МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждения образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет Информационных технологий

Кафедра Программной инженерии

Специальность 6-05-0612-01 Программная инженерия

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**КУРСОВОГО ПРОЕКТ:**

по дисциплине «Объектно-ориентированные технологии программирования и стандарты проектирования»

Тема «Платформа по поиску репетиторов»

Исполнитель

студент (ка) 2 курса группы 9 Дубина Артём Александрович

(Ф.И.О.)

Руководитель работы асс. Ромыш А.С. (учен. степень, звание, должность, подпись, Ф.И.О.)

Курсовой проект защищен с оценкой

Председатель Смелов В.В.

(подпись)

Минск 2025

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования   
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий   
Кафедра программной инженерии

Утверждаю

Заведующий кафедрой ПИ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В.Смелов

подпись инициалы и фамилия

“\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2025г.

**ЗАДАНИЕ**

**к курсовому проектированию**

**по дисциплине** "Объектно-ориентированные технологии программирования и стандарты проектирования"

|  |  |
| --- | --- |
| Специальность: 6-05-0612-01 Программная инженерия  Студент: Дубина Артём Александрович | Группа:\_\_9\_\_ |
| **Тема: Программное средство «Поиск репетиторов»** | |

**1. Срок сдачи студентом законченной работы**: "21 мая 2025 г."

**2. Исходные данные к проекту:**

**2.1**. Функционально ПС поддерживает:

* Функции администратора:
* Регистрация и авторизация.
* Управление пользователями (блокировка, изменение данных).
* Модерация рейтингов и отзывов.
* Функции клиента:
  + Регистрация и авторизация.
  + Просмотр личного кабинета.
  + Поиск репетиторов по предметам, рейтингу, цене.
  + Просмотр профиля репетитора.
  + Оставление отзывов и рейтингов.
  + Запись на занятие при помощи заполнения формы.
* Функции клиента-репетитора:
  + Регистрация и авторизация.
  + Личный кабинет с расписанием занятий.
  + Добавление информации о себе (опыт, предметы, стоимость).
  + Просмотр заявок учеников и отклик на них.

**2.2.** При выполнении курсового проекта необходимо использовать принципы проектирования ООП. Приложение разрабатывается под ОС Windows и представляет собой настольное приложение (desktop). Отображение, бизнес логика должны быть максимально независимы друг от друга для возможности расширения. Диаграммы вариантов использования, классов реализации задачи, взаимодействия разработать на основе UML. Язык разработки проекта – C#. Управление программой должно быть интуитивно понятным и удобным. При разработке использовать несколько наиболее подходящих шаблонов проектирования ПО.

**3. Содержание расчетно-пояснительной записки**

* Введение
* Постановка задачи и обзор литературы
* Проектирование архитектуры проекта
* Разработка функциональной модели и модели данных ПС
* Тестирование
* Заключение
* Список используемых источников
* Приложения

**4. Форма представления выполненной курсовой работы:**

* + Теоретическая часть курсового проекта должны быть представлены в формате docx. Оформление записки должно быть согласно выданным правилам.
  + Листинги программы представляются в приложении.
  + Пояснительную записку, листинги, проект (инсталляцию проекта) необходимо загрузить диск, указанный преподавателем.

***Календарный план***

|  |  |  |  |
| --- | --- | --- | --- |
| №  п/п | Наименование этапов курсового проекта | Срок выполнения этапов проекта | Примечание |
| 1 | Введение | 12.03.2025 |  |
| 2 | Аналитический обзор литературы по теме проекта. Изучение требований, определение вариантов использования | 19.03.2025 |  |
| 3 | Анализ и проектирование архитектуры приложения (построение диаграмм, проектирование бизнес-слоя, представления и данных) | 31.03.2025 |  |
| 4 | Проектирование структуры базы данных. Разработка дизайна пользовательского интерфейса | 09.04.2025 |  |
| 5 | Кодирование программного средства | 23.04.2025 |  |
| 6 | Тестирования и отладка программного средства | 07.05.2025 |  |
| 7 | Оформление пояснительной записки | 14.05.2025 |  |
| 9 | Защита проекта | 21.05.2025 |  |

**5. Дата выдачи задания \_\_\_\_**05.03.2025**\_\_\_\_**

Руководитель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ *асс. Ромыш А.С.*

(подпись)

Задание принял к исполнению \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ *Дубина А.А.*

(дата и подпись студента)

Содержание

[Введение 5](#_Toc194080174)

[1. Аналитический обзор прототипов и литературных источников 6](#_Toc194080175)

[1.1 Анализ прототипов 6](#_Toc194080176)

[1.1.1 Интернет-ресурс «profi-bel» 6](#_Toc194080177)

[1.1.2. Интернет-ресурс «repetit-center.by» 7](#_Toc194080178)

[1.1.4. Интернет-ресурс «protutor.ru» 9](#_Toc194080179)

# **Введение**

В современном мире цифровые технологии активно проникают во все сферы жизни, включая образование. Одним из ключевых аспектов обучения является поиск квалифицированных репетиторов, что зачастую становится сложной задачей как для учеников, так и для преподавателей. Решение этой проблемы требует автоматизации процесса подбора репетиторов с учётом индивидуальных потребностей пользователей.

Данный курсовой проект направлен на создание программного средства для поиска репетиторов, обеспечивающего удобный интерфейс, возможность фильтрации по различным критериям (предмет, стоимость, рейтинг) и обратную связь между пользователями. Приложение позволит студентам и школьникам быстрее находить подходящих преподавателей, а репетиторам — расширять клиентскую базу.

С помощью этого программного средства пользователи смогут находить квалифицированных репетиторов, выбирать удобное для них время занятий, а также фильтровать преподавателей по предмету, стоимости и рейтингу. Будет возможность онлайн-записи на занятия, чтобы пользователь мог связаться с репетитором и договориться о проведении уроков в удобном формате.

Для успешной реализации курсового проекта необходимо:

* провести анализ соответствующей литературы;
* ознакомиться с прототипами аналогов по данной теме;
* определить функциональные требования;
* продумать структуру базы данных;
* продумать структуру проекта;
* реализовать программное средство;
* протестировать программное средство;
* написать руководство пользователя.

В процессе выполнения курсового проекта будут применены принципы и подходы объектно-ориентированного программирования (ООП), а также будет использована база данных MS SQL Server и технология Windows Presentation Foundation (WPF).

# **Аналитический обзор прототипов и литературных источников**

## **Анализ прототипов**

Курсовой проект представляет из себя разработку приложения для поиска репетиторов. В этой сфере существует множество аналогичных решений, которые предоставляют пользователям возможность выбирать и записываться на занятия. В ходе выполнения курсового проекта были проанализированы цели и задачи, а также рассмотрены аналогичные примеры их решений. На основании анализа всех достоинств и недостатков данных альтернативных решений были сформулированы требования к данному программному средству.

## **Интернет-ресурс «profi-bel»**

Сайт profi-bel представляет поиск специалистов по разным областям в Минске. Интерфейс у сайта удобный и интуитивно понятный. Как минус можно выделить сильную нагрузку на пользователя из-за огромного спектра услуг. Интерфейс сайта представлен на рисунке 1.1.

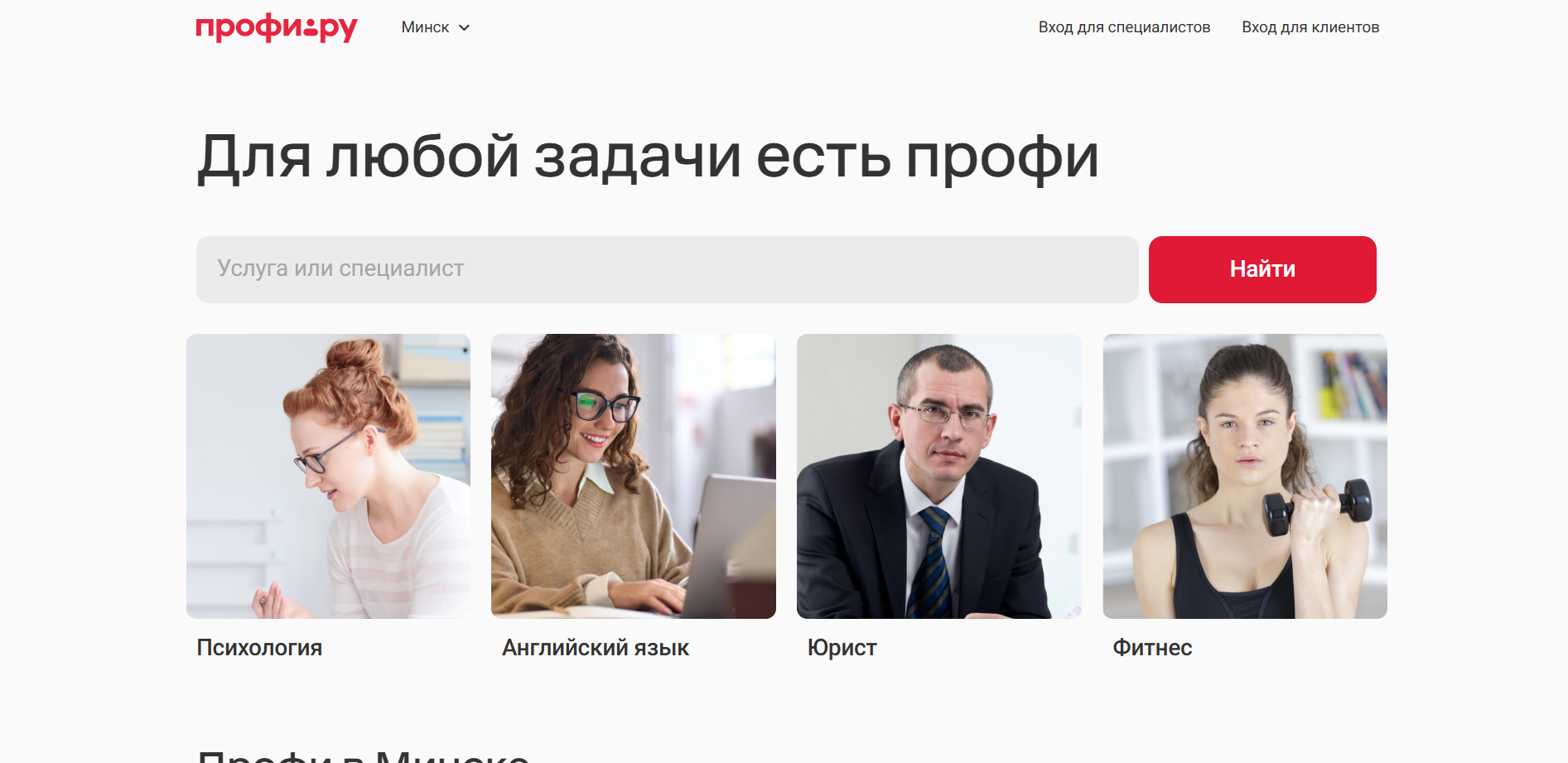


Рисунок 1.1 – Интерфейс сайта profi-bel.by

На этом сайте пользователь имеет возможность найти специалистов в различных областях, включая репетиторов, мастеров по ремонту, тренеров и других профессионалов. Пользователи могут ознакомиться с профилями специалистов, содержащими информацию об их опыте, услугах и контактных данных. Сайт также предлагает систему отзывов, позволяющую оценить качество работы специалистов на основе мнений других клиентов. ​

Главной целью сайта является соединение клиентов с профессионалами, предоставляя удобный инструмент для поиска и выбора подходящего специалиста. Пользователь может оставить заявку, указав свои требования и предпочтения, после чего специалисты откликаются на нее, предлагая свои услуги. Однако процесс записи на услуги осуществляется через прямой контакт со специалистом, и на сайте отсутствует возможность онлайн-бронирования или оплаты услуг, что может быть неудобством для некоторых пользователей. ​

Положительной стороной сайта является наличие системы отзывов, позволяющей клиентам делиться своим опытом взаимодействия со специалистами. Однако пользователи не могут оставлять отзывы или задавать вопросы непосредственно на сайте без предварительного взаимодействия со специалистом через платформу. ​

Сайт предоставляет информацию о местоположении специалистов и их контактные данные, что облегчает процесс связи.

В целом, сайт выполняет свою основную функцию по предоставлению информации о специалистах и их услугах. Однако отсутствие возможности онлайн-бронирования и ограниченные функции взаимодействия на сайте могут быть восприняты как недостатки для пользователей, ожидающих более полного онлайн-сервиса.

## **1.1.2. Интернет-ресурс «repetit-center.by»**

​ Сайт repetit-center.by предназначен для подбора репетиторов по различным предметам и направлениям. Пользователи могут изучить список преподавателей, ознакомиться с их опытом, квалификацией и специализацией. Каждый репетитор имеет личную анкету с указанием цен на занятия и контактной информации, что упрощает процесс выбора подходящего специалиста. Интерфейс сайта представлена на рисунке 1.2.

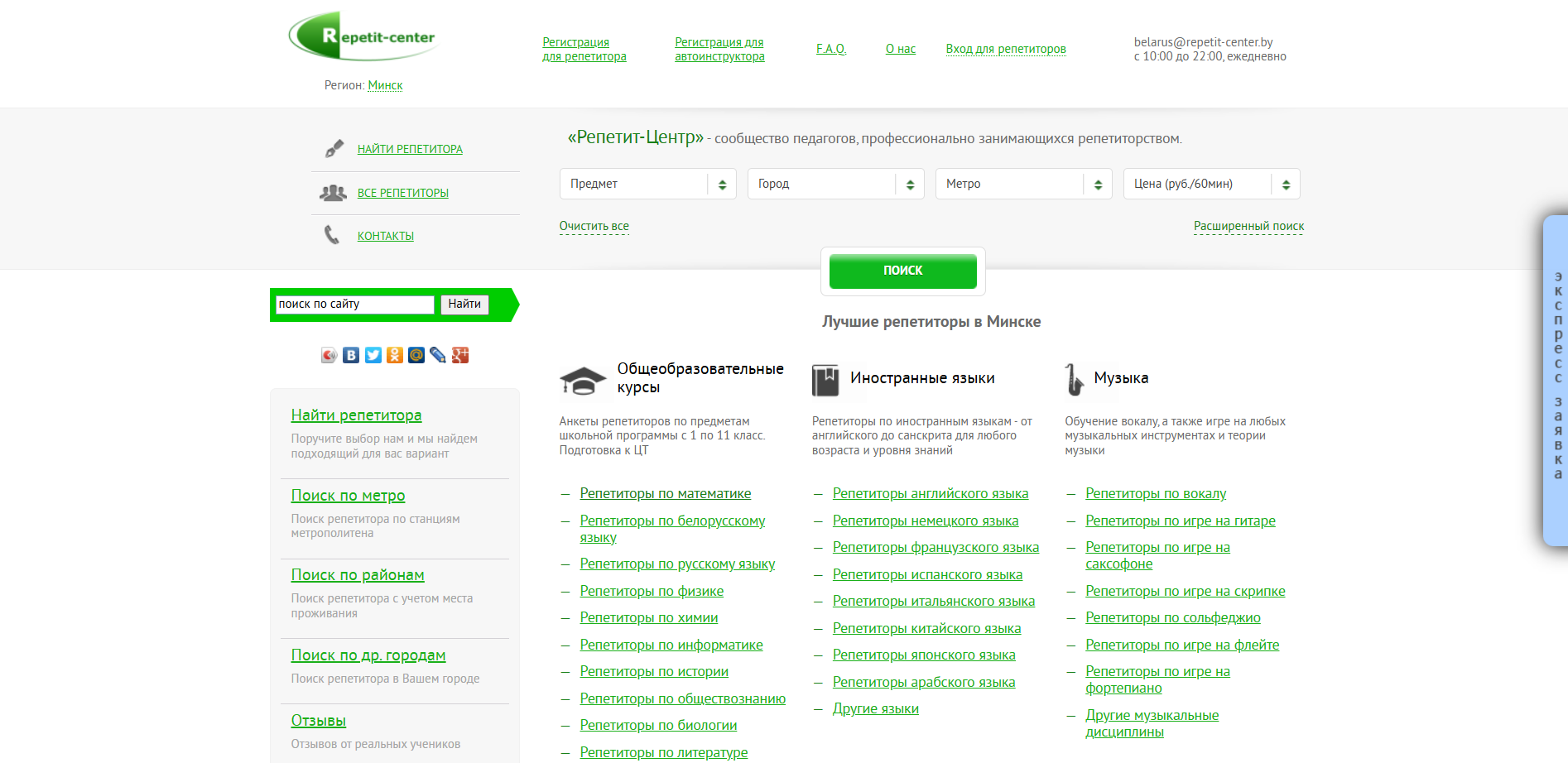


Рисунок 1.2 – Интерфейс сайта profi-bel

Основная функция сайта – помочь ученикам найти подходящего репетитора, связаться с ним и договориться о занятиях. Однако бронирование уроков и оплата происходят вне платформы, что может быть не всегда удобно. Пользователь оставляет заявку с указанием своих требований, после чего преподаватели могут связаться с ним.

Одним из преимуществ сайта является наличие отзывов о репетиторах, что помогает оценить их компетентность перед началом занятий. Однако функционал платформы не позволяет оставлять комментарии напрямую без предварительного взаимодействия с преподавателем.

Также на сайте доступна информация о местоположении преподавателей, их контактные данные, ссылки на социальные сети и способы связи. Это облегчает коммуникацию, но отсутствие онлайн-записи и личного кабинета для учеников является минусом, поскольку пользователям приходится управлять расписанием самостоятельно.

В целом, repetit-center.by – полезный ресурс для поиска репетиторов, но платформа могла бы быть более удобной, если бы включала дополнительные функции, такие как система бронирования и онлайн-оплата.

**1.1.3. Интернет-ресурс «repetit.ru»**

Основная функция сайта repetit.ru – помочь ученикам найти подходящего репетитора, связаться с ним и договориться о занятиях. Платформа предоставляет удобный поиск репетиторов по различным предметам и специализациям, что позволяет быстро выбрать подходящего кандидата.

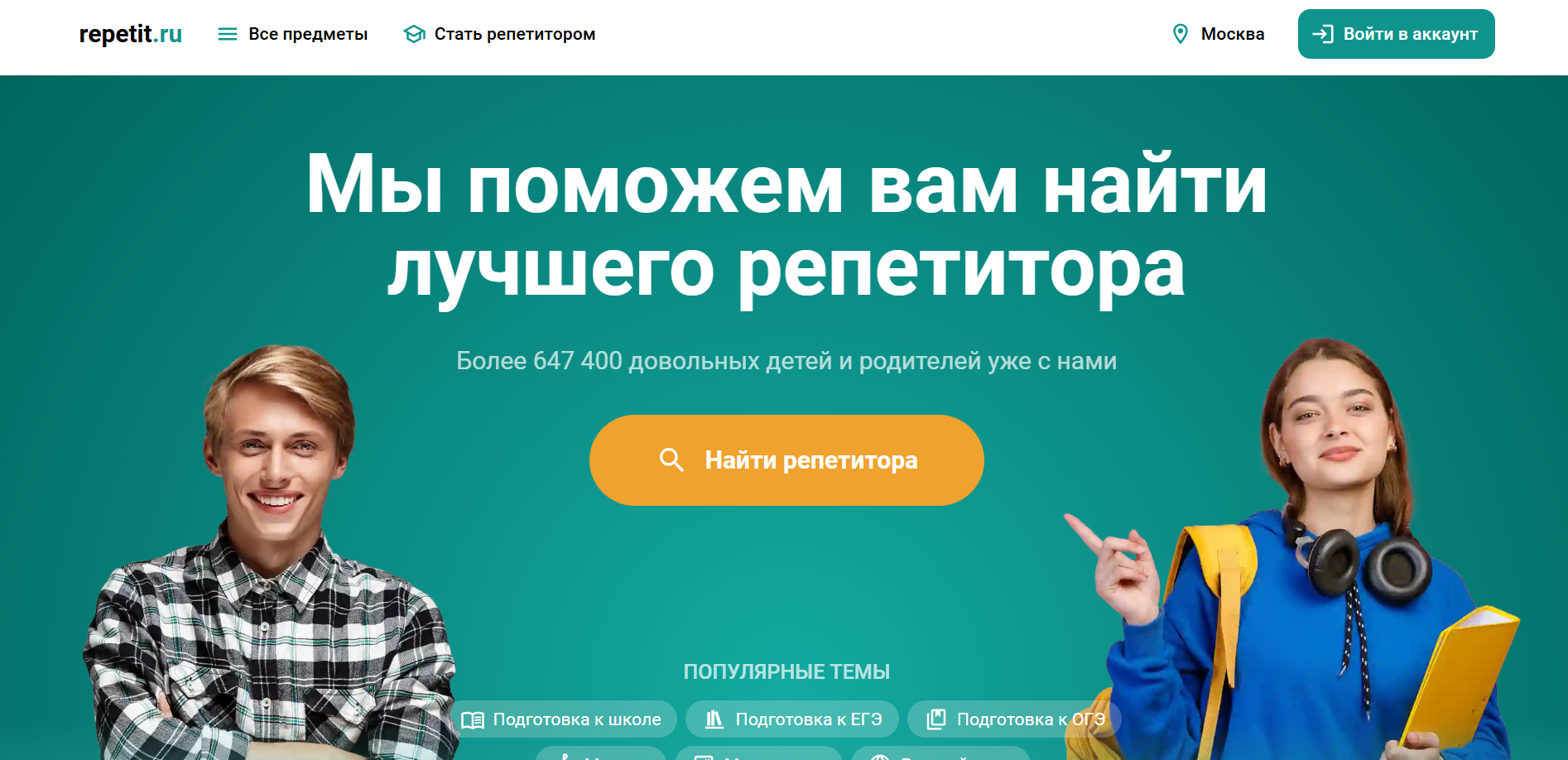


Рисунок 1.3 – Интерфейс сайта repetit.ru

Как и в случае с repetit-center.by, бронирование уроков и оплата происходят вне сайта, что может быть не всегда удобно для пользователей, так как процесс взаимодействия с преподавателем не интегрирован в саму платформу. Одним из ключевых преимуществ сайта является наличие отзывов о репетиторах, что помогает сформировать мнение о компетентности преподавателей. Отзывы могут быть полезны для учеников при принятии решения о выборе репетитора. Однако, как и на предыдущем сайте, функционал платформы не позволяет оставлять комментарии без предварительного взаимодействия с преподавателем, что ограничивает возможности для более детальной обратной связи.

Сайт предоставляет информацию о местоположении репетиторов, их контактные данные, а также ссылки на социальные сети, что помогает установить контакт. Однако отсутствие возможности онлайн-бронирования и личного кабинета для учеников ограничивает функциональность платформы, что усложняет управление расписанием и взаимодействие с преподавателем.

В целом, repetit.ru – полезный ресурс для поиска репетиторов, но платформа могла бы быть более удобной и эффективной, если бы добавила систему онлайн-бронирования и оплати, а также личный кабинет для учеников для удобного управления процессом обучения.

## **1.1.4. Интернет-ресурс «protutor.ru»**

Сайт protutor.ru представляет собой яркий и многофункциональный сайт по поиску репетиторов онлайн. Интерфейс данного сайта представлен на рисунке 1.4.

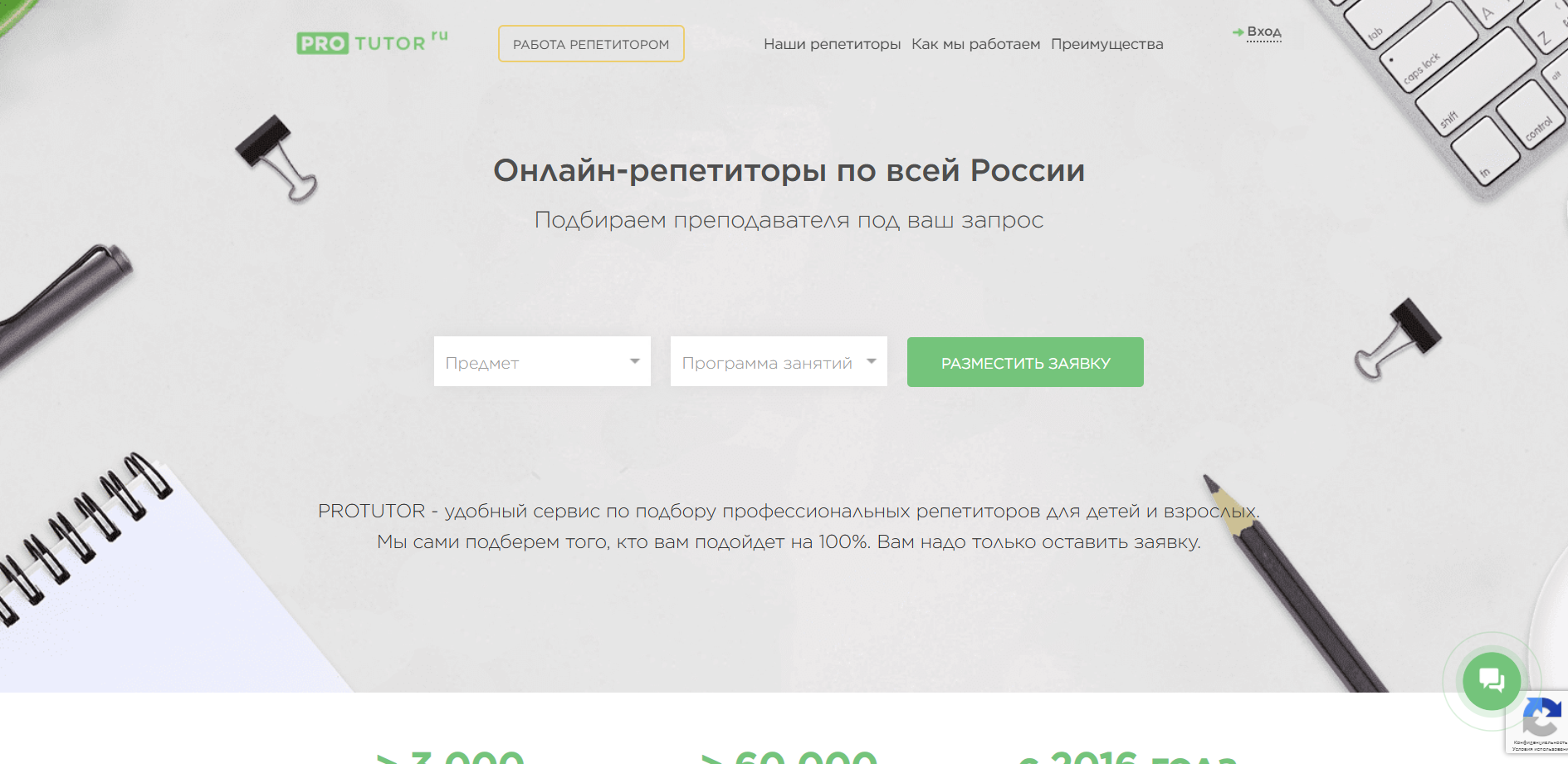


Рисунок 1.4 – Интерфейс сайта protutor.ru

Сайт protutor.ru ориентирован на помощь ученикам в поиске репетиторов и установлении контакта с ними. Платформа предлагает широкий выбор преподавателей по множеству предметов и специализаций, что является значительным плюсом для пользователей. Также сайт предлагает возможность фильтрации репетиторов по различным критериям, таким как стоимость, опыт работы и место проживания, что упрощает поиск подходящего кандидата.

Одним из главных преимуществ является наличие системы рейтингов и отзывов, которая помогает пользователям оценить качество работы репетитора. Однако стоит отметить, что отзывы можно оставить только после первых занятий, что ограничивает возможность получения информации о преподавателе до начала курса.

На сайте также предусмотрена возможность связи с преподавателями через личные сообщения, но отсутствие системы онлайн-бронирования и оплаты может создать неудобства для пользователей, так как все эти процессы приходится организовывать вне платформы.

Кроме того, сайт предоставляет полезную информацию о преподавателях, включая их профессиональное образование, сертификаты и опыт, что способствует более осознанному выбору репетитора. Однако, несмотря на наличие этих данных, отсутствие функционала для создания личных кабинетов для учеников и преподавателей затрудняет управление процессом обучения.

В общем, protutor.ru является хорошим ресурсом для поиска репетиторов с разнообразными специальностями и гибкими возможностями фильтрации, но добавление функций онлайн-бронирования и личных кабинетов сделало бы платформу более удобной и функциональной для пользователей.

1. **Анализ требований к программному средству и разработка функциональных требований**

Для того чтобы приступить к этапу разработки проекта, необходимо четко сформулировать функциональные требования к программному средству.

* 1. **Определение требований к программному средству**

Функционально ПС должно выполнять следующие задачи:

* Функции администратора:
* Регистрация и авторизация.
* Управление пользователями (блокировка, изменение данных).
* Модерация рейтингов и отзывов.
* Функции клиента:
  + Регистрация и авторизация.
  + Просмотр личного кабинета.
  + Поиск репетиторов по предметам, рейтингу, цене.
  + Просмотр профиля репетитора.
  + Оставление отзывов и рейтингов.
  + Запись на занятие при помощи заполнения формы.
* Функции клиента-репетитора:
  + Регистрация и авторизация.
  + Личный кабинет с расписанием занятий.
  + Добавление информации о себе (опыт, предметы, стоимость).
  + Просмотр заявок учеников и отклик на них.
  1. **Описание средств разработки**

При разработке приложения были использованы:

* интегрированная среда разработки Microsoft Visual Studio 2022;
* программная платформа .NET Framework;
* язык программирования C#;
* расширяемый язык разметки XAML;
* технология WPF;
* технология Entity Framework;
* Microsoft SQL Server.

В качестве интерфейса прикладного программирования был выбран обширный API-интерфейс – Windows Presentation Foundation (WPF), предназначенный для создания настольных программ с графически насыщенным пользовательским интерфейсом. В основе графической технологии WPF лежит мощная инфраструктура, основанная на DirectX. Это является одним из основных отличий WPF от более ранней технологии создания пользовательских интерфейсов – Windows Forms.

Для работы с WPF использовался объектно-ориентированный язык программирования с С-подобным синтаксисом – С#, разработанный для создания приложений на платформе Microsoft .NET Framework.

Есть различные механизмы работы с БД в С#. Т.к. моя база данных небольшая, мой выбор пал на Entity Framework. Entity Framework предоставляет простой способ доступа к данным в .Net Framework.

* 1. **Описание функциональности программного средства**

Описание функциональности программного средства представлено с помощью UML-диаграммы вариантов использования (еще не представлено, но будет)

Спецификация функциональности программного средства представлена в следующем пункте

* 1. **Спецификация функциональных требований**

Для функциональности ПС необходимо создание базы данных для хранения информации приложения. Подробно о базе данных описано в главе 3.

В программном средстве «Приложение по изучению английского языка» при запуске необходимо реализовать регистрацию и авторизацию пользователей для дальнейшего использования приложения. Для авторизации входными параметрами являются электронная почта и пароль пользователя, которые содержатся в базе данных. Для того чтобы зарегистрироваться необходимо ввести имя, электронную почту, пароль. Введенные данные, успешно прошедшие валидацию, заносятся в базу данных, а пользователь получает доступ к приложению.

При авторизации администратора он может на странице «*Добавление курсов*» добавлять, изменять и удалять курсы и слова в них при нажатии на соответствующие названиям действий кнопки, также добавлять список слов. На странице «*Добавление грамматики»* добавлять, изменять и удалять уроки и задания в них при нажатии на соответствующие названиям действий кнопки.

При авторизации пользователя он может на странице «*Статистика*» просматривать личную статистку прогресса за последние 2 недели, общую статистику всех пользователей за разные периоды времени. На странице «*Личный кабинет*» изменять настройки аккаунта и обучения. На странице «Тренировка слов» расширять лексикон посредством различных упражнений. На странице «Просмотр слов» может смотреть словари и слова в них, так же сбросить прогресс словаря (словарей). На странице «*Добавление словарей*» добавлять, изменять и удалять собственные словари и слова в них. На странице «*Тренировка грамматики*» расширять знания грамматики посредством выполнения заданий.

Кнопка «*сменить аккаунт*» возвращает пользователя к окну авторизации. Кнопка «*выход*» закрывает приложение. Кнопка «*свернуть*» сворачивает приложение. Кнопка «*руководство*» открывает окно с текстом пользовательского руководства.

Все данные о словарях, слова, уроках, заданиях, пользователях, пользовательских настройках, результатах обучения должны заноситься в базу данных.

1. **Проектирование программного средства** 
   1. **Архитектура проектируемого программного средства**

При создании программного средства использовался архитектурный паттерн Model-View-ViewModel (MVVM). Он был представлен архитектором программного обеспечения компании Microsoft Джоном Госсманом в 2005 году. Одной из главных целей шаблона является отделение и абстрагирование видимого пользовательского интерфейса от логики представления. MVVM состоит из трех компонентов: модель (Model), представление (View) и модель представления (ViewModel). Наглядное представление паттерна представлено на рисунке 3.1.

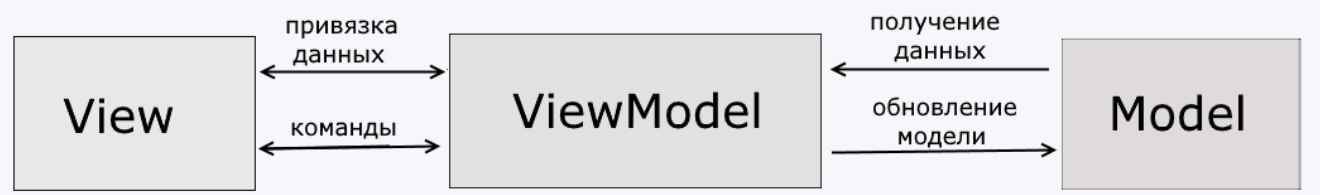


Рисунок 3.1 − Наглядное представление MVVM

Модель (Model) — это субъект, который описывает используемые в приложении данные. Это может быть что угодно, будь то обычный продукт или же сложный рецепт. Представление (View) — это графический элемент управления или набор элементов управления, отвечающих за отображение данных на экране. Модель представления (ViewModel) — это и есть та магия, стоящая за всем. Она содержит логику пользовательского интерфейса, команды, события и ссылку на модель. В MVVM модель преставления не обязана обновлять непосредственно данные представления. Благодаря мощному механизму привязки, представленному в WPF, пользовательский интерфейс обновляется, как только обновляется модель преставления. Для этого модель преставления должна реализовывать интерфейс INotifyPropertyChanged и запустить событие PropertyChanged.

3. 1. 1. **Схема и описание авторизации**

Схема авторизации представлена в приложении Б. Сразу после запуска программу пользователь попадает в окно авторизации. Если поля электронной почты и пароля будут пустыми, то границы textbox поменяют цвет на красный. Если поля заполнены, то программа обращается к базе данных и проверяет существует ли пользователь с соответствующим логином и паролем. Если пользователь не найден, будет выведено сообщение об ошибке, в противном случае вход в приложение

* + 1. **Реализация MVVM**

Для работы с паттерном использовалась библиотека DevExpressMVVM. Для реализации паттерна файлы программы были распределены по соответствующим пространствам имен и реализованы соответствующие функции. В листинге 3.1 видим, что класс MyViewModelBaseWindow наследует класс ViewModelBase, который лежит в пространстве имен DevExpress.Mvvm.

internal class MyViewModelBaseWindow : ViewModelBase, IMyViewModelBaseWindow

{

public double Top { get; set; }

public double Left { get; set; }

public ICommand closeWindow => new DelegateCommand(CloseWindow);

public void CloseWindow()

{

foreach (Window window in Application.Current.Windows)

{

if (window.DataContext == this)

{

window.Close();

}

}

\_unitOfWork.Dispose();

}

}

Листинг 3.1 – Пример использования DevExpressMVVM

* + 1. **Подключение к БД**

Есть различные механизмы работы с БД в С#. Так как моя база данных небольшая, мой выбор пал на Entity Framework. Entity Framework предоставляет простой способ доступа к данным в .Net Framework. Подход работы с базой данных является Code First. Подключение осуществляется в файле App.config строкой, которая указана в листинге 3.2.

<connectionStrings>

<add name="DbConnectionString" connectionString="data source=DESKTOP-FNUCRE9\SQLEXPRESS;Initial Catalog=English selfteacher;Integrated Security=true;" providerName="System.Data.SqlClient" />

</connectionStrings>

Листинг 3.2 – Подключение базы данных к проекту

* + 1. **Построение графиков в WPF**

NuGet предоставляет достаточное количество библиотек для построения графиков. Мой выбор упал на библиотеку LiveCharts. (листинг 3.3).

private SeriesCollection DrawGraph(int[] count)

{

SeriesCollection seriesCollection = new SeriesCollection

{

new LineSeries

{

Title = "",

Values = new ChartValues<int>(count),

PointGeometry = DefaultGeometries.Circle,

PointGeometrySize = 10,

PointForeground = (Brush)brushConverter.ConvertFrom("#FFA6E880")

}

};

GraphLabels = new string[RepositoryStatistics.PERIOD];

for (int i = 0; i < RepositoryStatistics.PERIOD; i++)

{

GraphLabels[i] = DateTime.Today.AddDays(-(RepositoryStatistics.PERIOD - i - 1)).Date.ToShortDateString();

}

GraphFormatter = value => value.ToString("A");

((LineSeries)seriesCollection[0]).Stroke = (Brush)brushConverter.ConvertFrom("#FF78BA52");

((LineSeries)seriesCollection[0]).Fill = (Brush)brushConverter.ConvertFrom("#2F78BA52");

return seriesCollection;

}

Листинг 3.3 – Пример использования LiveCharts

* 1. **Разработка базы данных**

Для обеспечения работоспособности программного средства была разработана база данных, что являлось обязательным условием курсового проекта. База данных была тщательно сконструирована до начала написания самого приложения. Для ее создания использовалась реляционная система управления базами данных (РСУБД) MS SQL.

Таким образом была разработана база данных English selfteacher, которая включает в себя таблицы Achievements, Dictionaries, DictionaryWords, GrammarProgresses, Lessons, LexiconProgresses, Statistics, Tasks, UserAchievements, Users, UserSettings, Words. Схема базы данных представлена в приложении В.

Таблица Achievements используется для хранения достижений. Включает 3 столбца: Id – id достижения, Name – описание достижения, PhotoPath – путь к картинке достижения.

Таблица Dictionaries используется для хранения словарей. Включает 3 столбца: id – id словаря, Name – название словаря, CreatorId – id создателя словаря.

Таблица DictionaryWords используется для организации связи многие-ко-многим между таблицами Dictionaries и Words, так как в одном словаре может быть много слова, а одно слово может быть во многих словарях.

Таблица GrammarProgresses используется для хранения результатов прогресса грамматики. Включает 3 столбца: id – id записи, Task\_id – id задания, User\_Id -–id пользователя.

Таблица Lessons используется для хранения уроков. Включается 4 столбца: Id – id урока, Name – название урока, Theory – теоретические сведения к уроку, PassingPercent – проходной процент, на который нужно пройти урок, чтобы получить доступ к следующему уроку.

Таблица LexiconProgresses используется для хранения результатов прогресса изучения лексикона. Содержит 5 столбцов: Id – id записи, ProgressCount – количество верных ответов подряд, isLearned – 0 не выучено, 1 – выучено, User\_Id – id пользователя, WordId – id слова.

Таблица Statistics используется для хранения статистики прогресса. Включает 5 столбцов: Id – id записи, Date – дата записи, WordsCount – количество правильных ответов в тренировках слов, TasksCount - количество правильных ответов в тренировках грамматики, User\_Id – id пользователя.

Таблица Tasks используется для хранения заданий. Включает 4 столбца: Id – id задания, EnglishSentence – английский вариант предложения, RussianSentence – русский вариант предложения, Lesson\_Id – id урока, к которому принадлежит задание.

Таблица UserAchievements используется для организации связи многие-ко-многим между таблицами Users и Achievements, так как у одного пользователя может быть много достижений, а одно достижение может быть у многих пользователей.

Таблица Users используется для хранения пользователей. Включает 6 столбцов: id – id пользователя, Name – имя пользователя, Email – электронная почта пользователя, Password – пароль пользователя, при записи в базу данных хэшируется, IdEnableLesson – id доступного к изучению урока, isAdmin – 0 – пользователь, 1 - администратор.

Таблица UserSettings используется для хранения пользовательских настроек. Включает в себя 6 столбцов: Id – id настройки (связь с таблицей Users один-к-одному), CountRepeatWordForLearned – количество повторений подряд, чтобы слово считалось выученным, CountLearningWordSimultaneously – количество слов изучаемых одновременно, ModeLearningWords – режим изучения слов, ModeLearningTasks – режим изучения грамматики.

Таблица Words используется для хранения слов. Содержит 5 столбцов: Id – id слова, EnglishSpelling – английский вариант слова, Transcription – транскрипция слова, RussianSpelling – русский вариант слова, Description – описание слова.

В общей сложности проект включает в себя 7 корневых папок. Его структура представлена на рисунке 3.2.

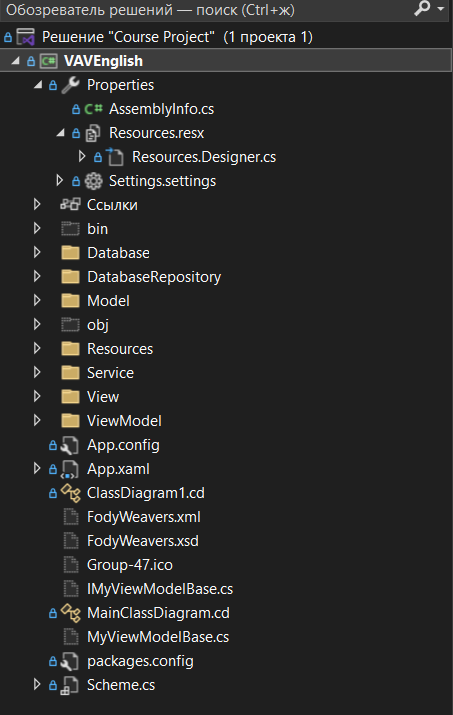


Рисунок 3.2 − Структура проекта

Папка Database хранит классы и интерфейсы, которые необходимы для подключения и работы с базой данных, так же её предварительного заполнения.

Папка DatabaseRepository хранит классы и интерфейсы, которые необходимы для непосредственной работы с базой данных, а именно получение, изменение и удаление данных.

Папка Model хранит классы, описывающие используемые в проекте сущности (объекты).

Папка Resources хранит различные картинки.

Папка Service хранит классы, предоставляющие различный вспомогательный функционал.

Папка View содержит элементы пользовательского интерфейса, а именно окна и страницы, которые пользователь может увидеть на экране.

Папка ViewModel содержит в себе интерфейсы и классы со свойствами и командами, которые связаны механизмом привязки с объектами папки View.

Взаимосвязь между классами хорошо видна в диаграмме классов (Приложение Г).

Приложение включает в себя 6 окон и 7 страниц. Связи между ними представлены на приложении Д. Также в проекте есть окно оповещения об ошибке.

1. **Описание интерфейсов программного средства, его тестирование и анализ полученных данных**
   1. **Описание интерфейсов программного средства**

Сразу после запуска приложения перед пользователем открывается окно авторизации, в которое требуется ввести электронную почту и пароль. Есть возможность запомнить данные для входа, что ликвидирует потребность вводить их каждый раз при входе. Внешний вид окна представлен на рисунке 4.1.

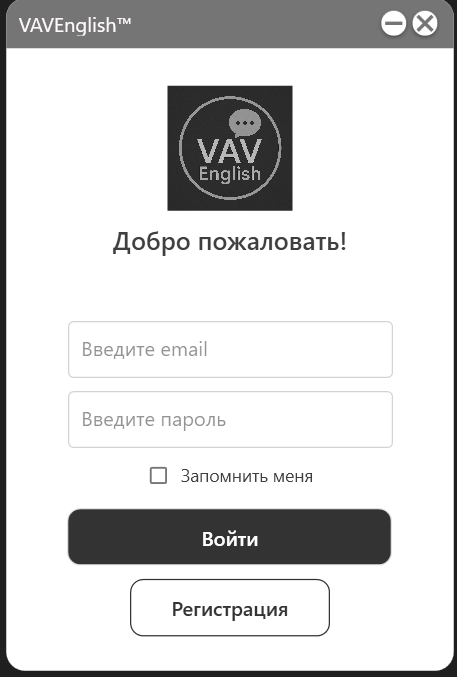


Рисунок 4.1 − Внешний вид окна авторизации

Из окна авторизации пользователи, которых нет в базе данных приложения, могут зарегистрироваться, нажав на кнопку «Регистрация». После того, как это произойдет, откроется окно регистрации новых пользователей, интерфейс которого представлен на рисунке 4.2.

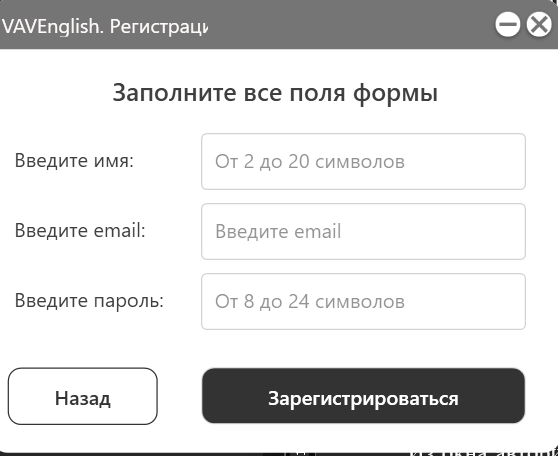


Рисунок 4.2 − Интерфейс окна регистрации

Также в окне регистрации есть функция возврата к окну авторизации, которую реализует элемент управления с надписью «Назад».

В случае корректного заполнения полей данными и нажатия на кнопку «Зарегистрироваться» открывается окно подтверждения регистрации, интерфейс которого представлен на рисунке 4.3.

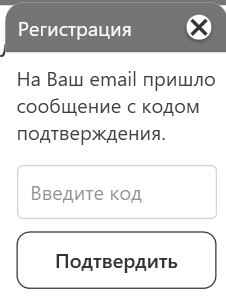


Рисунок 4.3 − Интерфейс окна подтверждения регистрации

1. 1. 1. **Вход в качестве пользователя**

После успешной авторизации открывается главное окно приложения, которое встречает пользователя. На главной странице пользователь может видеть личную статистику и топ среди всех пользователей. Внешний вид главного окна пользователя представлен на рисунке 4.4.

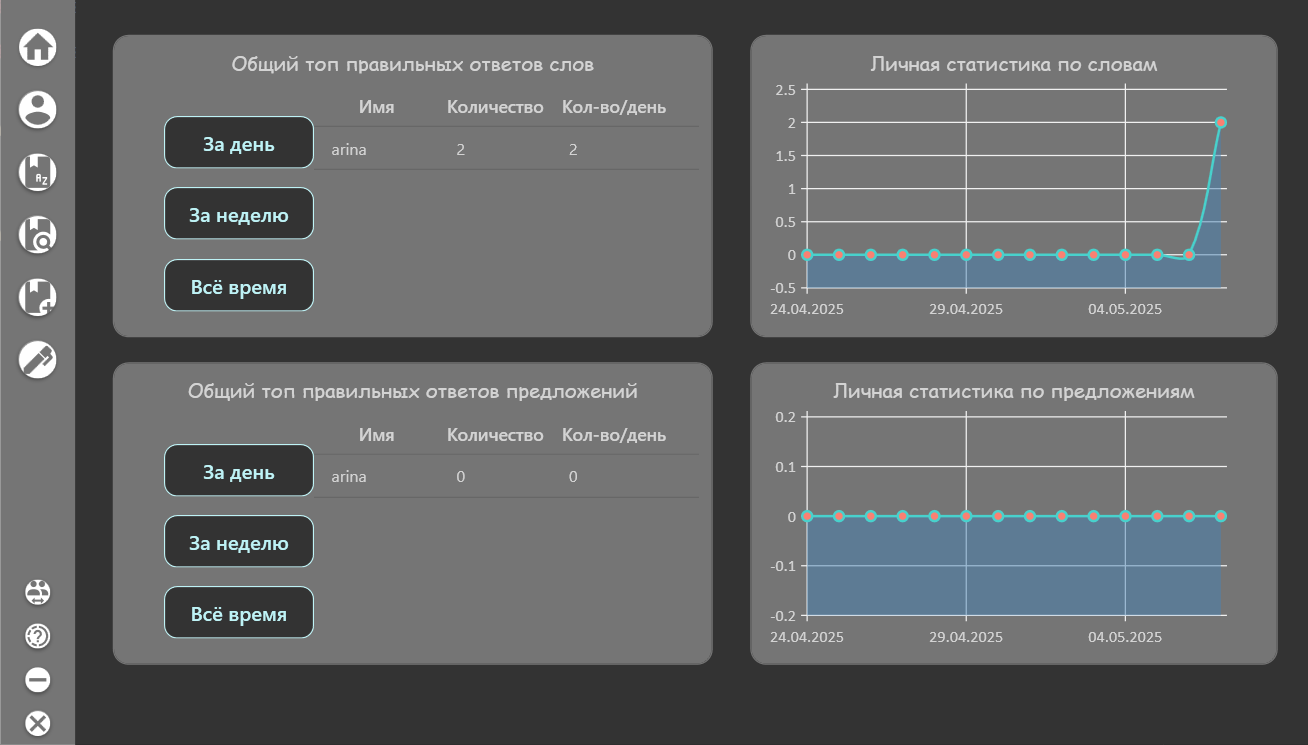


Рисунок 4.4 − Внешний вид главного окна пользователя

При нажатии на кнопки в верхней панели открываются страницы «Личный кабинет», «Тренировка слов», «Просмотр слов», «Добавление словарей», «Тренировка грамматики», так же есть кнопки, которые позволяют сменить аккаунт, закрыть приложение, свернуть приложение и открыть окно пользовательского руководства. На рисунках 4.5 – 4.9 представлен внешний вид этих страниц и окна пользовательского руководства.



Рисунок 4.5 − Страница «Личный кабинет»

В «Личном кабинете» пользователь может просмотреть личную информацию, изменить её и установить нужные настройки обучения.

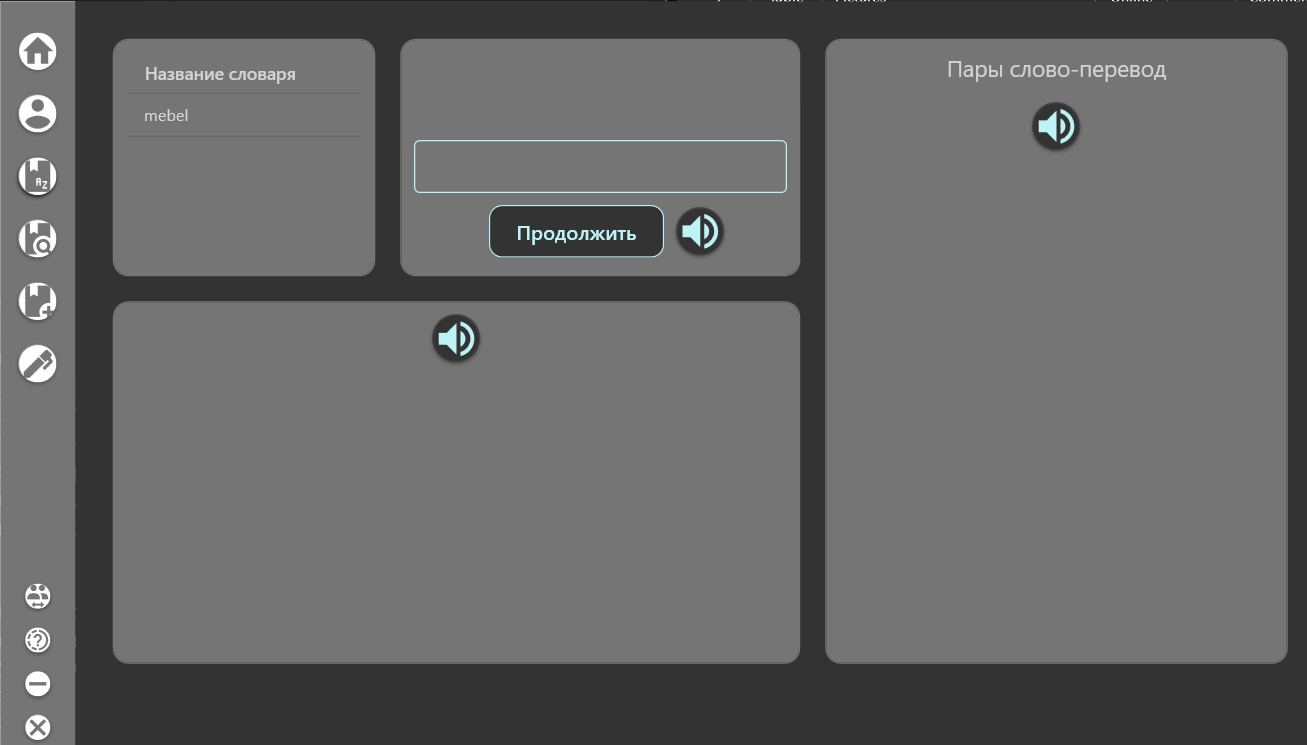


Рисунок 4.6 − Страница «Тренировка слов»

На странице «Тренировка слов» пользователь может расширять словарный запас посредством тренировки одним или несколькими способами, доступными на странице.

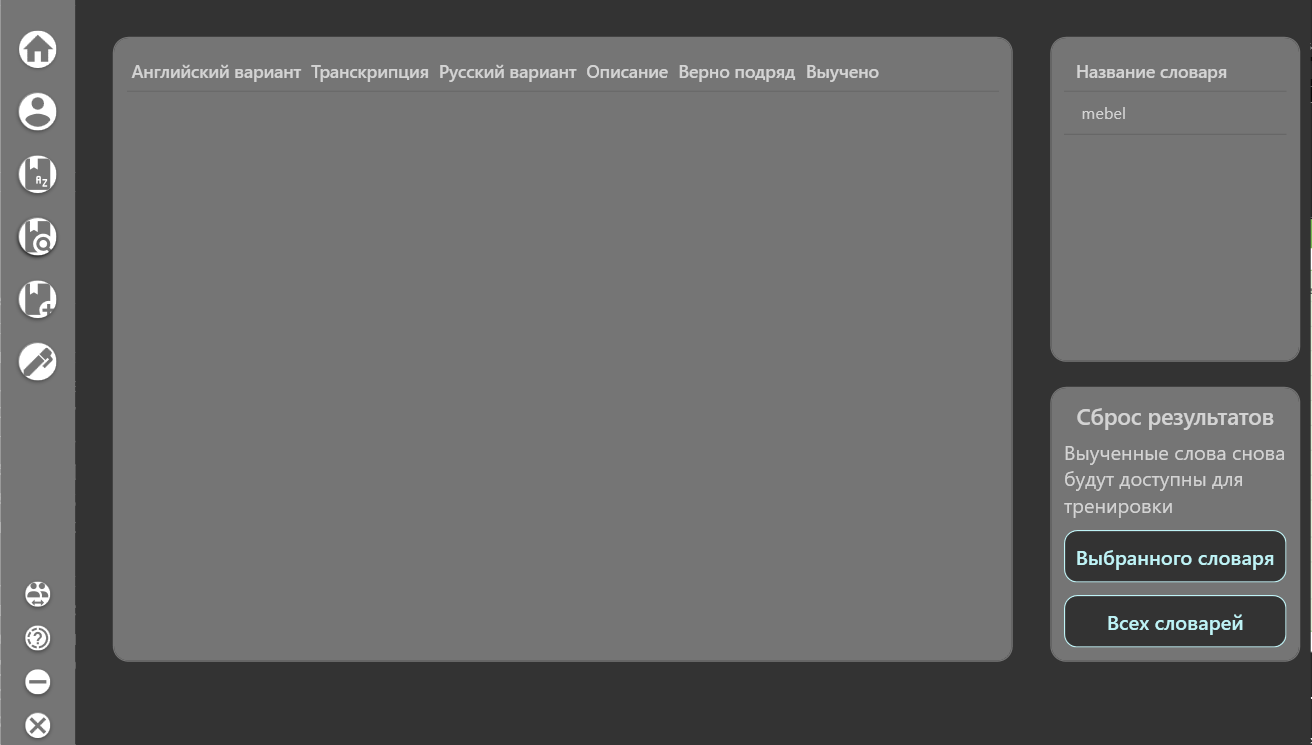


Рисунок 4.7 − Страница «Просмотр слов»

На странице «Просмотр слов» (рис. 4.8) пользователь может изучить все доступные словари, посмотреть прогресс по каждому слову, также сбросить результаты тренировок слов одного словаря или всех сразу.

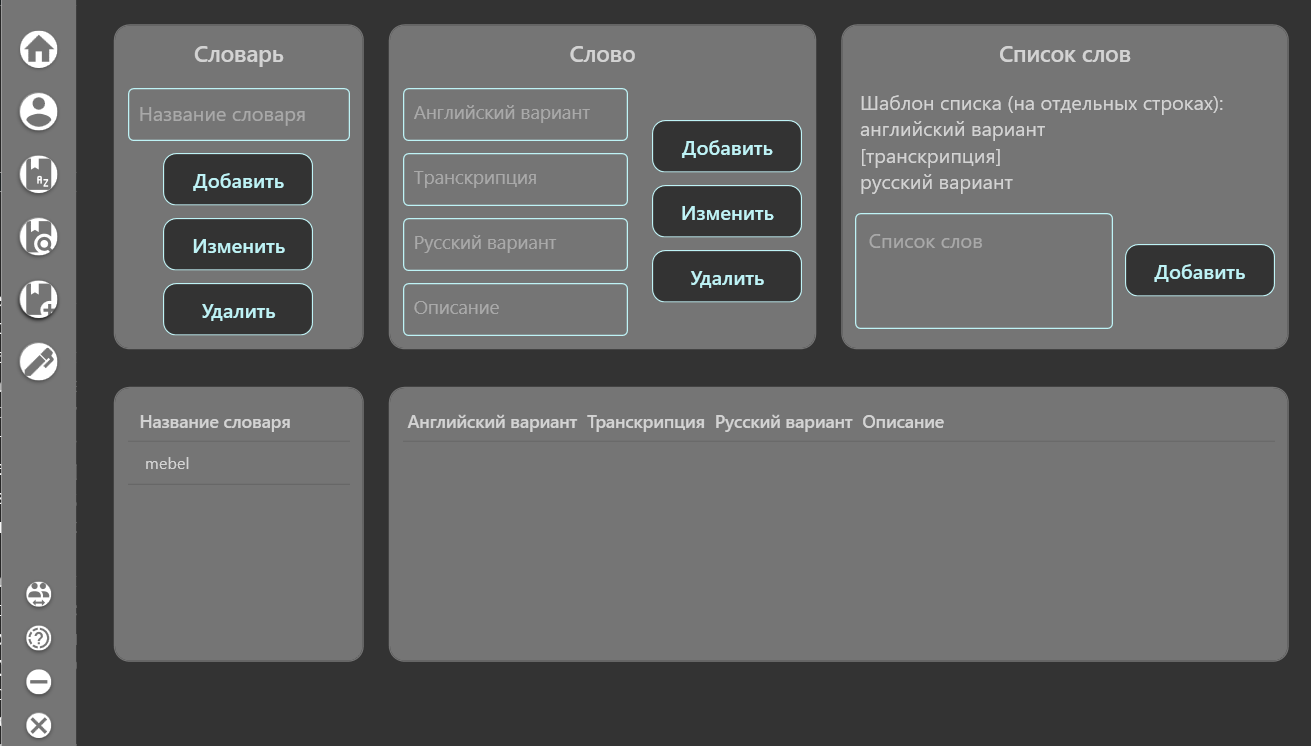


Рисунок 4.8 − Страница «Добавление словарей»

На странице «Добавление словарей» (рис. 4.9) пользователь может добавлять, изменять или удалять собственные словари и слова в них. Добавление слов в словарь возможно по одному слову или целым списком.

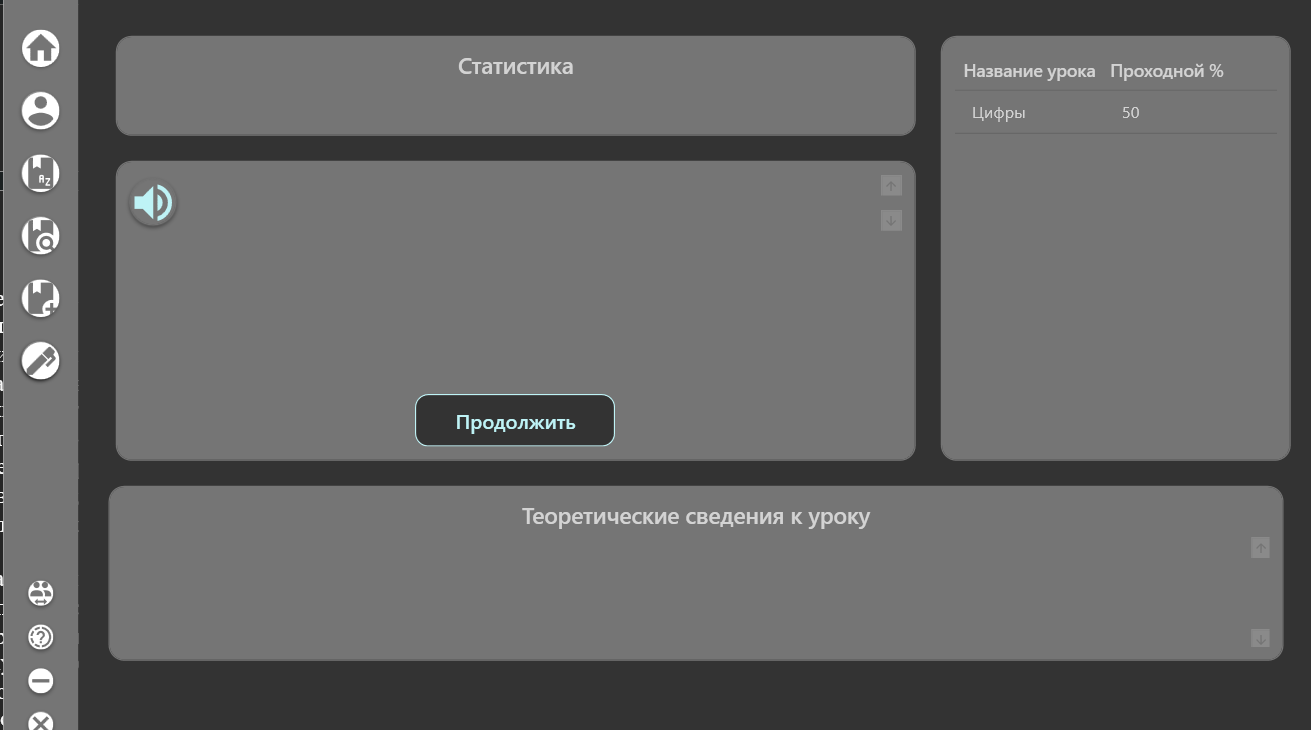


Рисунок 4.9 − Страница «Тренировка грамматики»

На странице «Тренировка грамматики» можно улучшить свои грамматические навыки посредством тренировки. В уроке несколько заданий, суть которых правильно перевести предложение путём выбора в верном порядке кнопок со словами. Последующие уроки становятся доступны после прохождения текущего урока на проходной процент.

* + 1. **Вход в качестве администратора**

После успешной авторизации открывается окно администратора с открытой страницей «Добавление словарей» (рис. 4.8). Тут администратор может добавлять, изменять и удалять слова и словари, которые были созданы администраторами. Пользовательские словари остаются видны только пользователям, а словари администраторов видны все пользователям, но только для тренировки и просмотра.

Вторая страница в окне администратора это страница «Добавление грамматики». На ней администратор может добавлять, изменять и удалять уроки и задания в них. Внешний вид страницы «Добавление грамматики» представлен на рисунке 4.10

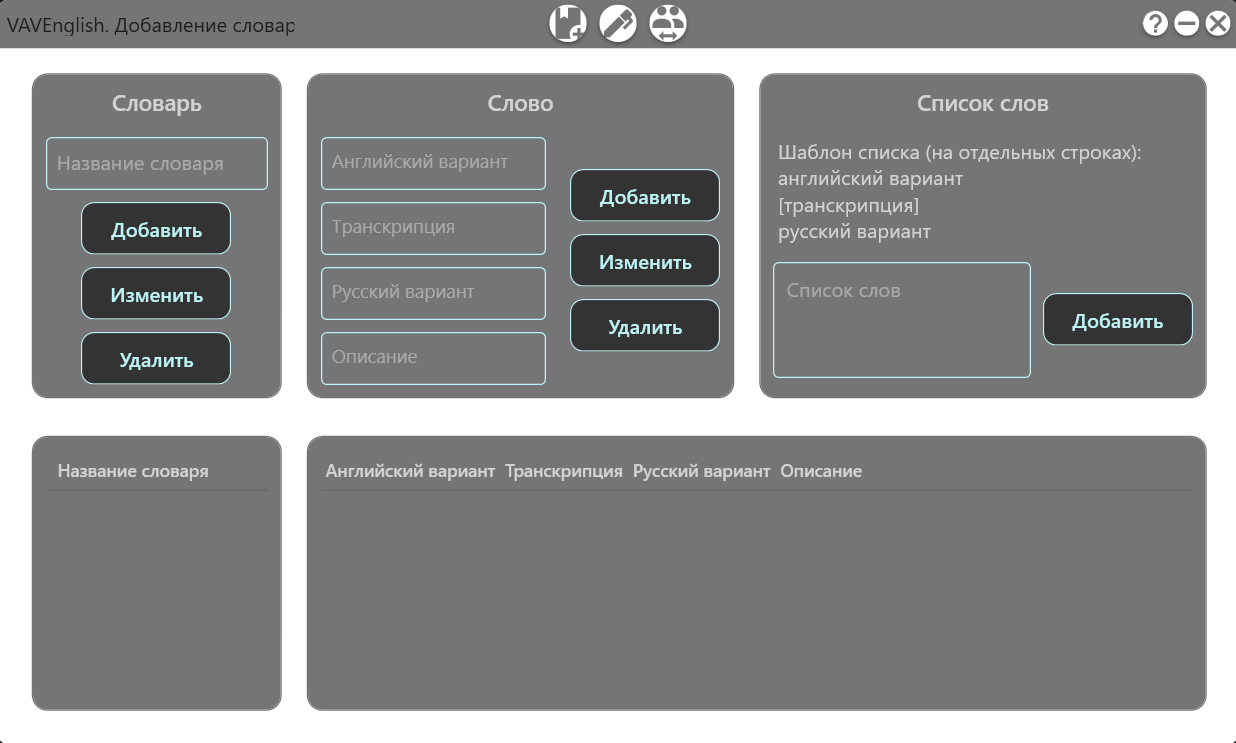


Рисунок 4.10 − Страница «Добавление грамматики»

* 1. **Тестирование программного средства и анализ полученных данных**

Для проверки работоспособности программного средства было проведено его тщательное тестирование, в ходе которого осуществлялась проверка заполняемости базы данных, корректность извлекаемых из базы данных значений, а также реакция программы на вводимые, заведомо неверные, данные. Было проработано множество ситуаций, которые могут возникнуть в ходе работы с программным средством.

Так, при входе осуществляется проверка на действительные электронную почту и пароль. Например, пользователь ввел неверный пароль или логин (результат на рис 4.11):



Рисунок 4.11 − Результат входа с неверной электронной почтой или паролем

Также, например, при попытке войти с каким-либо незаполненным полем, данное поле поменяет цвет границ на красный (результат на рис. 4.12):

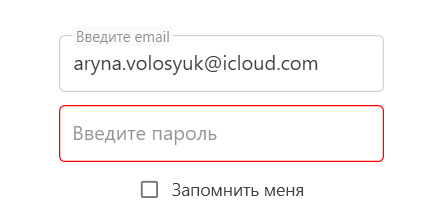


Рисунок 4.12 − Пустые поля при попытке авторизации

При рассмотрении исключительных ситуаций в окне регистрации также была организована проверка вводимых пользователем данных. Так, например, если при регистрации пароль будет меньше 8 символов, то границы поля поменяют цвет на красный (результат на рис. 4.13):

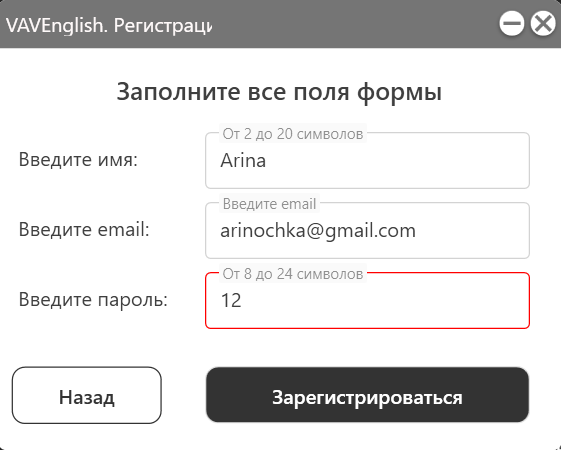


Рисунок 4.13 − Короткий пароль

Также повторное использование электронной почты для создания нового аккаунта недопустимо. Помимо смены цвета границы, также поменяется подсказка (результат на рис 4.14):

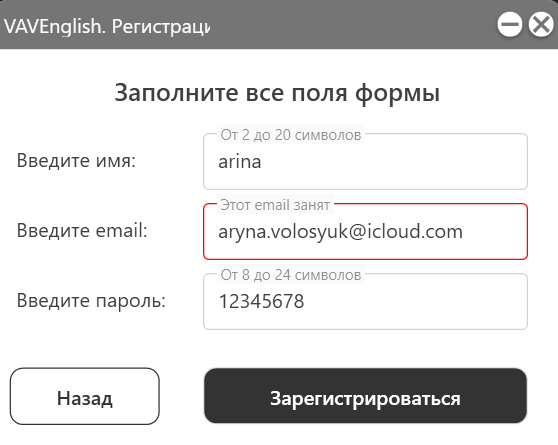


Рисунок 4.14 − Повторное использование электронной почты

При попытке изменить или удалить словарь, предварительно не выбрав его, появляется окно с соответствующим текстом ошибки (результат на рис 4.15):

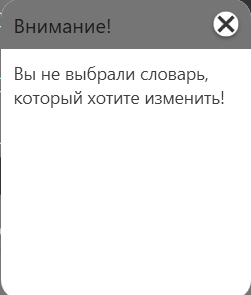


Рисунок 4.15 − Не выбран словарь для манипуляции

Также добавление списка слов, не соответствующих шаблону, вызовет ошибку (результат рис. 4.16):

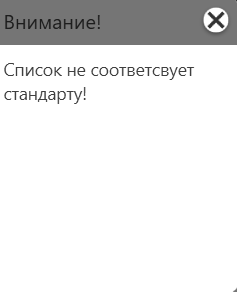


Рисунок 4.16 − Список слов не соответствует шаблону

Пользователь, который хочет сбросить результаты словаря, не выбрав нужный словарь, получит ошибку (результат рис. 4.17):

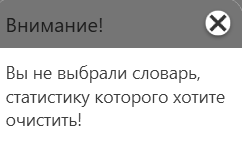


Рисунок 4.17 − Сброс результатов не выбранного словаря

Таким образом, мы ограничиваем пользователя, чтобы вероятность ошибки работы с базой данных свести к минимуму.

1. **Руководство по использованию**

При запуске приложения появляется окно авторизации, которое требует ввода электронной почты и пароля для начала работы.

В случае, если вас нету своей учётной записи, её можно создать, нажав на кнопку «Регистрация» в окне авторизации. Вы будете направлены в окно регистрации, где вам нужно будет заполнить все поля. После корректного заполнения всех полей и нажатия кнопки «Зарегистрироваться» появится окно подтверждения регистрации, в котором нужно ввести четырёхзначный код подтверждения. После успешной регистрации откроется основное окно пользовательского интерфейса.

При наличии аккаунта необходимо ввести корректные данные и войти в аккаунт. Окно авторизации представлено на рисунке 6.1

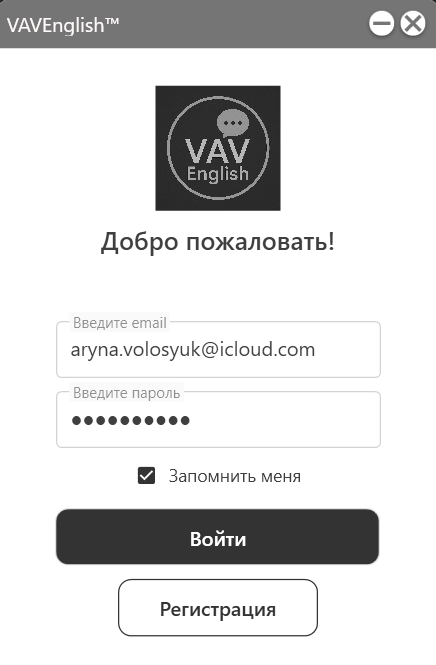


Рисунок 5.1 – «Окно авторизации»

После успешной авторизации пользователем откроется окно со страницей «Статистика» на рисунке 5.2. На странице отображается личный прогресс за последние 2 недели по правильным ответам слов и грамматических заданий в виде графиков. Также есть статистика всех пользователей в виде таблицы за день, неделю и всё время.

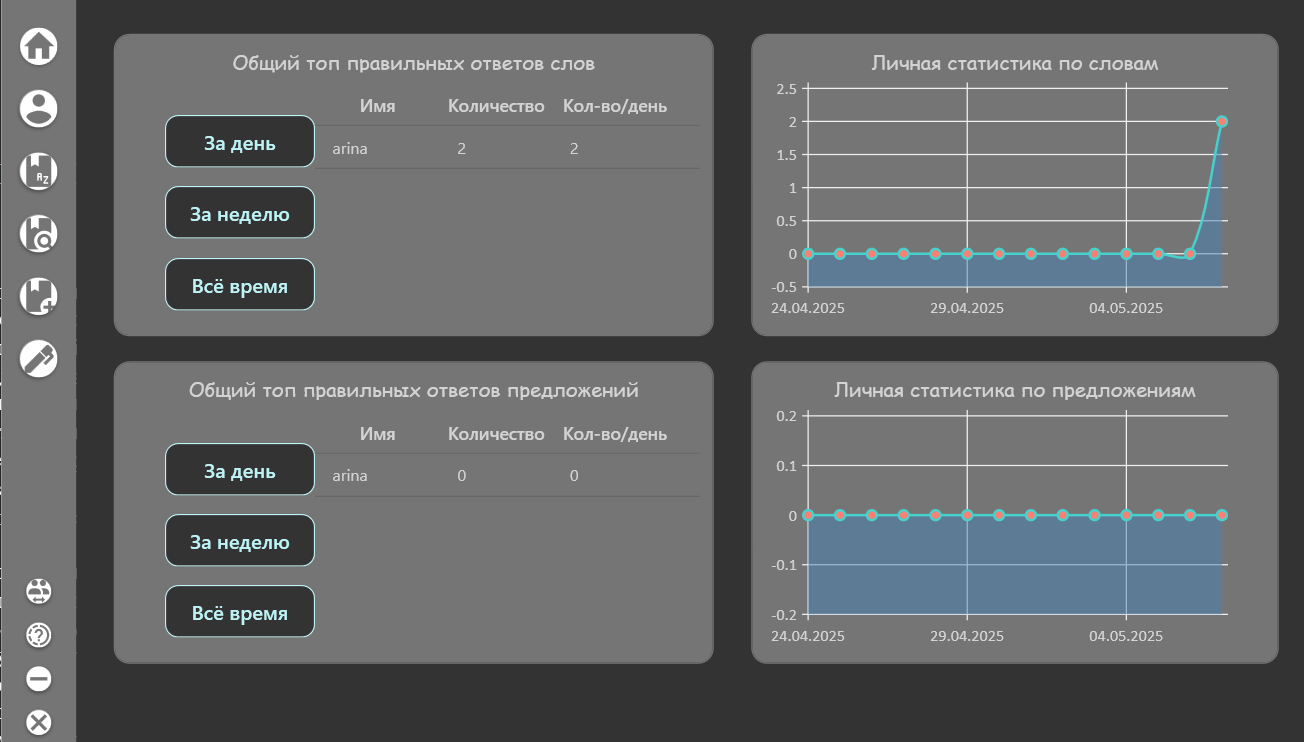


Рисунок 5.2 − Страница – «Статистика»

Переключение между страницами происходит за счёт нажатия на нужную кнопку на боковой панели окна. Помимо кнопок переключения окно на панели есть кнопки «Пользовательское руководство», «Свернуть» и «Закрыть».

Следующей страницей является «Личный кабинет», скриншот которого представлен на рисунке 5.3. Информативную часть окна составляют данные пользователя, а именно имя и электронная почта. Функциональная часть это изменение пароля аккаунта и имени, также настройки обучения.



Рисунок 5.3 − Страница – «Личный кабинет»

Далее следует страница «Тренировка слов», представленная на рисунке 5.4. Для начала тренировки необходимо выбрать словарь. В разделе имеется 3 режима изучения слов: ввод перевода вручную, выбор перевода из набора слов, нахождение пар слово-перевод. Общее количество слов изучаемых одновременно можно настроить в личном кабинете. Если слов для изучения недостаточно в словаре, то не все режимы изучения будут доступны, так как в каждом из них находятся уникальные слова.

Каждый верный ответ слова фиксируется и при достижении необходимого количества правильных ответов подряд (устанавливает пользователь в личном кабинете) слово исключается из тренировочных режимов.

В каждом режиме доступна озвучка текущего слова. Для того, чтобы озвучить слово в режиме «Пары слово-перевод», необходимо нажать на нужное слово.

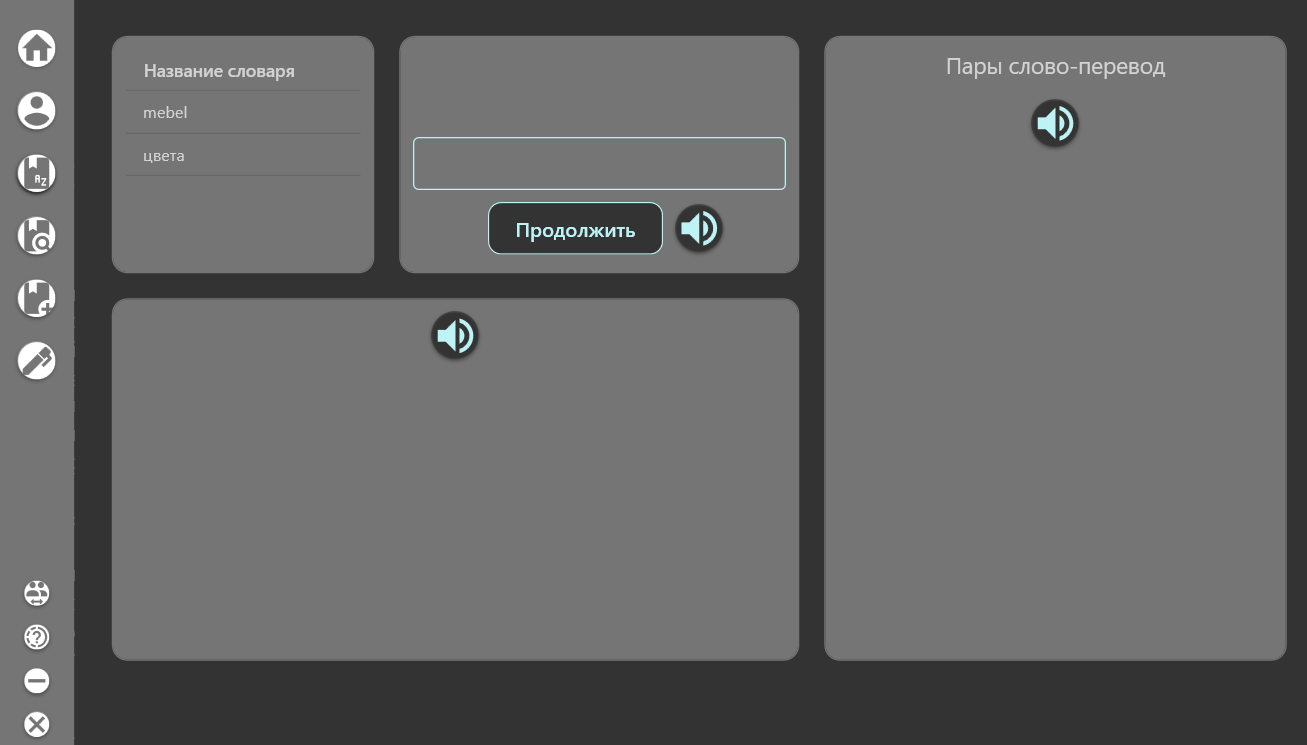


Рисунок 5.4 − Страница – «Тренировка слов»

Окно «Просмотр словарей», представленное на рисунке 5.5, содержит в себе совокупность словарей, добавленных как самим пользователем, так и администраторами. На этой странице можно просматривать слова по словарям с дополнительной информацией, такой как количество верных ответов на слово подряд и состояние слова на данный момент (выучено или нет).

При необходимости тренировать слова после того, как они были выучены, есть возможность сбросить результаты тренировок для одного словаря или для всех сразу. После сброса слова будут снова доступны для тренировок на странице «Тренировка слов».

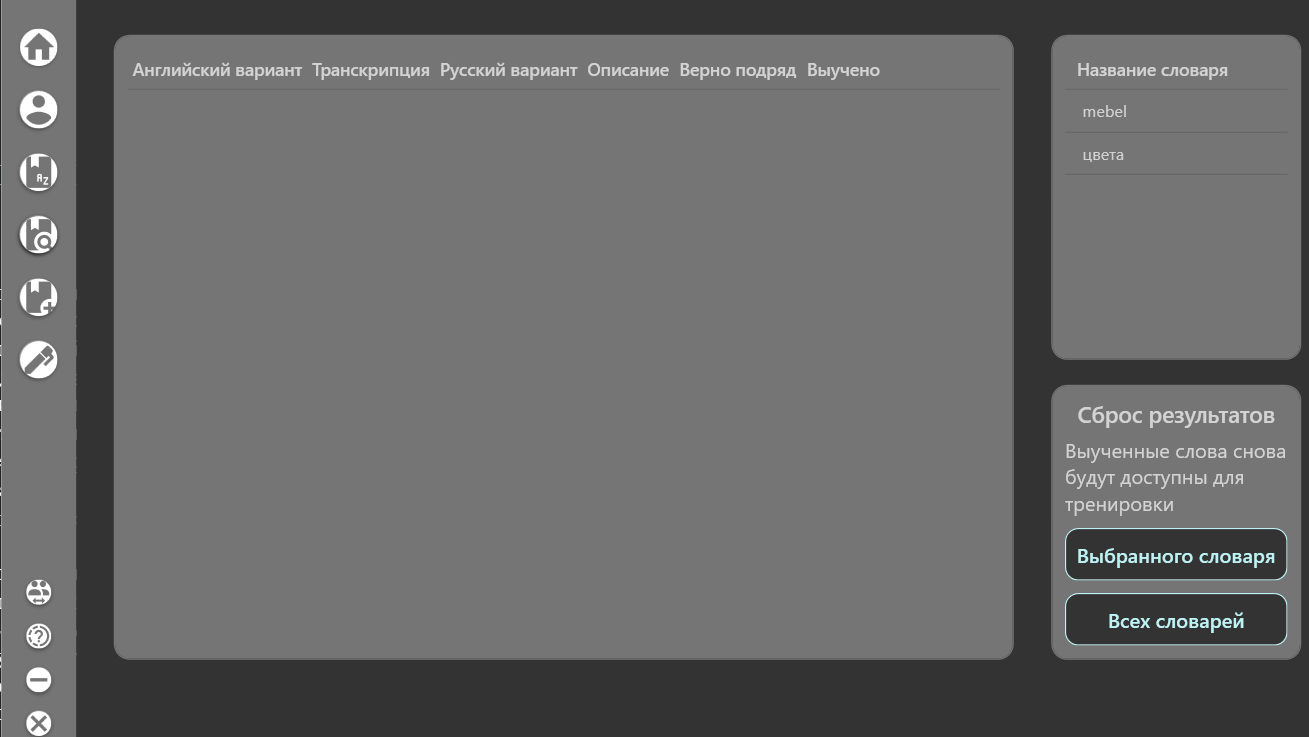


Рисунок 5.5 − Страница «Просмотр слов»

Следующая страница «Добавление словарей» представлена на рисунке 5.6. Страница является общей для пользователя и администратора. Различия лишь в том, что администратор видит словари, которые добавили администраторы, а пользователь видит собственные словари.

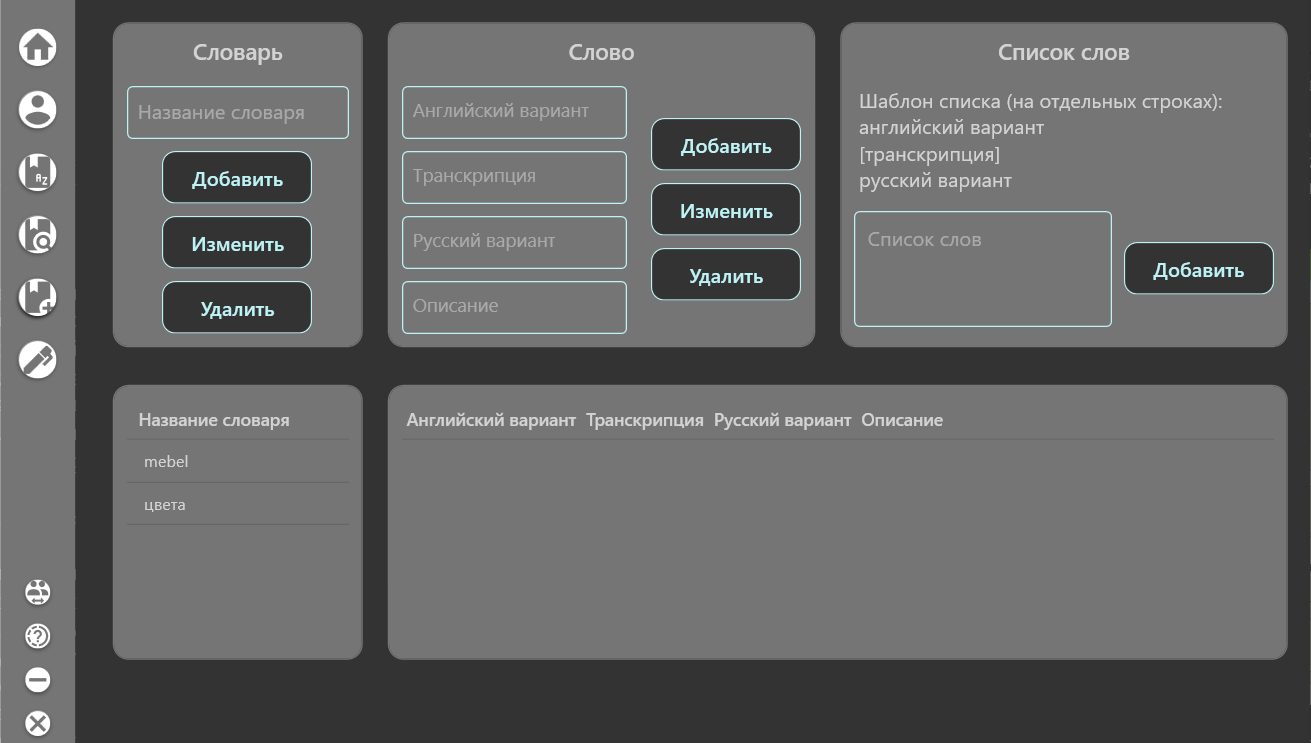


Рисунок 5.6 − Страница «Добавление словарей»

Функционал: пользователь/администратор может добавлять, изменять или удалять словари и слова в них. Добавление слов в словарь возможно по одному слову или целым списком. Так же для избежания недопонимания перевода некоторых слов можно добавить описание к слову. В добавлении списком данная опция отсутствует, так как процент слов, которым необходимо описание, очень мал. Рациональнее добавить список слов не описания, а после добавить его к существующему слову.

Далее следует страница «Тренировка грамматики», которая представлена на рисунке 5.7. Изначально пользователю доступен один урок. Для получения доступа к следующему уроку пользователь должен пройти данный урок не менее, чем на проходной процент этого урока. В уроке несколько заданий, суть которых правильно перевести предложение путём выбора в верном порядке кнопок со словами. Внизу страницы доступна теория, которая поможет разобраться в грамматических правилах и дать верные ответы.

При необходимости можно озвучить текст для перевода с помощью кнопки, находящей слева от текста для перевода.

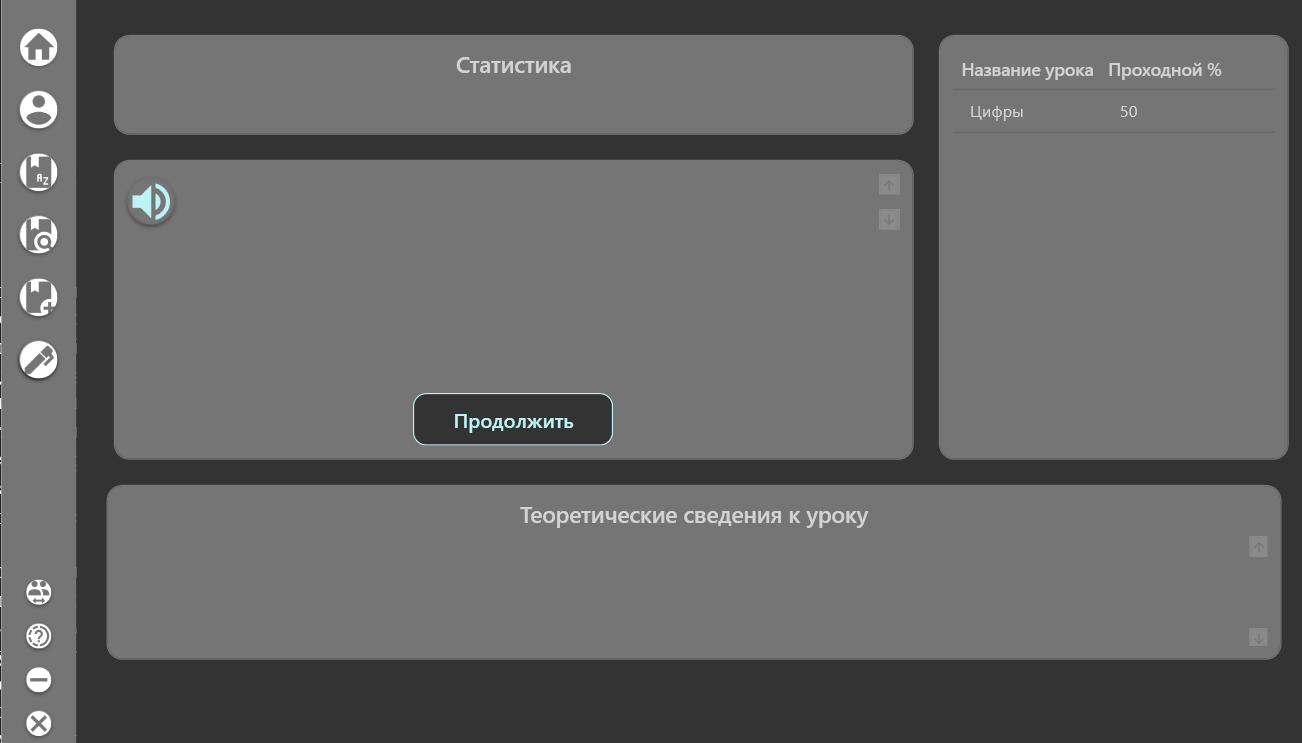


Рисунок 5.7 − Страница «Тренировка грамматики»

Следующая страница доступна только для администраторов, а именно страница «Добавление грамматики», представленная на рисунке 5.8. На этой странице администратор может добавлять, изменять или удалять уроки и заданиях в них. Уроки и задания представлены в виде таблиц, что делает просмотр данных и манипуляции над ними удобным.

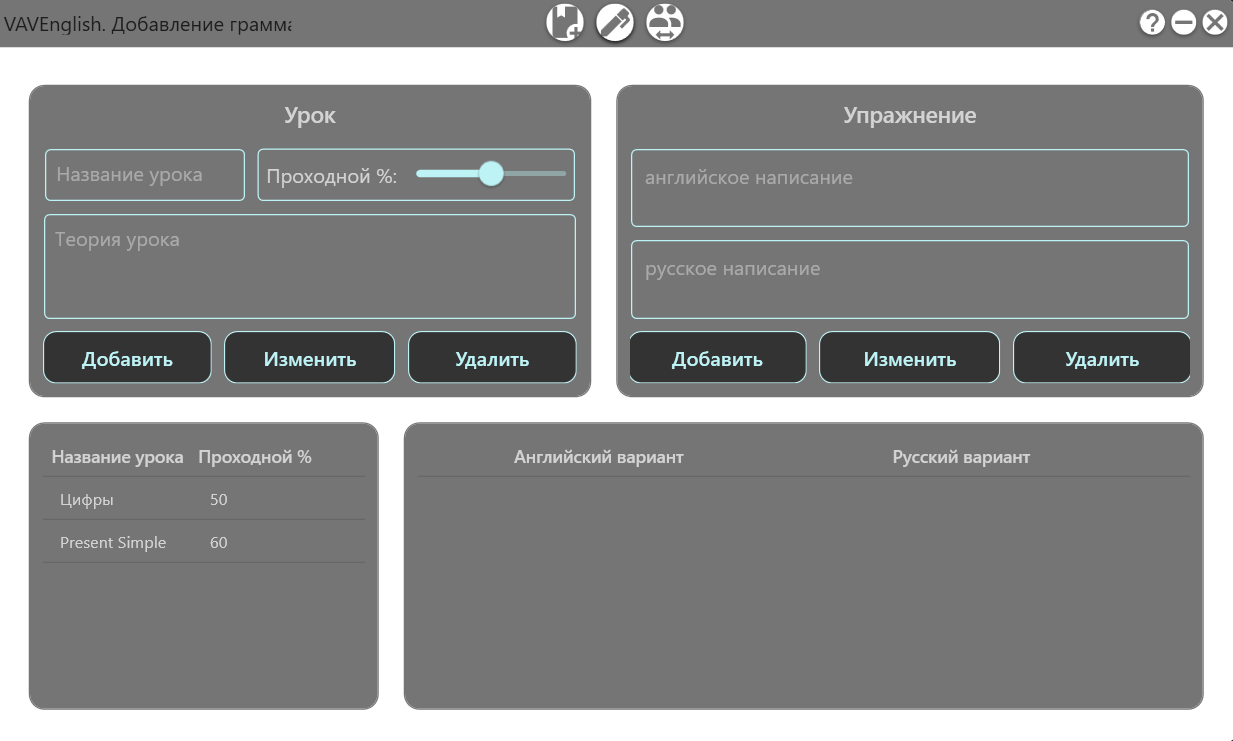


Рисунок 5.8 − Страница «Добавление грамматики»

Окно «Пользовательское руководство», представлено на рисунке 5.9.



Рисунок 5.9 − Окно «Пользовательское руководство»

В нём содержится текст, объясняющий пользователю нюансы работы в приложении. Для обычных пользователей и администраторов содержимое окна разное.

**Заключение**

В ходе выполнения курсовой работы был разработан проект, являющийся помощником в изучении английского языка. Во время выполнения данной курсовой работы было изучено немало теоретического материала, также просмотрено и разобрано большое количество уже готовых решений тех или иных задач.

В ходе выполнения курсовой работы было выполнено следующее:

* разработка базы данных для хранения информации;
* разработка архитектуры приложения;
* разработка функциональных части приложения;
* разработка пользовательского интерфейса;
* написание исходного кода приложения;
* тестирование приложения.

Проектирование осуществлялось в соответствии с паттерном проектирования MVVM. Для абстрагирования и удобного потенциального расширения приложения были реализованы паттерны работы с данными Repository и Unit of Work. Для разработки дизайна использовалась библиотека MaterialDesign и библиотека LiveCharts для графического представления данных. Использована библиотека Speech для озвучки текста.

В соответствии с полученным результатом работы программы можно сделать вывод, что разработанная программа работает, верно, а требования технического задания выполнены в полном объеме.

Само приложение – первый опыт в данной области. Естественно, существует ряд приложений, имеющих функционал и дизайн в разы лучше. Подводя итог всей курсовой работе, можно сделать выводы, что в разработке приложений мне предстоит еще долгий путь.

**Список использованных источников**

1. Garofallo, R. Building Enterprise Applications with Windows Presentation Foundation and the Model View ViewModel Pattern / R. Garofallo − California: O’Reilly Media, − 2011 − 220 p.
2. Информационный сайт: Diabetes [Электронный ресурс]. Режим доступа: https://play.google.com/store/apps/details?id=ru.zengalt.simpler&hl=ru&gl=US. Дата доступа: 16.04.2025.
3. Информационный сайт: Duolingo [Электронный ресурс]. Режим доступа: https://play.google.com/store/apps/details?id=com.duolingo&hl=ru&gl=US. Дата доступа: 17.04.2025.
4. Информационный сайт: базы данных [Электронный ресурс]. Режим доступа:[inform.sch117nn.edusite.ru/DswMedia/klyaks\_netuchitelyutestyibazyidannyix.htm](http://inform.sch117nn.edusite.ru/DswMedia/klyaks-_netuchitelyutestyibazyidannyix.htm) – Дата доступа: 24.04.2025.
5. Информационный сайт: Полиглот [Электронный ресурс]. Режим доступа: https://play.google.com/store/apps/details?id=com.axidep.poliglot&hl=ru&gl=US. Дата доступа: 02.05.2025.
6. Сайт о программировании Metanit: Введение в Entity Framework [Электронный ресурс]– Режим доступа: https://metanit.com/sharp/efcore/1.1.php – Дата доступа: 24.04.2025.