**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**КЕМЕРОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**ИНСТИТУТ ЦИФРЫ**

**ОТЧЁТ**

**О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ**

«**Разработка требований к информационной системе**»

Студентов 3 курса, группы ФИТ-211

**Малоштановой Ольги Алексеевны,**

**Шабалиной Анны Николаевны,**

**Колесник Полины Олеговны**

Направление 02.03.02 – «Фундаментальная информатика и информационные технологии»

Руководитель:

Кандидат технических наук,

доцент Завозкин С. Ю.

Работа защищена

« »

“ ” 2024 г.

Кемерово 2024 г.



Рис.1. Система “MedLab”

**Описание:**

Главная идея заключается в том, чтобы создать единую систему, которая объединит действующие лаборатории города в одном приложении. В настоящее время существует большое количество лабораторий, которые специализируются на проведение медицинских анализов. Однако, есть недостаток данной системы, так как за каждым результатом необходимо ехать или искать на различных сайтах лабораторий, где осуществлялся сбор. Цель системы состоит в том, чтобы создать единую платформу, которая будет объединять все лаборатории города, чтобы все результаты анализов были доступны в одном месте. Это облегчит процесс получения, хранения и анализа результатов анализов пациентов, а также повысит качество предоставления медицинских услуг в городе.

Чтобы сформировать и проанализировать требования для системы, необходимо рассмотреть диаграммы опорных точек зрения на основании метода VORD :



Рис. 2. Диаграмма идентификации точек зрения



Рис. 3. Диаграмма иерархии точек зрения

Данные диаграммы представляют различные точки зрения участников системы, которые помогают выделить важные роли и сервисы, а также взаимосвязи между ними.

В нашей системе присутствует три основные точки зрения, у каждой из которых присутствуют свои сервисы и обязанности:

* “Сотрудники”;
* “Лаборатории”;
* “Пользователи”;

Информация о каждой точке зрения

**Точка зрения “Лаборатории”** необходима для нашей системы, так как именно она заносит в свои медицинские карточки данных о сделанных анализах, которые мы будем требовать по API сверяя данные, чтобы другим пользователям не показывались чужие результаты. Также лабораторий предоставляют список услуг, которые они могут оказать пользователям.

**Точка зрения “Сотрудники”** разделяется на “Администратор”, “Консультант”.

“Администратор” системы будет иметь такие возможножности как:

* Управление пользователями. Он может создавать, редактировать и удалять учётные записи пользователей.
* Мониторинг и анализ данных. Администратор будет отслеживать статистику использования системы, выявление ошибок и проблем.
* Управление результатами анализов. Получение результатов анализов через API, их хранение и отображение в системе, обеспечение безопасности и конфиденциальности данных.
* Управление лабораториями в системе. Добавление новых лабораторий в систему, обновление информации о существующих лабораториях,
* Управление платформой. Обновление приложения, исправление ошибок, обеспечение безопасности данных и защиты от несанкционированного доступа

Роль “Консультант” помогает пользователям, отвечая на интересующие их вопросы по работе системы. То есть для него предоставлена возможность: ответ на вопросы пользователей, помощь в решении проблем, обучение новых пользователей работе с системой.

**Точка зрения “Пользователи”** также разделяется на “Ребёнок” и “Взрослый”.

Для точки зрения “Ребёнок” пользователи должны находиться в возрастной категории до 18 лет. Они не могут сами записаться на сбор анализов. Для них предоставлена возможность просматривать свои результаты, а также скачивать результаты на свой мобильный телефон.

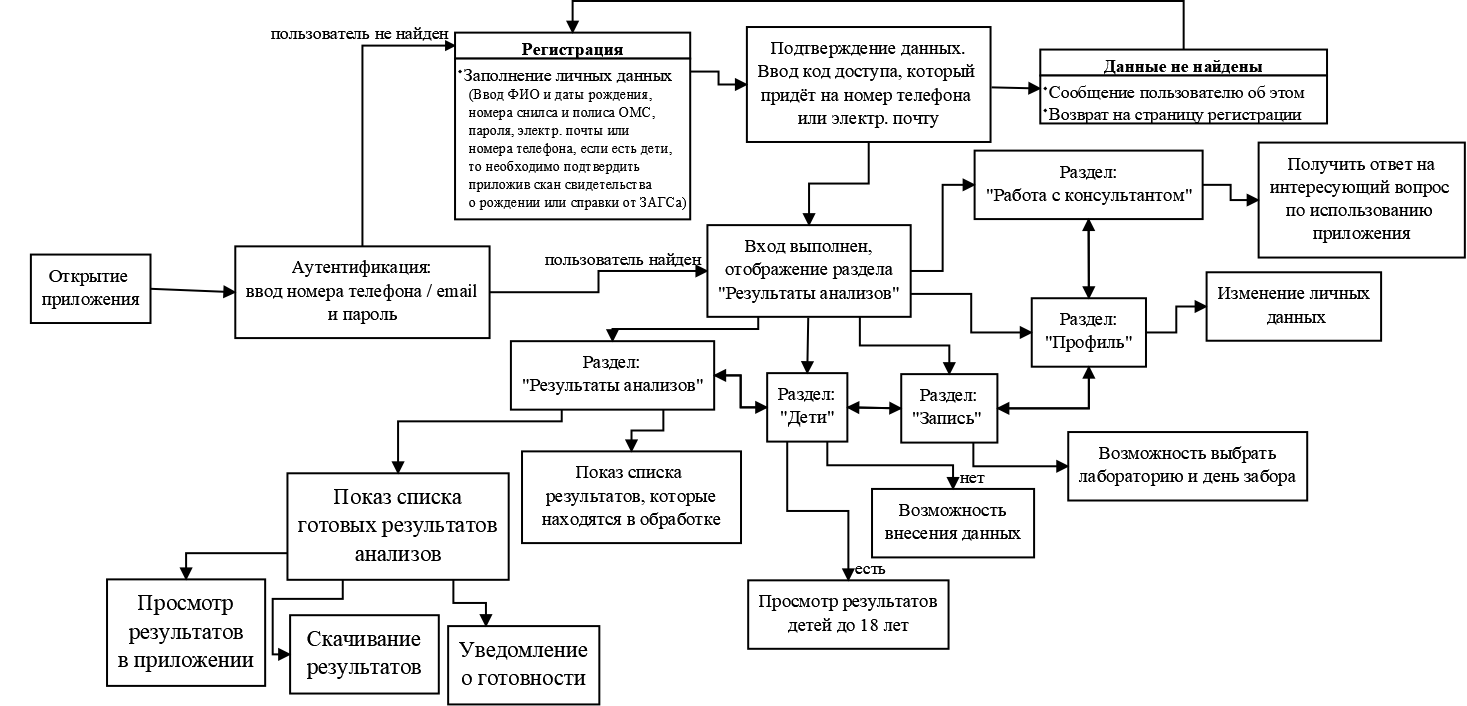
Для точки зрения “Взрослый” пользователи должны находиться в возрасте старше 18 лет. Есть возможность скачивать результаты на свой мобильный телефон или просматривать в приложении. Для них необходимо добавить раздел для записи в лабораторию, а для “Взрослых” с детьми нужно учесть выбор кого он будет записывать (себя или ребёнка).

Таким образом, данные требования для каждой точке зрения позволит разработчикам создать систему, которая будет удовлетворять потребности каждой из сторон, обеспечивая им удобство использования и эффективность взаимодействия с системой.



Рис. 4. Информационная модель

Данная модель отражает, что пользователь записывается в лабораторию на сдачу анализа. Лаборатория же в свою очередь предоставляет перед пользователем свой список услуг. После сдачи анализа, результаты записываются в карточку и связываются с пользователем и услугой.

Рис. 5. Сценарий событий

Возможный исход событий: при открытии приложения открывается окно аутентификации, где у пользователя есть выбор либо ввести номер телефона или электронную почту и пароль либо зарегистрироваться в системе. Если пользователя нет в системе, то перед пользователем появляется окно регистрации. Там необходимо ввести личные данные человека, такие как ФИО, даты рождения, номера снилса и полиса ОМС, номера телефона, адрес электронной почты и пароль с подтверждением. Если у человека есть ребёнок, то необходимо предоставить скан свидетельства о рождении, чтобы до достижения 18 лет, результаты показывались в карточке родителя.

Когда пользователь входит в систему перед ним появляется раздел “Результаты анализов”. В котором находятся два списка:

* Список готовых результатов анализов.
* Список результатов, которые находятся в обработке.

Каждый из этих списков представляет из себя короткую информацию об анализе, такую как идентификатор, тип и дату сдачи. В списке готовых результатов анализов есть возможность просмотреть полную информацию о результате в самом приложении, а также сохранить его на свой мобильный телефон. Когда результат готов, пользователю приходит оповещение об этом.

На каждом из окон есть нижняя форма с названиями разделов для удобства навигации по системе. Форма содержит такие разделы как “Результаты анализов”, “Дети”, “Запись”, “Профиль”, “Консультант”.

Если при регистрации указали, что есть дети, то в соответствующем разделе также будет два списка с готовыми результатами и которые в обработке, но просматривать их есть возможность только до совершеннолетнего возраста.

В разделе “Запись” предоставлен список лабораторий и услуг, которые они предоставляют. Здесь можно записаться в любую лабораторию города. Также при записи можно указать кого именно человек хочет записать - себя или ребёнка, если он есть в базе.

В Профиле можно изменить свои личные данные, если они изменились со временем или сменить пароль.

Консультанту можно задать интересующие вопросы по работе системы.

Требования к техническому обеспечению:

* Совместимость с операционной системой Android и iOS;
* Возможность загружать результаты анализов в форматах: PDF, DOCX, DOC;
* Разрешение экрана 720\*1280 пикселей или больше;
* Доступ к интернету через Wi-Fi или мобильную сеть для загрузки приложения;
* Достаточно свободного места на устройстве для сохранения результатов анализов;

### Требования к иллюстрациям:

Все рисунки и фото объемом не более 1mb (кроме элементов дизайна страницы). Все рисунки должны быть в формате png или jpg.

### Требования к языкам программирования:

Для реализации интерактивных элементов клиентской и серверной части должен использоваться язык Dart.

### Требования к хранению данных:

Все данные приложения должны храниться в структурированном виде под управлением реляционной СУБД, таких как PostgreSQL или MySQL. Исключения составляют файлы данных, предназначенные для просмотра и скачивания (результаты). Такие файлы сохраняются в файловой системе, а в БД размещаются ссылки на них.

Наполнение различных разделов, функционирование которых поддерживается одной и той же инсталляцией системы, должно храниться под управлением единой СУБД.

Требования к безопасности данных:

Реализация мер безопасности и защиты данных, включая шифрование данных, использование безопасных протоколов передачи данных и соблюдение стандартов защиты персональных данных.

Разработка системы аутентификации и авторизации для обеспечения безопасного доступа к системе, включая поддержку двухфакторной аутентификации.

Требования к структуре системы:

* Масштабируемость
* Модульность.

Пользовательские требования:

*Для взрослых пользователей:*

* Возможность просматривать результаты всех своих анализов в одном месте
* Возможность записываться на сдачу анализов
* Возможность получать уведомления о готовности результатов анализов

*Для детей:*

* Родители или опекуны могут просматривать результаты анализов своих детей
* Родители или опекуны могут записывать своих детей на сдачу анализов
* Родители или опекуны могут получать уведомления о готовности результатов анализов своих детей

*Для всех пользователей:*

* Быстрый и надёжный доступ к результатам анализов
* Возможность сохранять и распечатывать результаты анализов
* Возможность записываться на медицинские услуги в любую лабораторию города
* Высокий уровень безопасности и конфиденциальности данных

# **Требования к графическому дизайну приложения**

## **Требования к дизайну приложения**

При разработке приложения должны быть использованы преимущественно светлые стили.

Основные разделы приложения должны быть доступны с главной страницы после аутентификации или авторизации в системе.

На первой странице необходимо сделать аутентификацию пользователя.

После входа в систему должна быть основная информация – результаты анализов.

В дизайне приложения не должны присутствовать:

- мелькающие баннеры;  
 - много сливающегося текста;  
 - рекламы;  
 - ненужной информации;

#### **Требования к представлению страниц входа и регистрации в системе.**

Страница входа должна содержать графическую часть, название системы и возможность вписать номер мобильного телефона или адрес электронной почты и пароль, чтобы пользователь смог войти в систему и ознакомиться с основной информацией. В случае если пользователь забыл пароль, то необходимо предусмотреть возможность восстановления пароля.

В случае, если пользователь использует приложение впервые для него предоставляется возможность регистрации в системе. В данном меню будут такие поля ввода как:

- поля фамилии, имени, отчества

- поле даты рождения

- поле места проживания

- выбор пола (женский / мужской)

- поля номера СНИЛСа и полиса ОМС

При наличии ребёнка:

- поля фамилии, имени, отчества

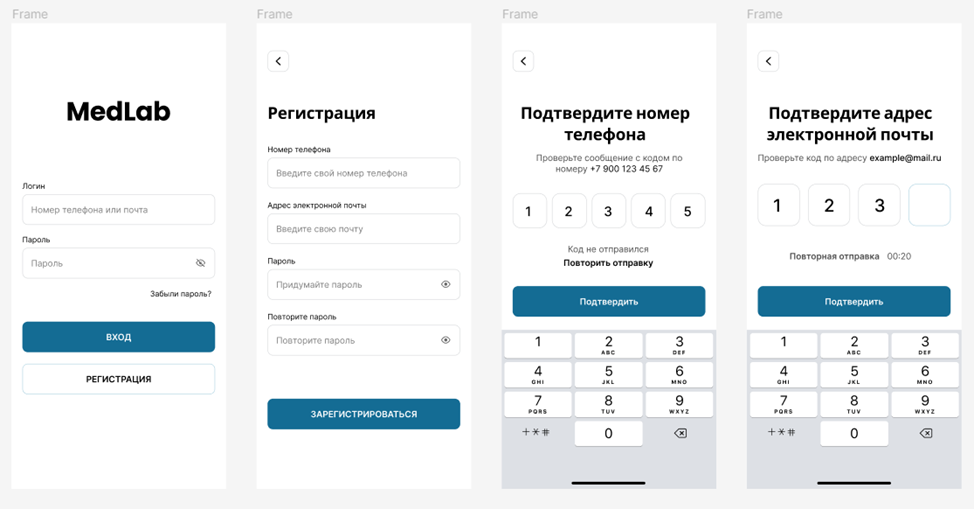
- поле даты рождения

- поле места проживания

- выбор пола (женский / мужской)

- поля номера СНИЛСа и полиса ОМС

- поле для прикрепления скана свидетельства о рождении или справки из ЗАГСа

Рис. 6. Пример размещения элементов страницы входа и регистрации.

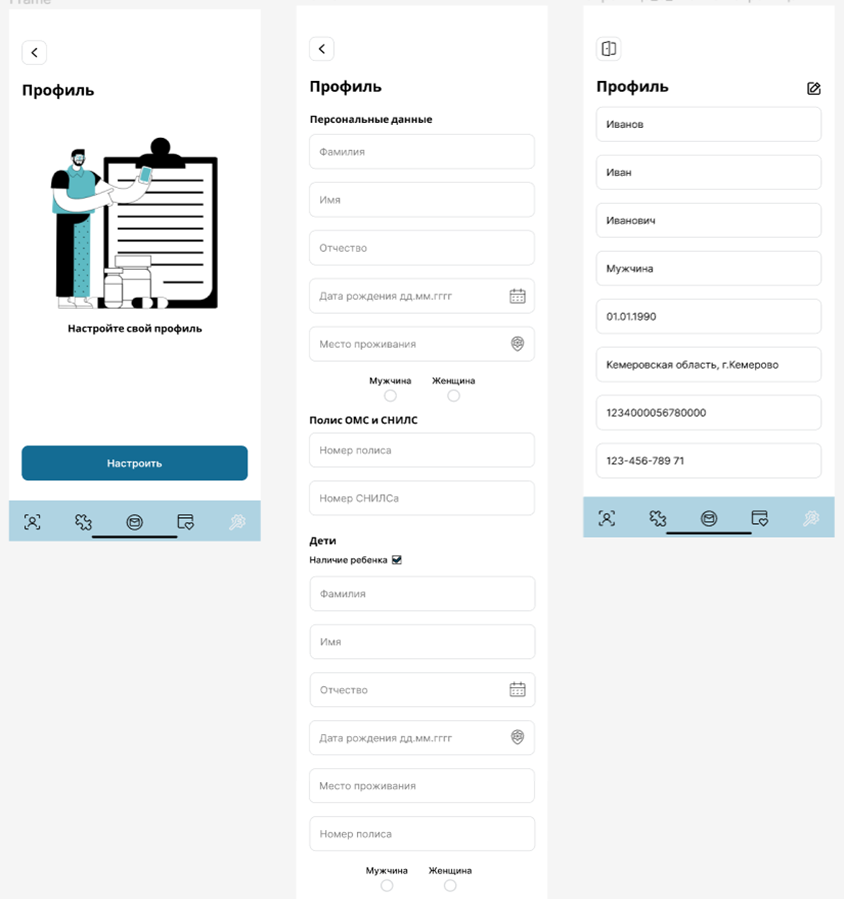


Рис. 7. Пример размещения элементов страницы настройки профиля.

#### **Графическая оболочка внутренних страниц (общая для всех подразделов)**

Графическая оболочка внутренних страниц должна делиться на следующие разделы:

- возможность выйти из системы;

- навигационное меню приложения, которое содержит основные разделы такие как “Результаты анализов”, “Дети”, “Консультант”, “Запись”, “Профиль”;

#### **Графическая оболочка раздела «Результаты анализов»**

В данном разделе должны находиться два основных списка:

- список готовых результатов анализов;

- список результатов, которые находятся в обработке;

Каждый из этих списков должен представлять из себя короткую информацию об анализе, такую как идентификатор, тип и дату сдачи. В списке готовых результатов анализов должна быть возможность просмотреть полную информацию о результате в самом приложении, а также сохранить его на свой мобильный телефон в формате PDF. Когда результат готов, пользователю приходит уведомление.

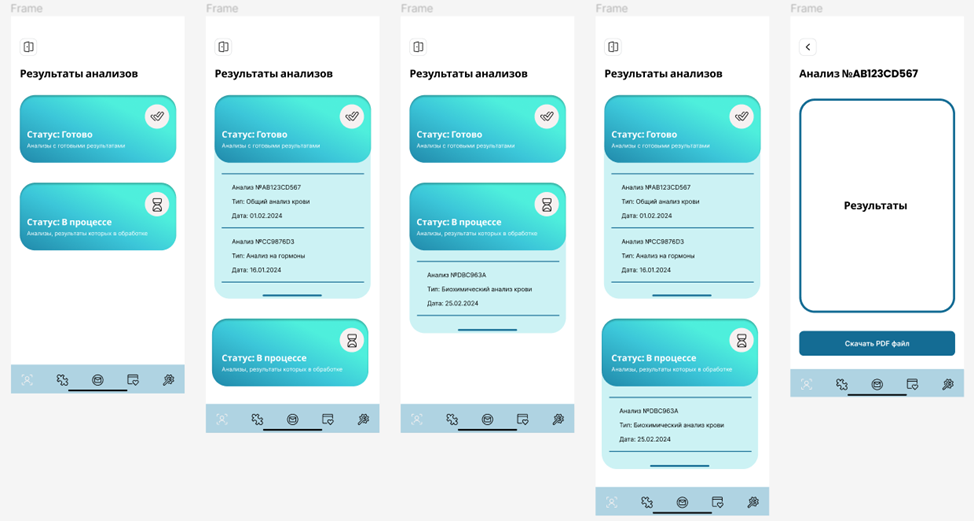


Рис. 8. Пример размещения элементов раздела «Результаты анализов».

#### **Графическая оболочка раздела «Консультант»**

В данном разделе можно задать вопрос об использовании приложения.

Следует учитывать следующие требования для его внедрения:

- должен иметь полное понимание работы системы, её функций и особенностей, чтобы точно и полно отвечать на вопросы пользователей;  
 - должен иметь доступ к актуальной информации о системе, чтобы быстро и эффективно отвечать на вопросы пользователей;  
 - должен обладать коммуникативными навыками, умением четко и понятно объяснять информацию, быть внимательным к вопросам и просьбам;

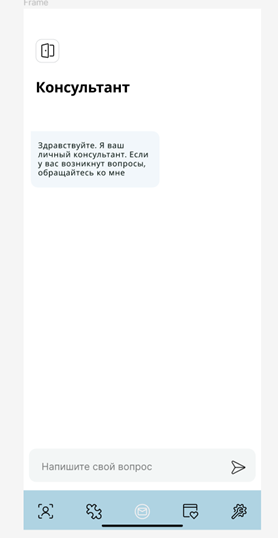


Рис. 9. Пример размещения элементов раздела «Консультант».

#### **Графическая оболочка страницы «Дети»**

В данном разделе должны находиться результаты детей, которые пользователи указали при регистрации.

Возможности:

- выбор - результаты какого ребёнка просмотреть;  
 - два списка: с готовыми результатами и в процессе;  
 - добавление нового ребёнка;  
 - заполнение формы про ребёнка;

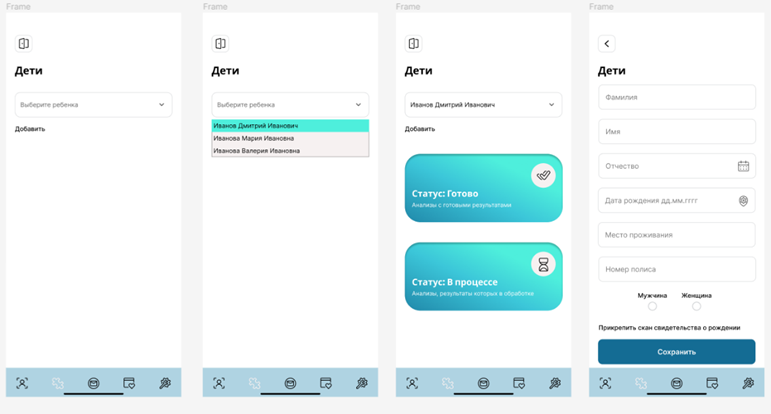


Рис. 10. Пример размещения элементов раздела «Дети».

#### **Графическая оболочка раздела «Запись»**

В данном разделе должна находиться форма для записи в любую лабораторию города. Необходимо выбрать вид анализа, в зависимости от вида будут предлагаться нужные анализы, которые относятся к виду.

Должен быть список лабораторий города для удобного выбора, и день забора анализа. При выборе дня будет отображаться календарь, для облегчения записи, а при выборе лаборатории будет писаться цена за услугу.

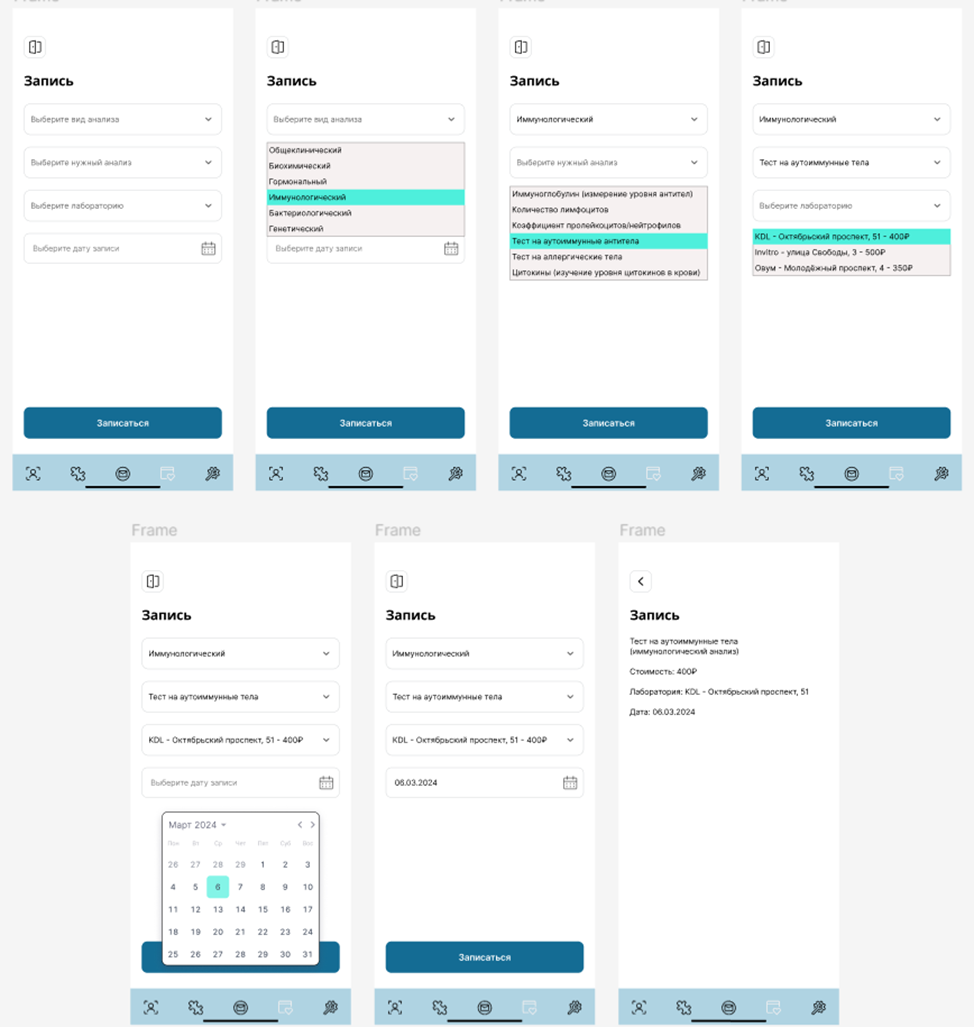


Рис. 11. Пример размещения элементов раздела «Запись».

## **Требования к разделению доступа**

Все возможные разделы приложения должны открываться для доступа только после аутентификации пользователя.

Отчёт по модели системы “MedLab” (Единая система лабораторных анализов города):

1. Уровни модели:

* Уровень 1: Пользовательский интерфейс (для удобства использования);
* Уровень 2: Логика приложения (функциональные блоки и потоки данных);
* Уровень 3: Физический уровень (конкретные компоненты и технологии)
* Уровень 4: База данных;

1. Функциональные блоки:

* Модуль получения результатов через API;
* Регистрация пользователей;
* Аутентификация пользователей;
* Просмотр своих результатов анализов;
* Просмотр результатов анализов детей;
* Запись на медицинские услуги;
* Управление профилем пользователя;
* Работа с консультантом;

1. Потоки данных:

* Записи на анализы и передача информации в лаборатории;
* Получение результатов анализов и их отображение пользователю;

1. Хранилища:

* База данных пользователей (Хранит информацию о пациентах с персональными данными);
* База данных лабораторий (Хранит информацию о пациентах, о проводимых анализах);
* Хранилище результатов анализов (Хранит информацию о результатах анализов);

1. Внешние объекты:

* Пользователи;
* Лаборатории города;
* Системы передачи данных между лабораториями и платформой (Предоставляют результаты анализов через API);

Архитектура приложения:

Клиентский уровень

* Интерфейс пользователя: Предоставляет интуитивно понятный интерфейс для пользователей, позволяя им легко просматривать результаты анализов и получать информацию о лабораториях.
* Логика приложения: Обрабатывает взаимодействие с сервером, хранит данные локально и обеспечивает функциональность приложения.

Серверный уровень

* API-интерфейс лаборатории: Предоставляет унифицированный интерфейс для взаимодействия с различными лабораториями, позволяя приложению получать результаты анализов.
* База данных пациентов: Хранит информацию о пациентах, включая их медицинские записи и результаты анализов.
* Серверная логика: Обрабатывает запросы от клиентского уровня, взаимодействует с API-интерфейсом лаборатории и обновляет базу данных пациентов.

Пользовательские роли

* Взрослые пользователи: Могут просматривать свои результаты анализов, получать информацию о лабораториях и управлять своими медицинскими записями.
* Пользователи-дети: Родители или опекуны могут просматривать результаты анализов своих детей и управлять их медицинскими записями.

Безопасность

* Аутентификация и авторизация: Приложение реализует надежные механизмы аутентификации и авторизации для защиты данных пациентов.
* Шифрование: Все данные, передаваемые между клиентским и серверным уровнями, шифруются для обеспечения конфиденциальности.
* Соответствие нормативным требованиям: Приложение соответствует всем применимым нормативным требованиям, касающимся защиты данных пациентов.