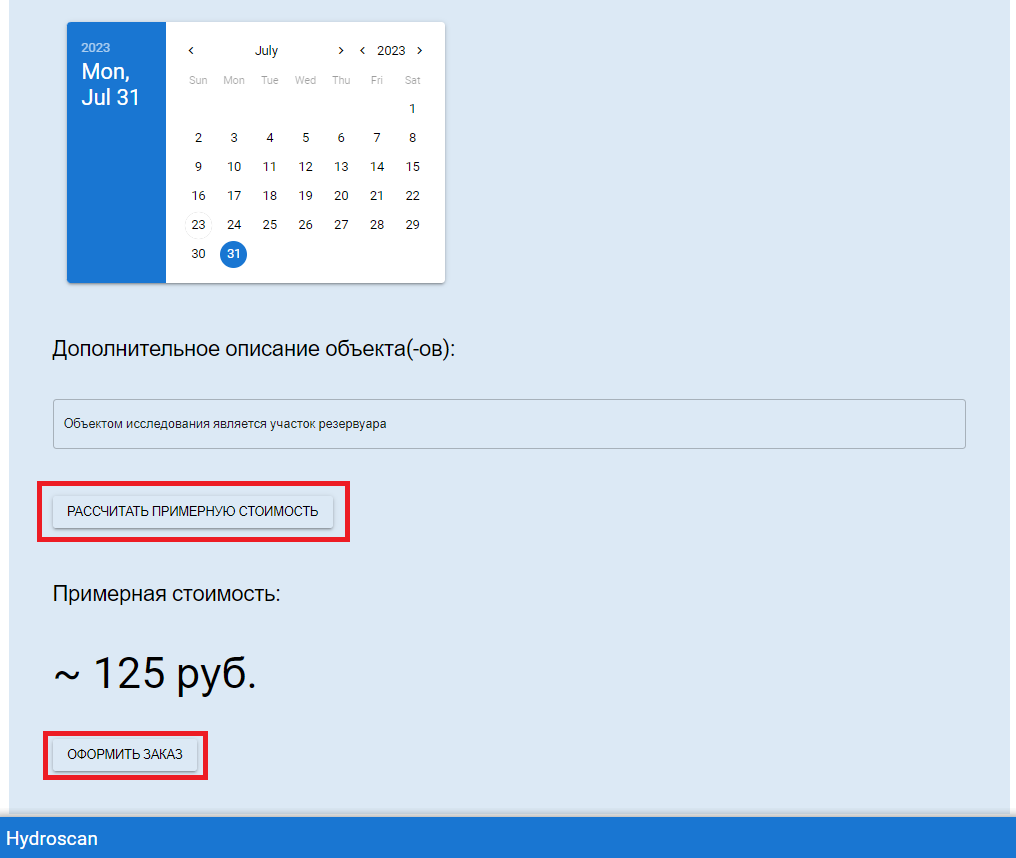


Рисунок 23 – Оформление заказа

Информация о заказе отображается в личном кабинете: можно отследить стадии его выполнения, скачать отчет о проделанной работе, 3D-модель исследуемого объекта, узнать конечную стоимость работы (рисунок 24). Также в личном кабинете можно просмотреть полученную 3D-модель; файл с моделью скачивается в формате PLY (формат файлов описания геометрии, известный также как Polygon File Format и Stanford Triangle Format). Нажатием кнопки «Настройки», пользователь вернется на страницу заказа, где сможет изменить параметры оказываемой услуги.

|  |  |
| --- | --- |
| C:\Users\asaar\Desktop\Проект по вебу\24.png | Рисунок 24 – Просмотр заказа |

## Страница администратора сайта

Для управления данными сайта и заказами пользователей, есть страница администратора, изображенная на рисунке 25.

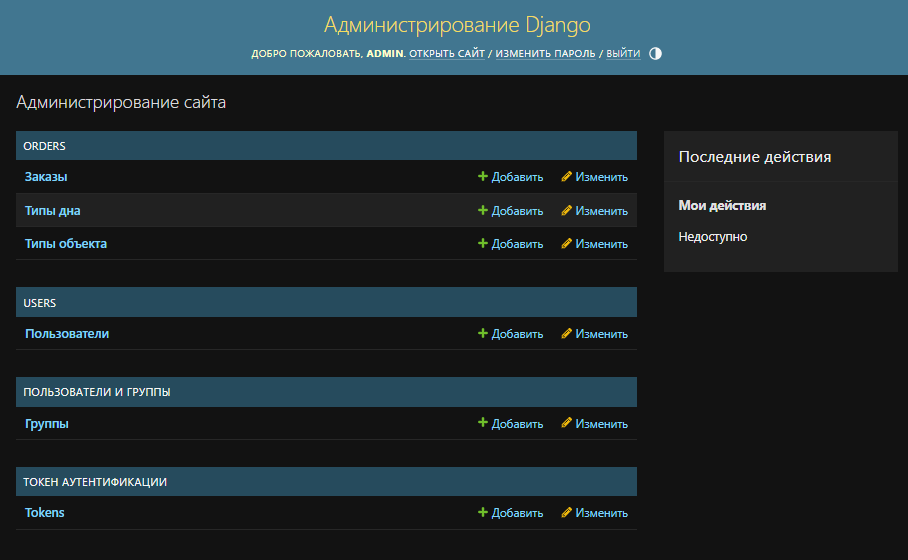


Рисунок 25 – Страница администратора

С данной страницы администратор вносит данные о заказе, редактирует параметры сайта, управляет данными пользователей (рисунок 26).

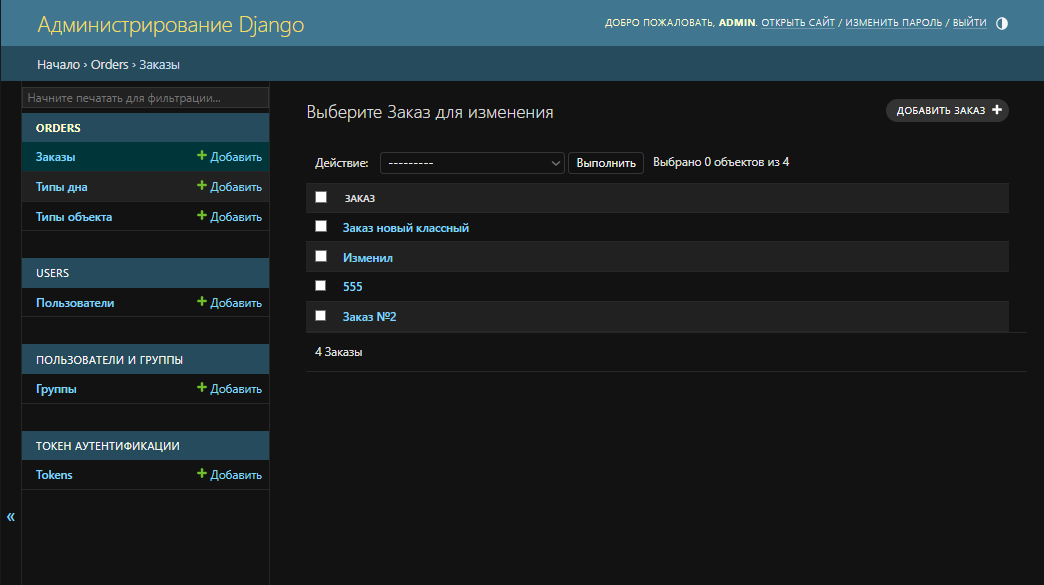


Рисунок 26 – Управление данными

Таким образом, после редактирования администратором деталей заказа (рисунки 27 и 28), данные о работе обновятся на основном сайте (рисунок 29).

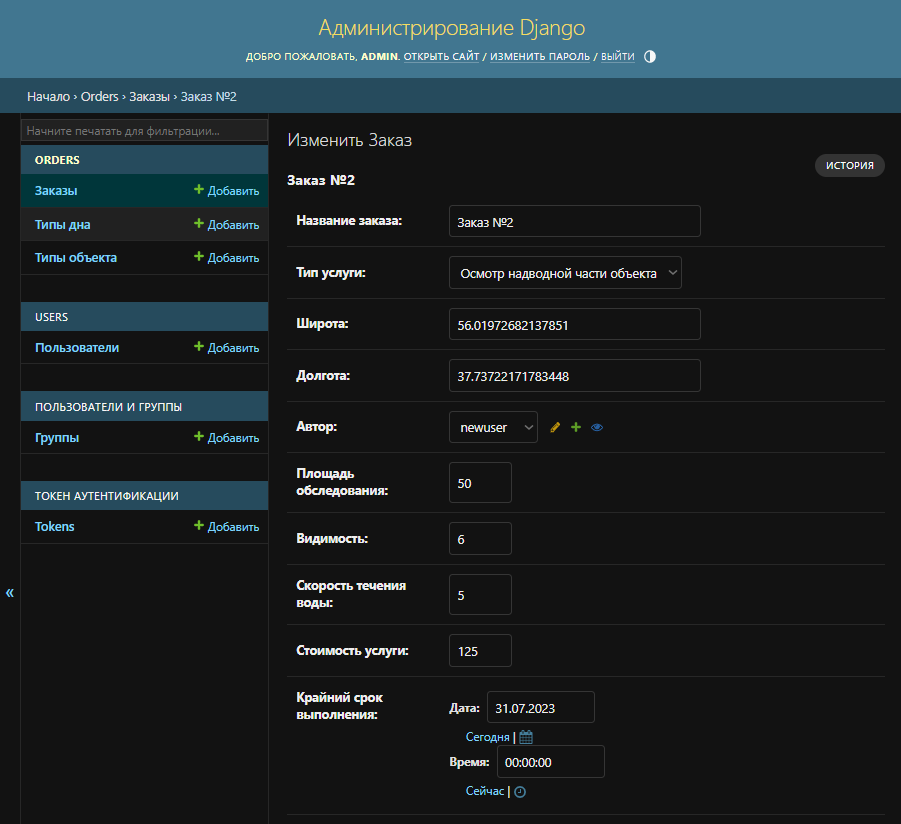


Рисунок 27 – Редактирование деталей заказа

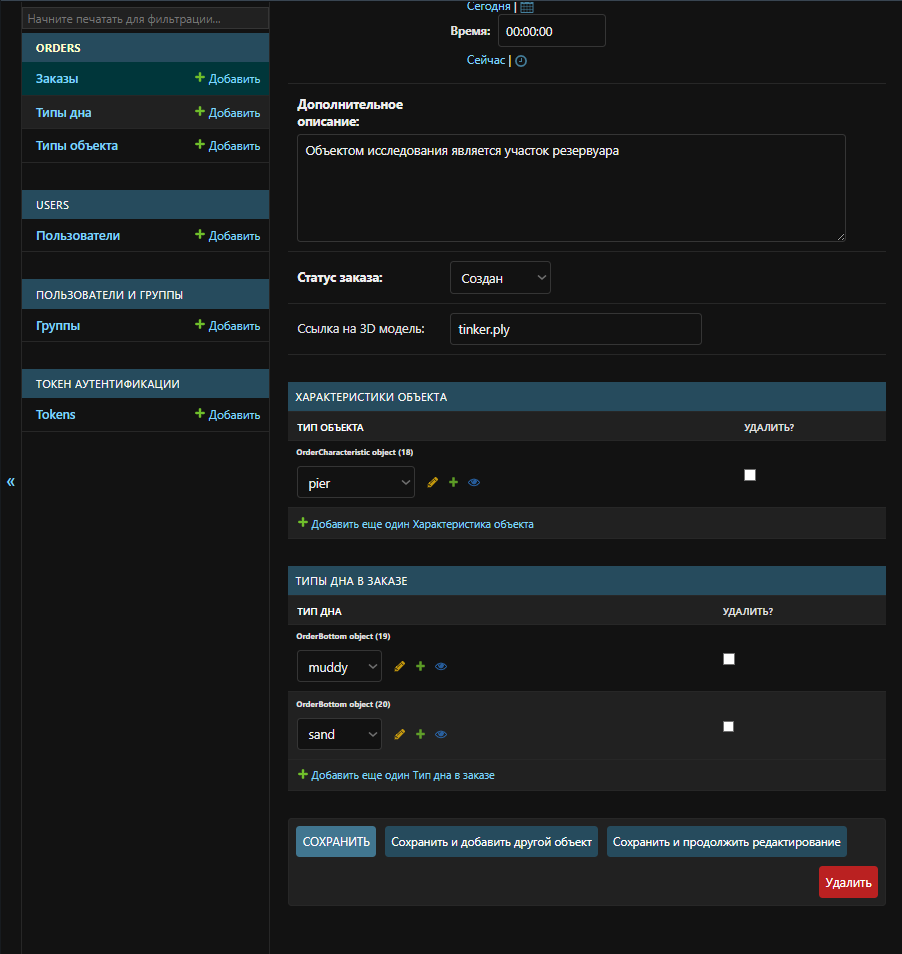


Рисунок 28 – Редактирование деталей заказа

|  |  |
| --- | --- |
| C:\Users\asaar\Desktop\Проект по вебу\29.png | Рисунок 29 – Изменение параметров заказа |

Пользователь также может скачать отчет в формате «pdf» о проделанной работе, нажав на кнопку «Скачать отчёт». Пример отчета представлен   
на рисунке 30.

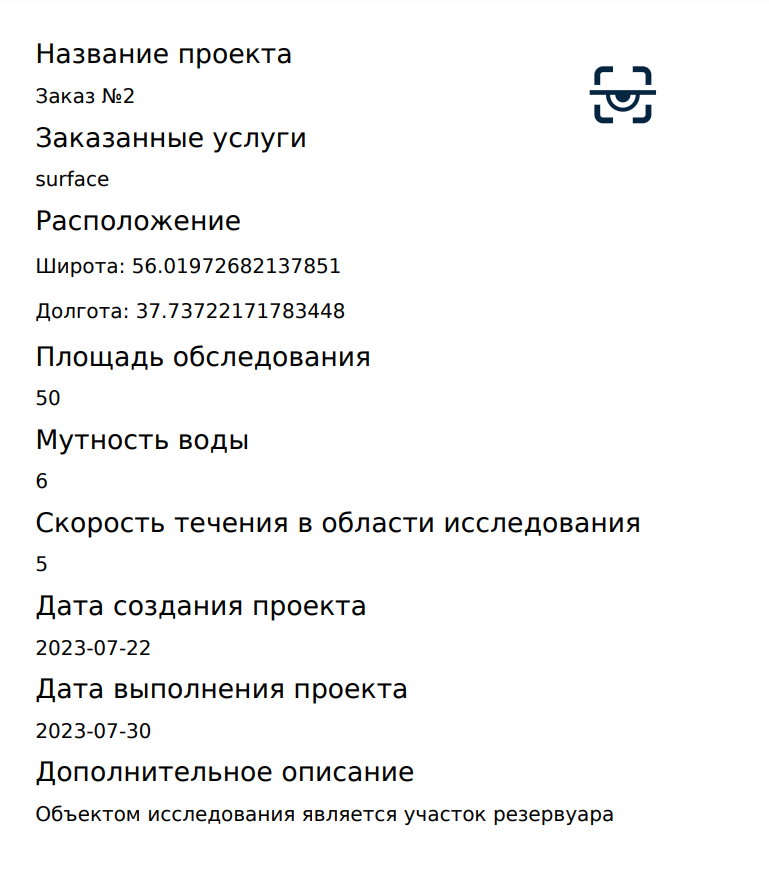


Рисунок 30 – Пример отчета