Schedule Management Studio V.1.00

**Аннотация:**

Идея приложения заключается в удобном доступе преподавателей к бронированию свободных кабинетов школы на конкретную дату и номер урока, а так же в свободном доступе учеников к расписанию своих преподавателей, для наиболее удобной ориентации в расписании.

**Адрес основного репозитория:**

<https://github.com/ivanvershinin/ScheduleManagement>

**Участники проекта и их задачи:**

1. Бордунова Анжелика Эдуардовна – разработка пользовательского интерфейса, разработка логики программы, мотивационная функция

2. Вершинин Иван Алексеевич – разработка базы данных, разработка логики программы, функция сохранения спокойствия в команде

**Список классов и их описание:**

**ScheduleManagement.Logic:**

Model (Cabinet/School/Tutor/TutorCabinet) **–** папка, содержащая классы описания сущностей и их связи: Школа, Кабинет, Учитель, УчительКабинет; для создания базы данных.

Repository (CabinetRepository, SchoolRepository, TutorRepository, TutorCabinetRepository, Repository(parent class)) – папка, содержащая классы репозиториев, осуществляющих доступ к данным из БД, созданию/чтению/обновлению/удалению данных БД.

Authorization – класс, содержащий в себе функцию хеширования паролей.

UnitOfWork – класс, содержащий реализацию паттерна UnitOfWork

Context – класс, осуществляющий связь с локальным сервером для создания модели и таблиц базы данных на основе классов папки Model.

Storage – класс-хранилище для данных текущей сессии.

**ScheduleManagement.GUI:**

Папка Pages, содержащая в себе следующие страницы пользовательского интерфейса:

AccountPage – страница, позволяющая пользователю настроить параметры для поиска свободных кабинетов, зайти в режим просмотра расписания, выйти из аккаунта.

GuestViewPage – страница, позволяющая незарегистрированным пользователям просмотреть расписание учителя на любую дату.

LoginPage – страница, выполняющая функцию авторизации пользователя.

RegistrationPage – страница, позволяющая пользователю при вводе электронной почты, пароля, имени и фамилии зарегистрироваться в системе.

StartingPage – страница, выполняющая 3 перехода: в режим регистрации, авторизации, просмотра расписаний в качестве гостя.

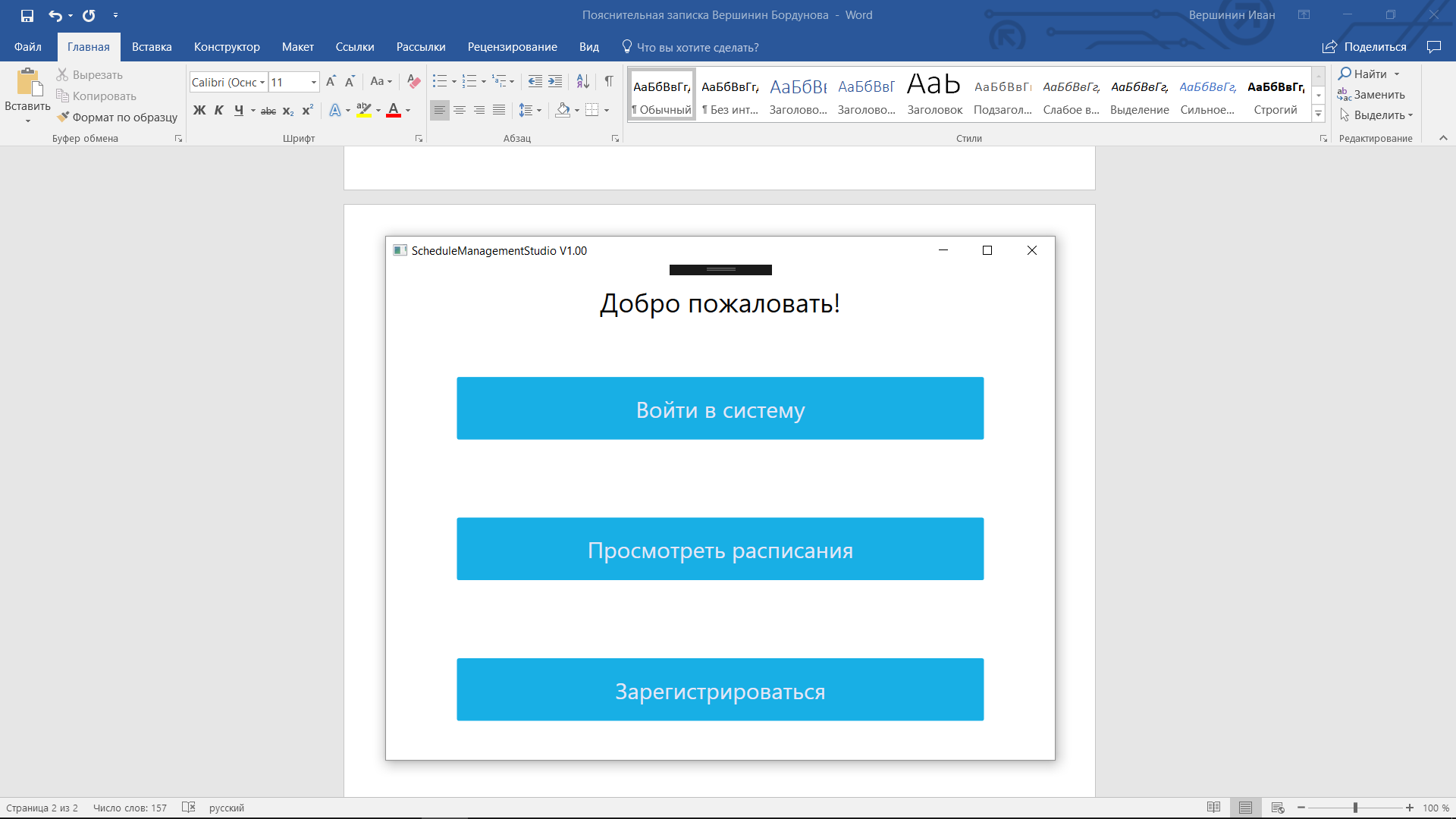
ViewPage – страница, на которой отображены удовлетворяющие введенным параметрам свободные кабинеты, которые можно выбрать и занять.

ViewSchedulePage – страница, на которой зарегистрированный пользователь может просмотреть свое расписание на любую дату.

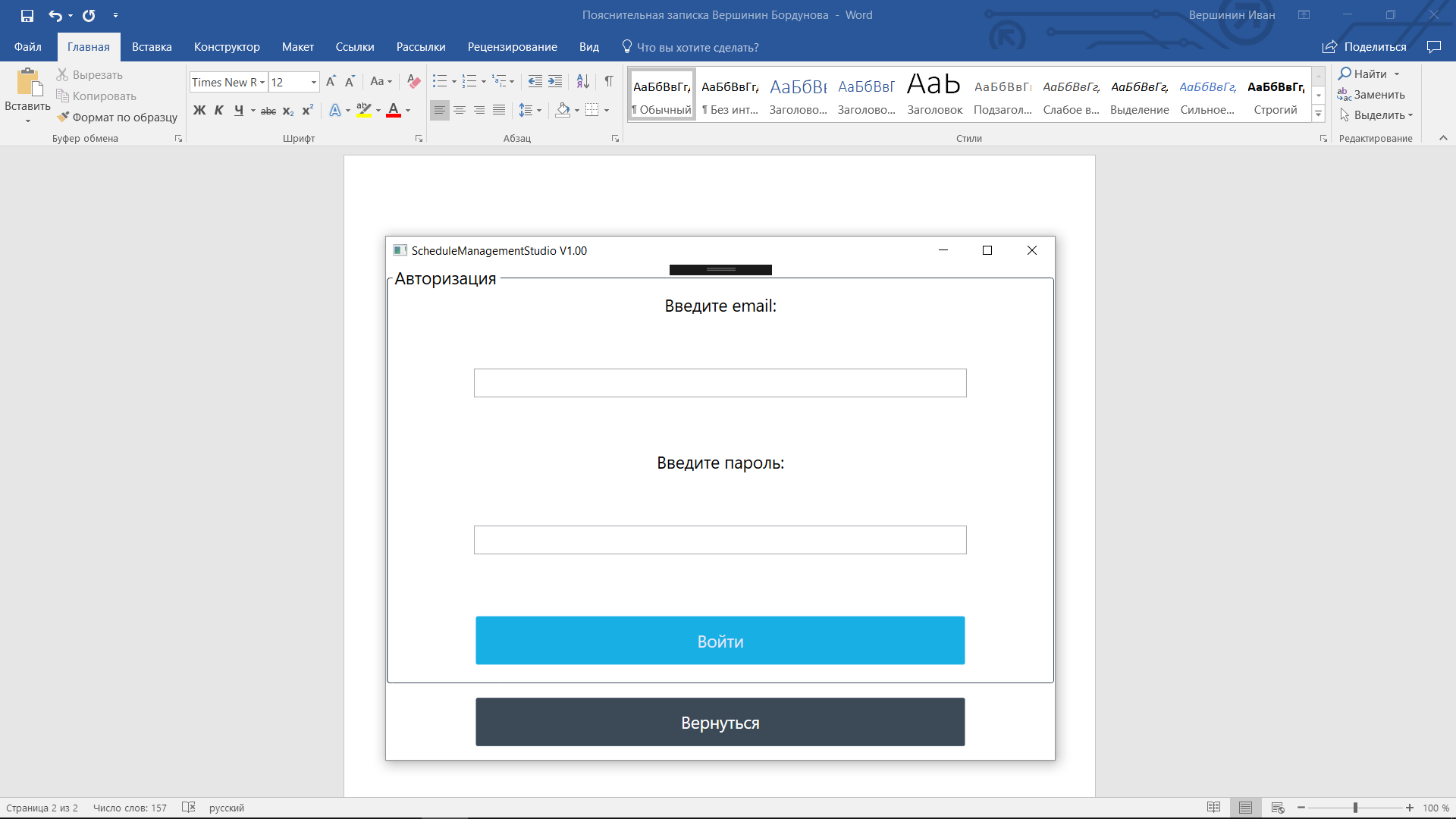
PagesStorage – класс, реализующий паттерн factory для хранения страниц.

**Скриншоты приложения:**

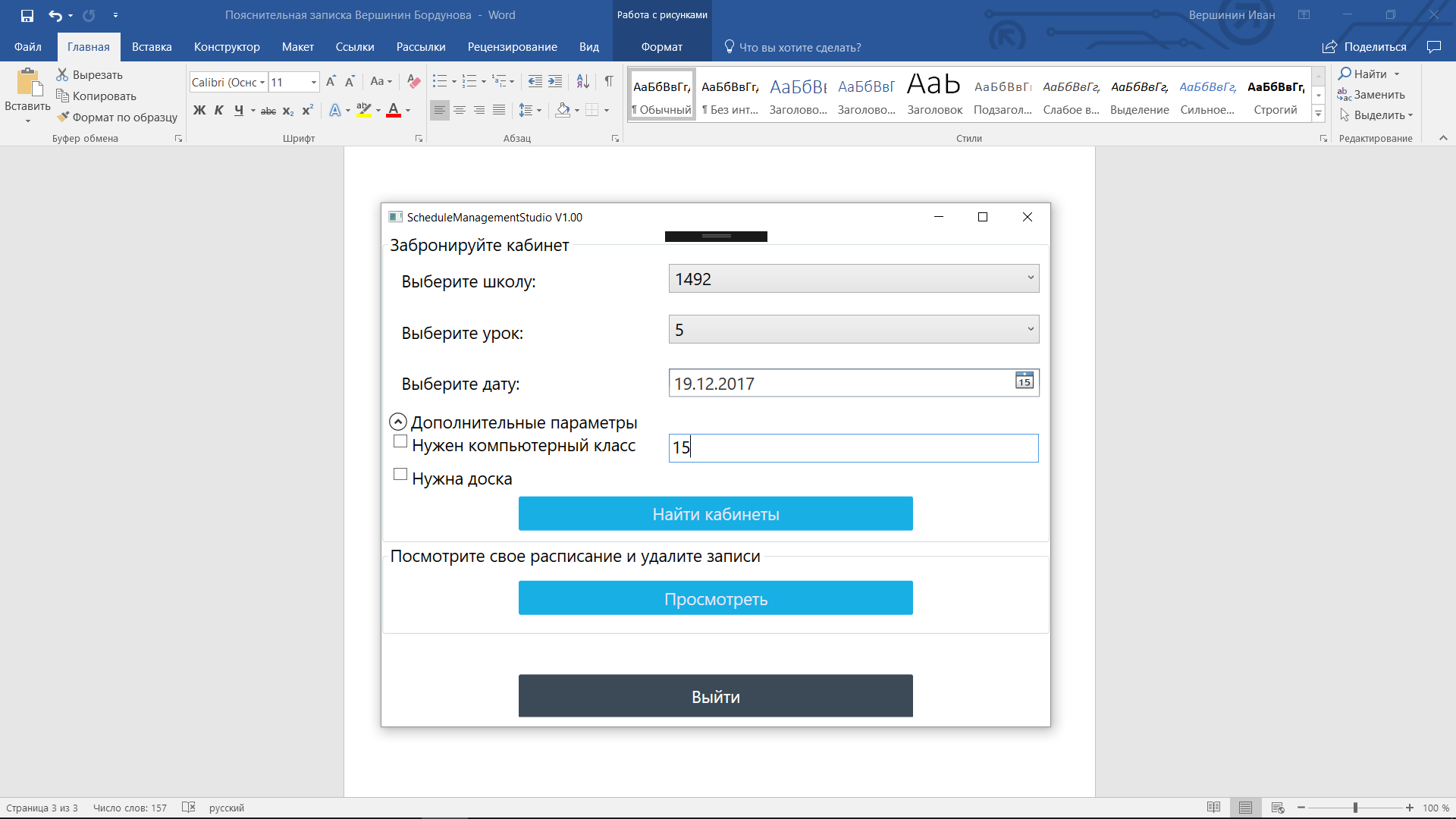
1. Главная страница:



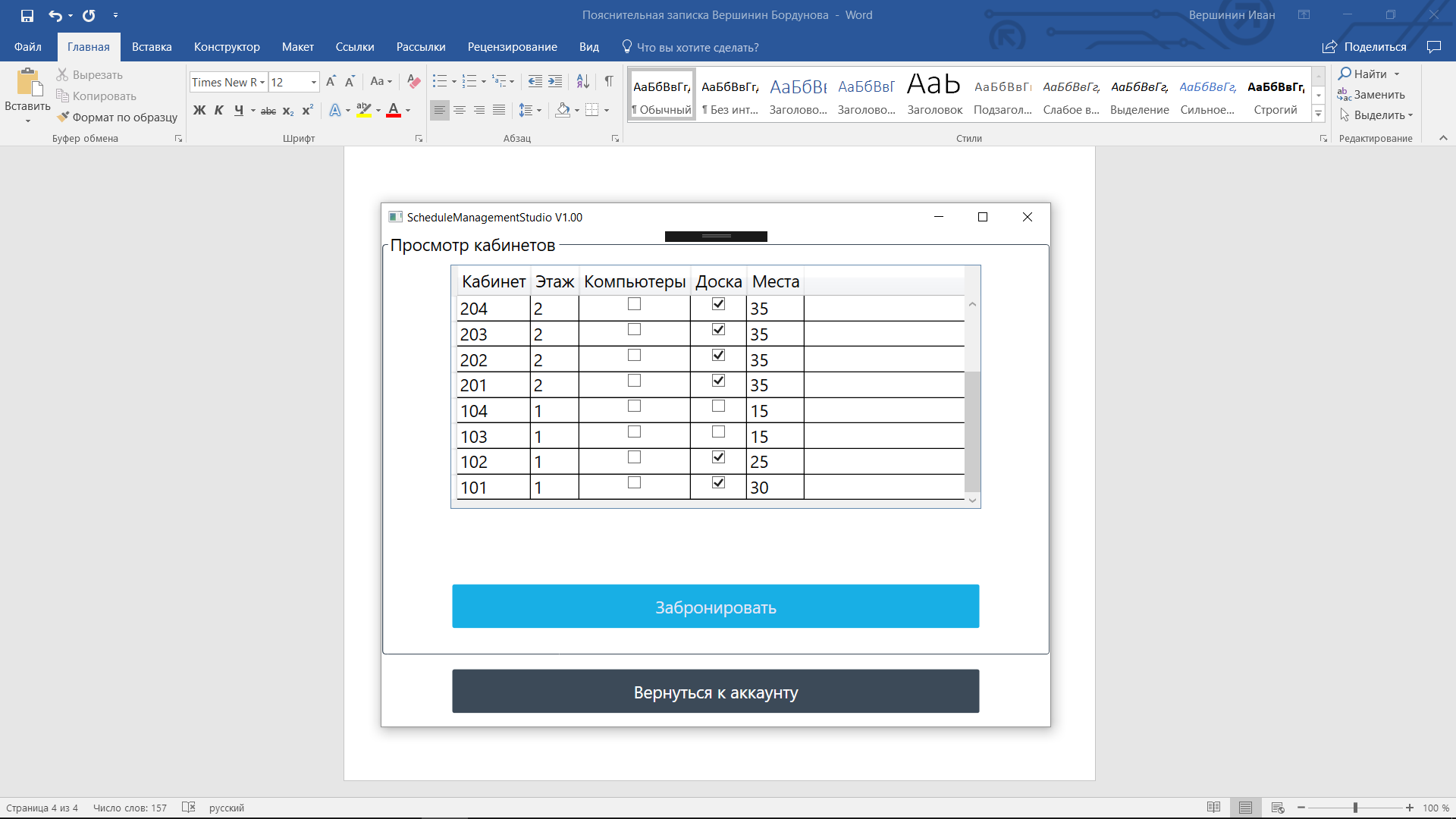
1. Страница авторизации:



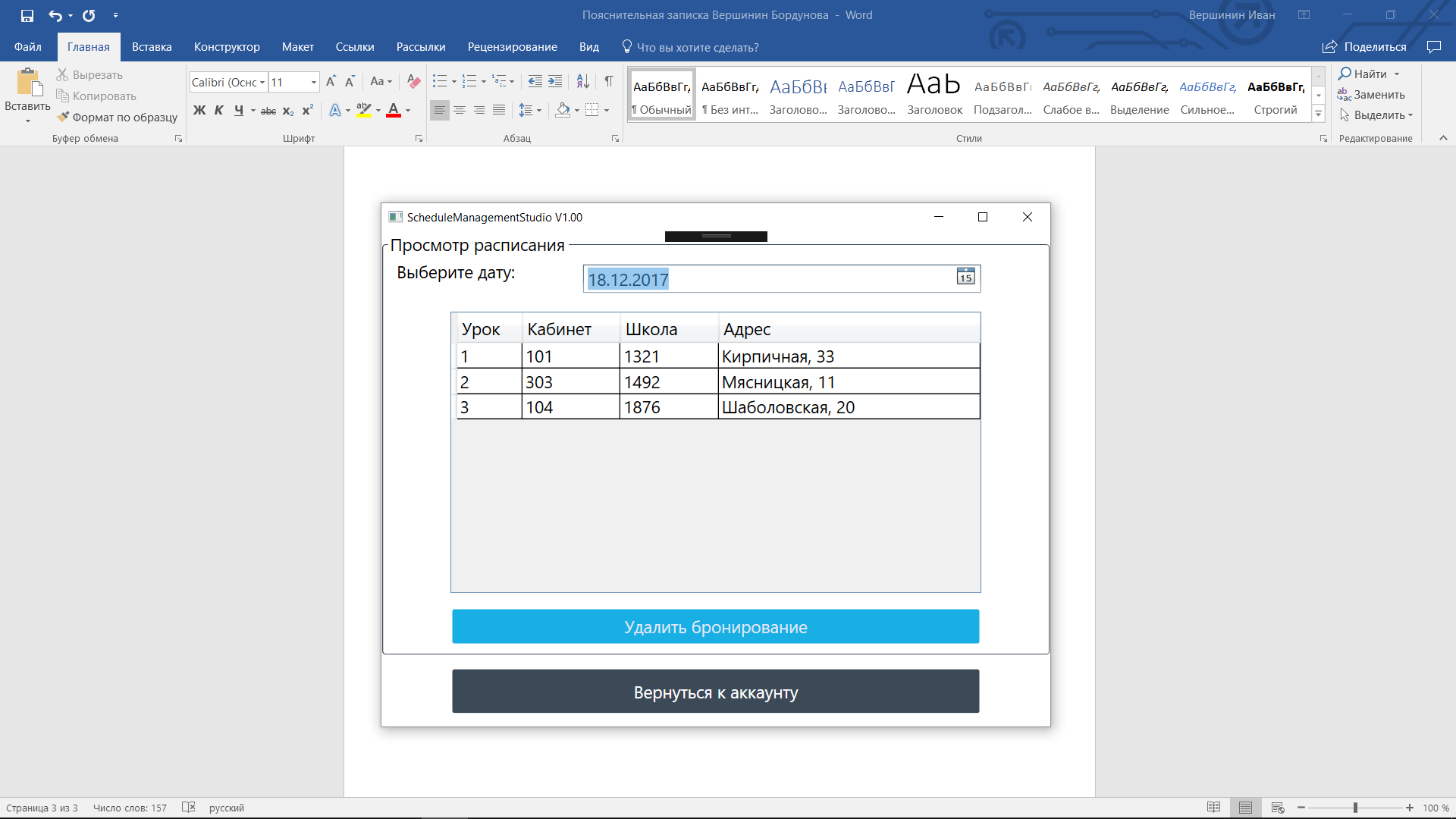
1. Личная страница пользователя:



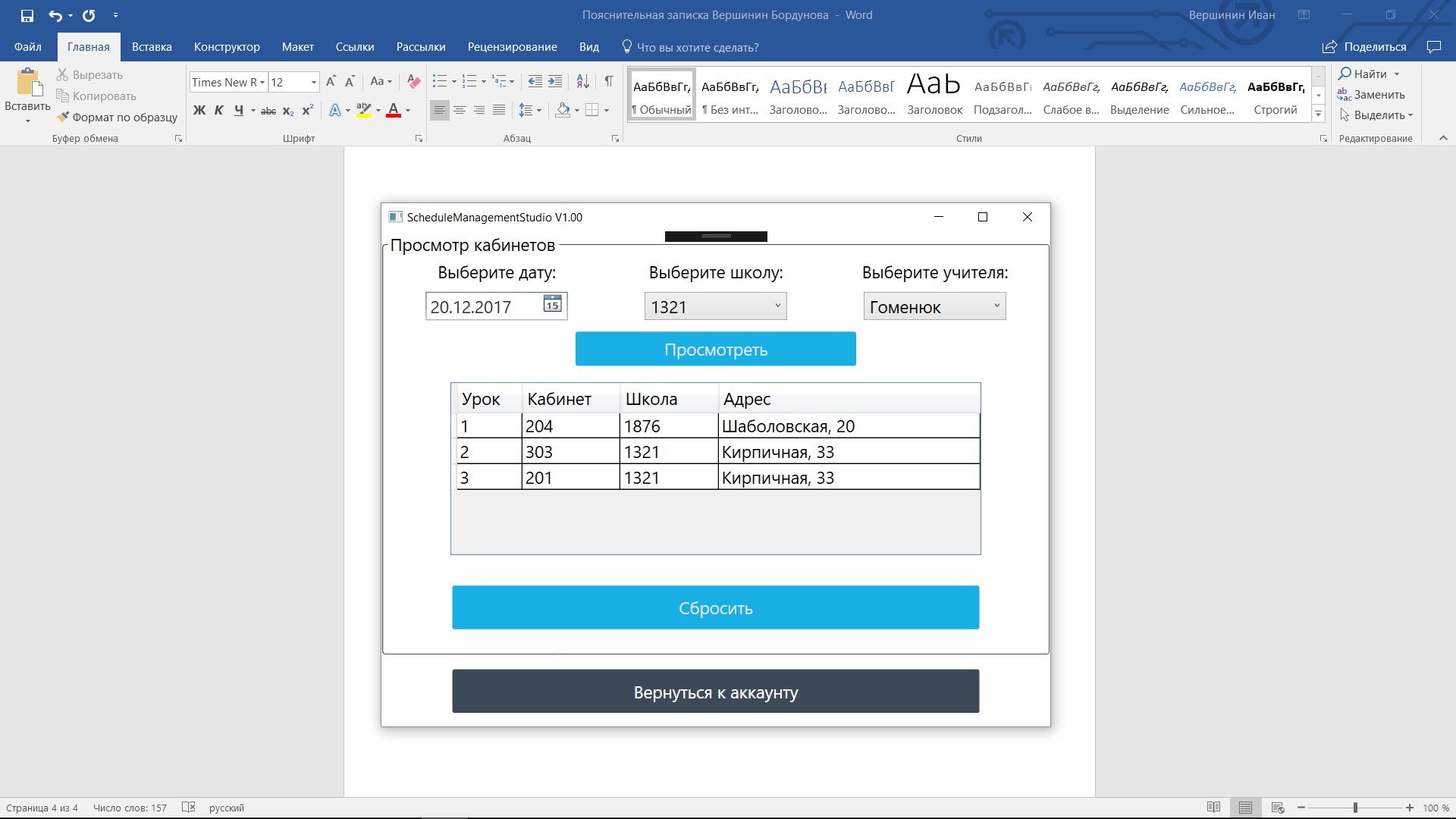
1. Страница бронирования кабинета:



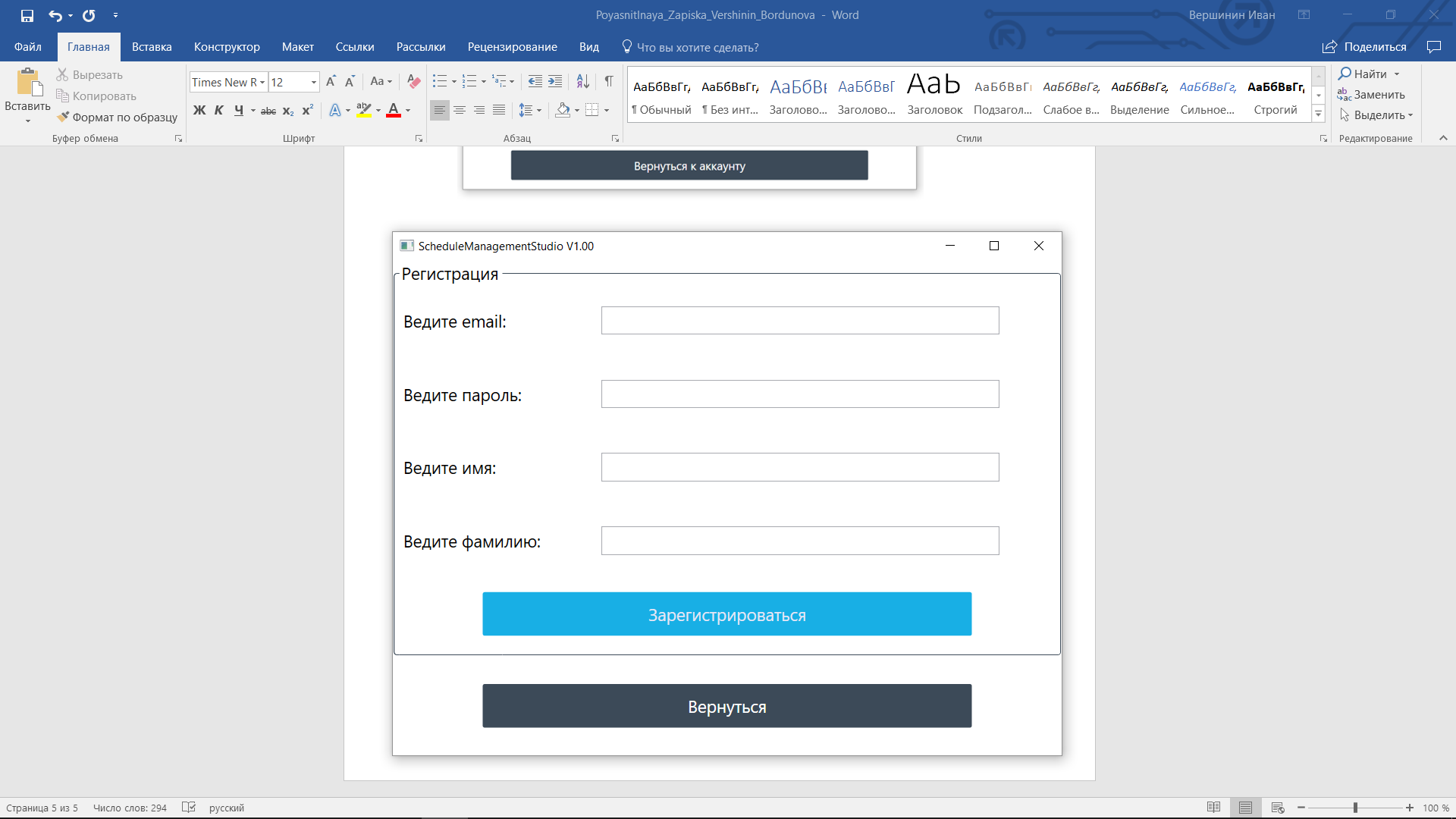
1. Страница просмотра личного расписания пользователя:



1. Страница просмотра расписания гостем:



1. Страница регистрации:



**Test cases:**

1. Попытка зарегистрировать пользователя с существующим емеилом.
2. Попытка регистрации пользователя без заполнения всех полей.
3. Попытка входа в аккаунт с некорректными данными.
4. Попытка бронирования кабинета без заполнения всех обязательных полей.
5. Попытка бронирования кабинета на дату раньше сегодняшней.
6. Попытка бронирования двух первых уроков на одну дату одним пользователем.
7. Проверка корректности отображения расписания после удаления брони.

Спасибо за внимание!

(Все элементы данного проекта являются интеллектуальной собственностью, использование элементов данного проекта коммерческими организациями в целях получения прибыли строжайше запрещено)