**Теоретический материал**

**Ajax подгрузка контента на jQuery**

Иногда стоит задача реализовать AJAX загрузку контента с сервера без перезагрузки страницы. Например, обновить комментарии или подгрузить полный текст статьи при клике на кнопку «Подробнее». AJAX используется, для добавления что-то нового на страницу без перезагрузки. Когда делали [форму обратной связи на сайте](https://thecode.media/php-form/), то она работала так: в вводите сообщение и нажимаете «отправить»; загружается новая страница с подтверждением отправки. Часто это неудобно: хочется оставить посетителя на странице, а не показывать другую страницу.

AJAX – это аббревиатура от «Asynchronous JavaScript and XML», которая дословно переводится как «асинхронный JavaScript и XML». Для асинхронной загрузки в jQuery есть удобный метод [jQuery **load(**)](http://api.jquery.com/load/). Он делает асинхронный запрос на сервер и вставляет полученные данные (HTML разметку) в **указанный элемент (**блок, например блок <div id=" result "></div> ).

$("#result").load("ajax.html");

**Загрузка фрагмента страницы**

Метод .load() в отличии от $.get(), позволяет нам извлечь часть загружаемой страницы. Этого можно достичь, если после адреса файла указать селектор той области страницы, которую нужно извлечь.

Можно чуть исправить пример, указанный выше, чтобы извлечь только часть документа:

|  |
| --- |
| $('#result').load('ajax/test.html #container'); |

**Практические задания**

**Задание 1.** Скопируйте пример в папке new\_folder на свой локальный сервер. Протестируйте его.

**Задание 2.** Разработайте SPA-приложение

Необходимо реализовать SPA приложение, которое потом можно будет подключить к уже разработанной API. Приложение необходимо разместить на локальном сервере (в папке C:\xampp\htdocs). Необходимо реализовать **переход между экранами с подгузкой их на главную страницу (Главный экран)**. **Связь с БД устанавливать не надо! Информация с экранов будет только отображаться статически!**

Ваше SPA должно состоять из следующих экранов:

* Главная
* Вход в личный кабинет
* Регистрация в личном кабинете
* Личный кабинет
* Результаты поиска
* Бронирование
* Управление бронированием
* Выбор места в салоне воздушного судна

**В задании даны примерный интерфейс форм, необходимо выполнить дизайн экранов**

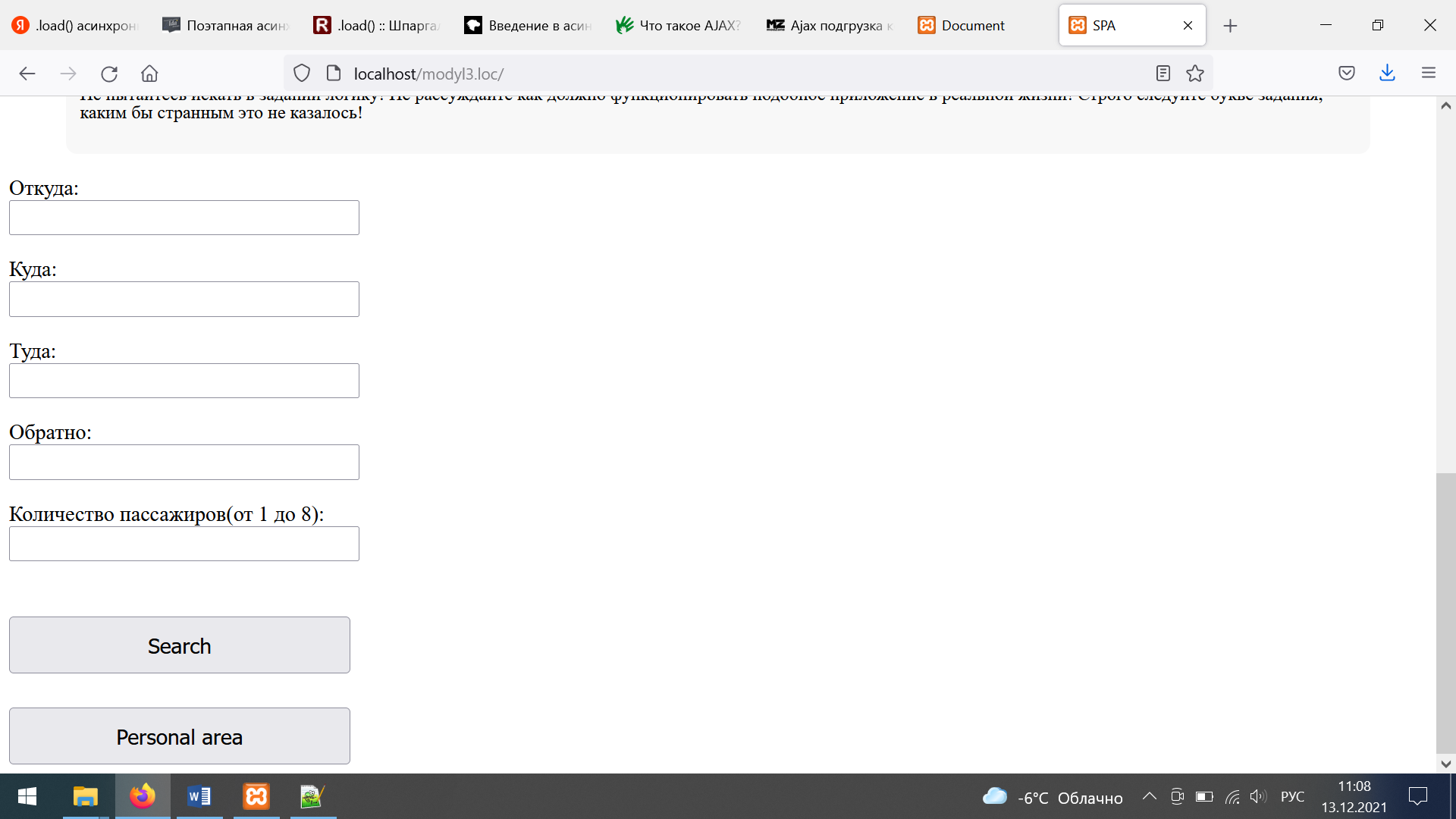
**Главный экран**

На первом экране приложения расположена форма для ввода пользовательской информации для поиска рейсов. При отправке формы должен выводиться перечень найденных рейсов. "Результаты поиска" будут отображаться два рейса туда и два - обратно.

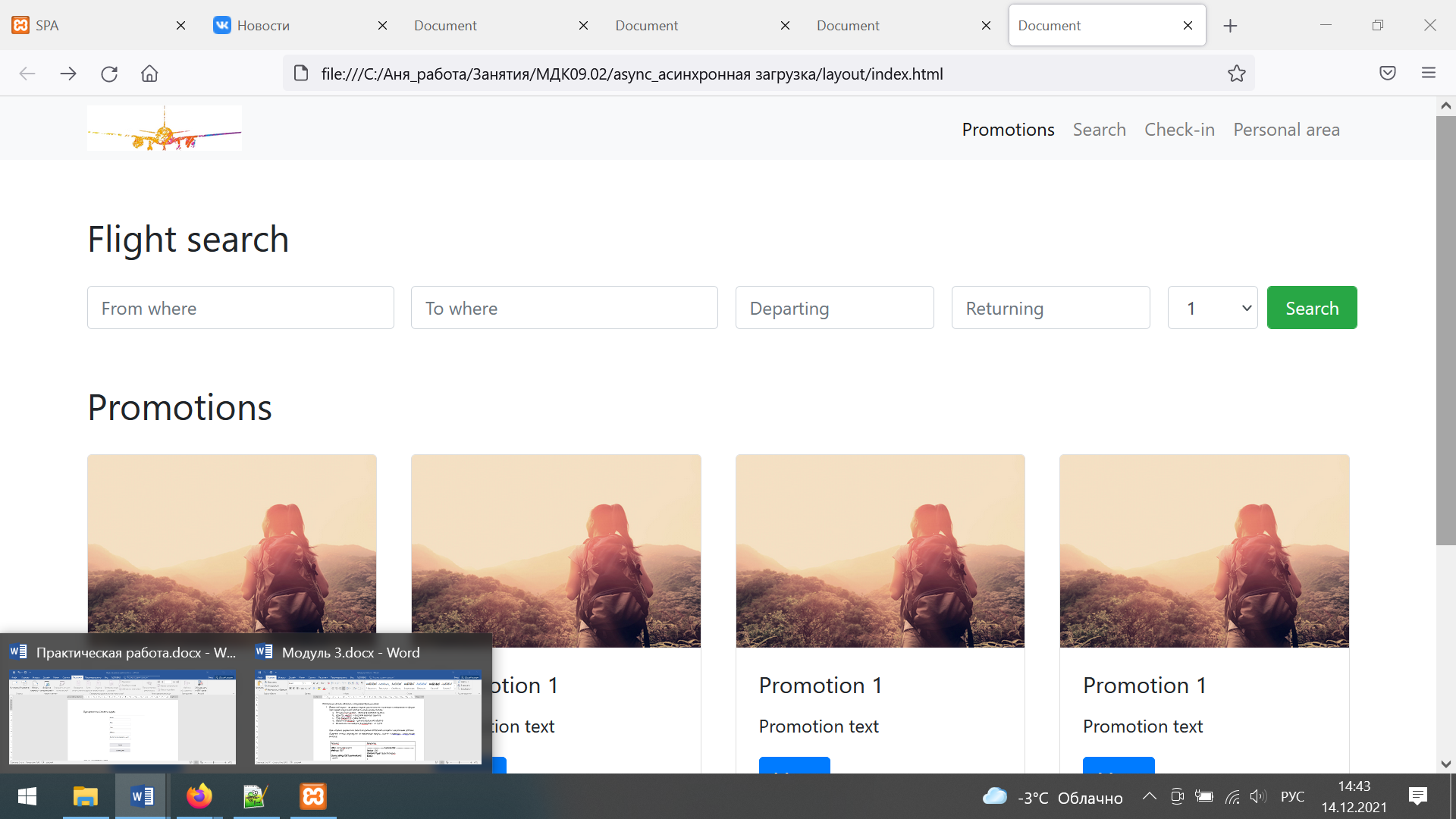
Нажатие на кнопку “Personal area” (Аутентификация) ведет на экран аутентификации.

Нажатие на кнопку “Search” (Поиск) ведет на экран с найденными рейсами.

Примерный вид Главного экрана.



Или другой вид, можете выполнить на выбор или свой вариант

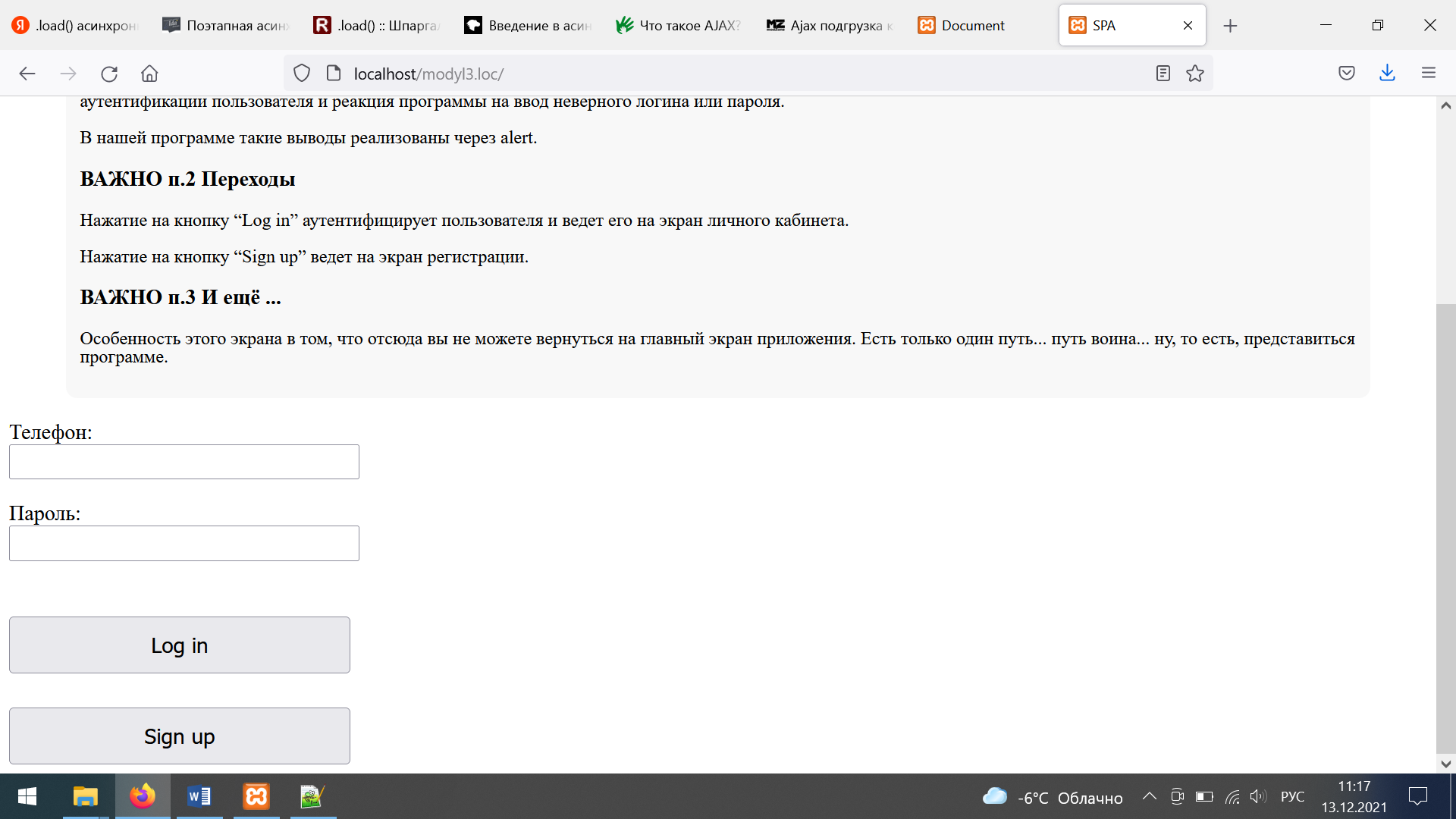


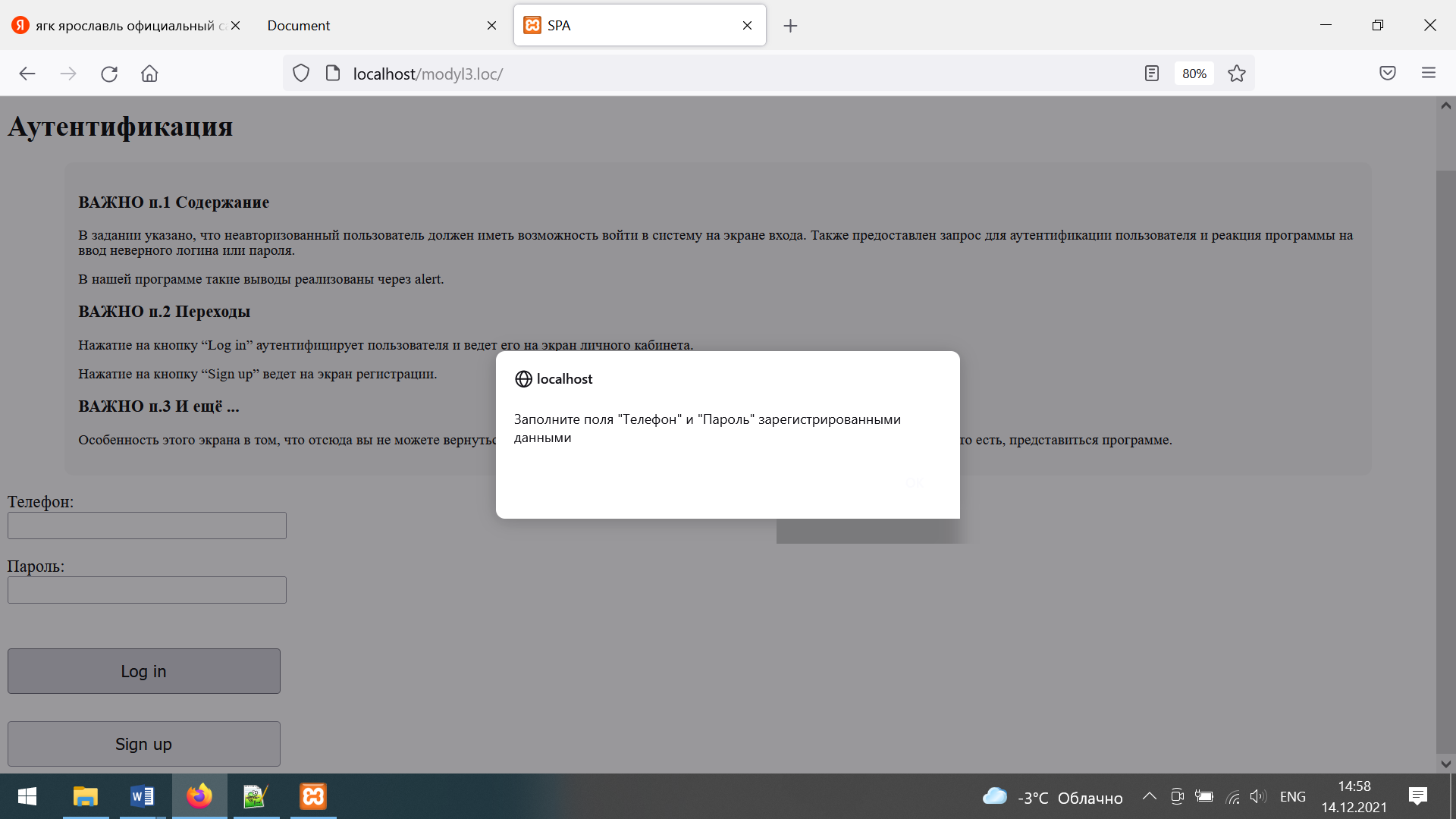
**Экран аутентификации (**Personal area**)**

Нажатие на кнопку “Log in” (Авторизация) пользователь проходит аутентификацию и переходит его на экран личного кабинета.

Нажатие на кнопку “Sign up”(Регистрация) ведет на экран регистрации.

Примерный вид экрана. Пользователь не может пройти авторизацию, если не заполнил поля





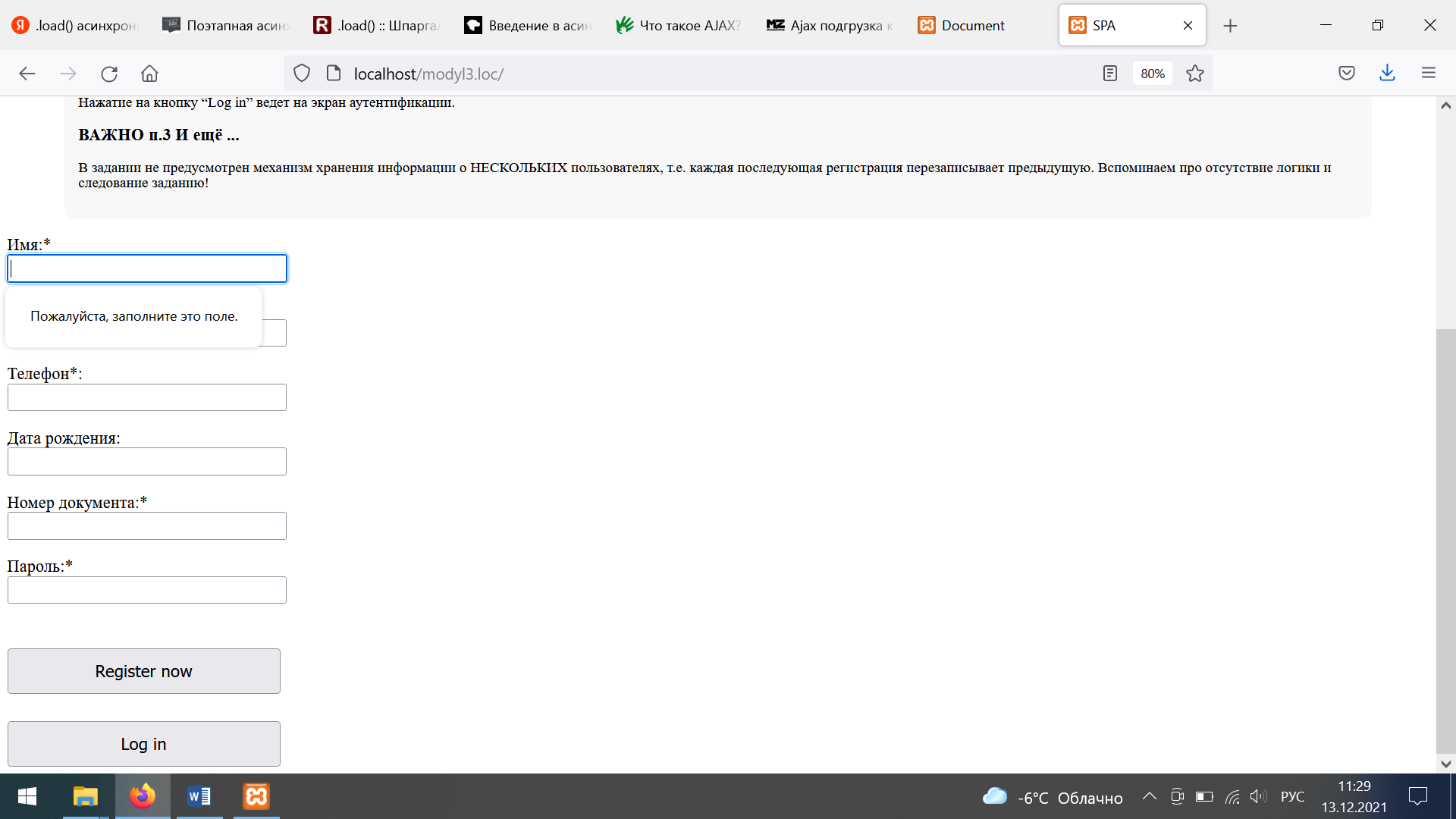
**Экран регистрации**

Неавторизованный пользователь должен иметь возможность зарегистрироваться в системе на экране регистрации.

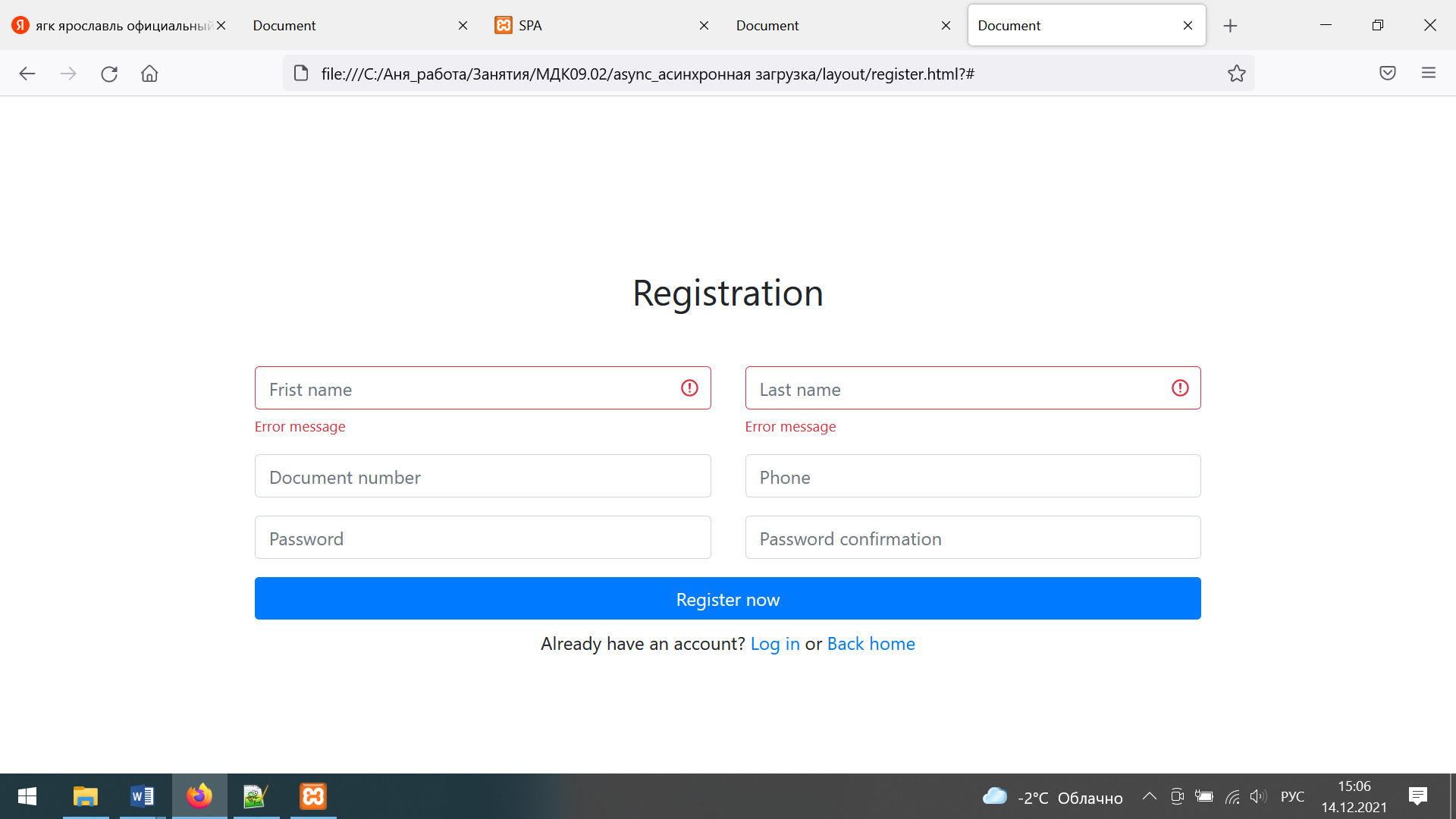
**Проверку полей необходимо реализовать стандартными возможностями HTML5(см. пример в папке)**. **Обратите внимание на подсказки! Проверку совпадения полей реализовать на JS. При успешной регистрации выводим соответствующее сообщение.**

Нажатие на кнопку “Register now”(Зарегистрироваться) регистрирует пользователя. Переход куда-либо на ваше усмотрение.

Нажатие на кнопку “Log in” ведет на экран аутентификации.



**Или**



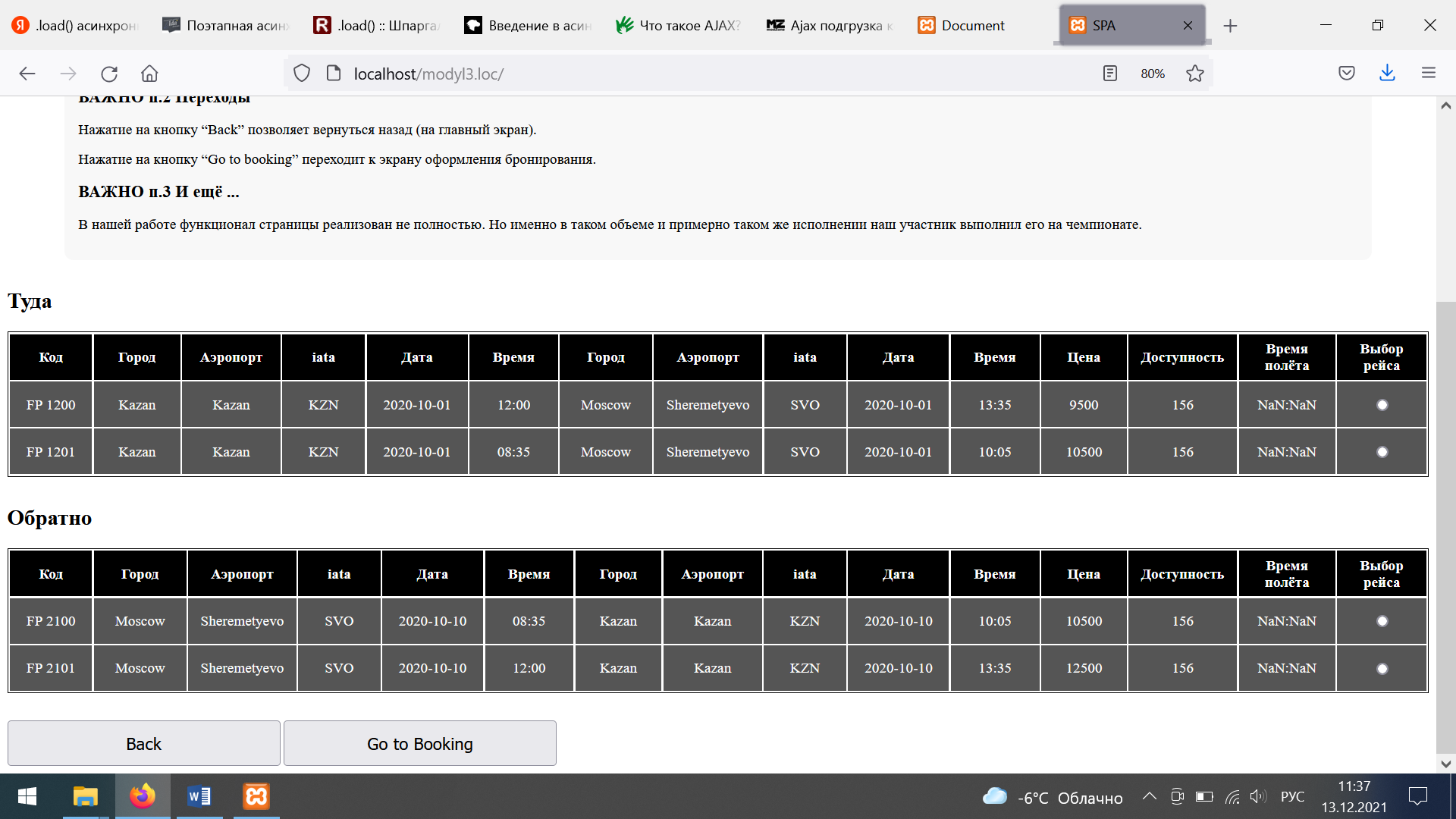
**Экран с найденными рейсами**

Представляем информацию о найденных рейсах. Для это используется табличный вид.

**Пользователь выбирает рейс и переходит на страницу оформления бронирования. Если рейсы не выбраны, то переход не осуществляется. Для выбора рейса используется элемент переключатель.**

Нажатие на кнопку “Back” позволяет вернуться назад (на главный экран).

Нажатие на кнопку “Go to booking” (Забронировать) переходит к экрану оформления бронирования.

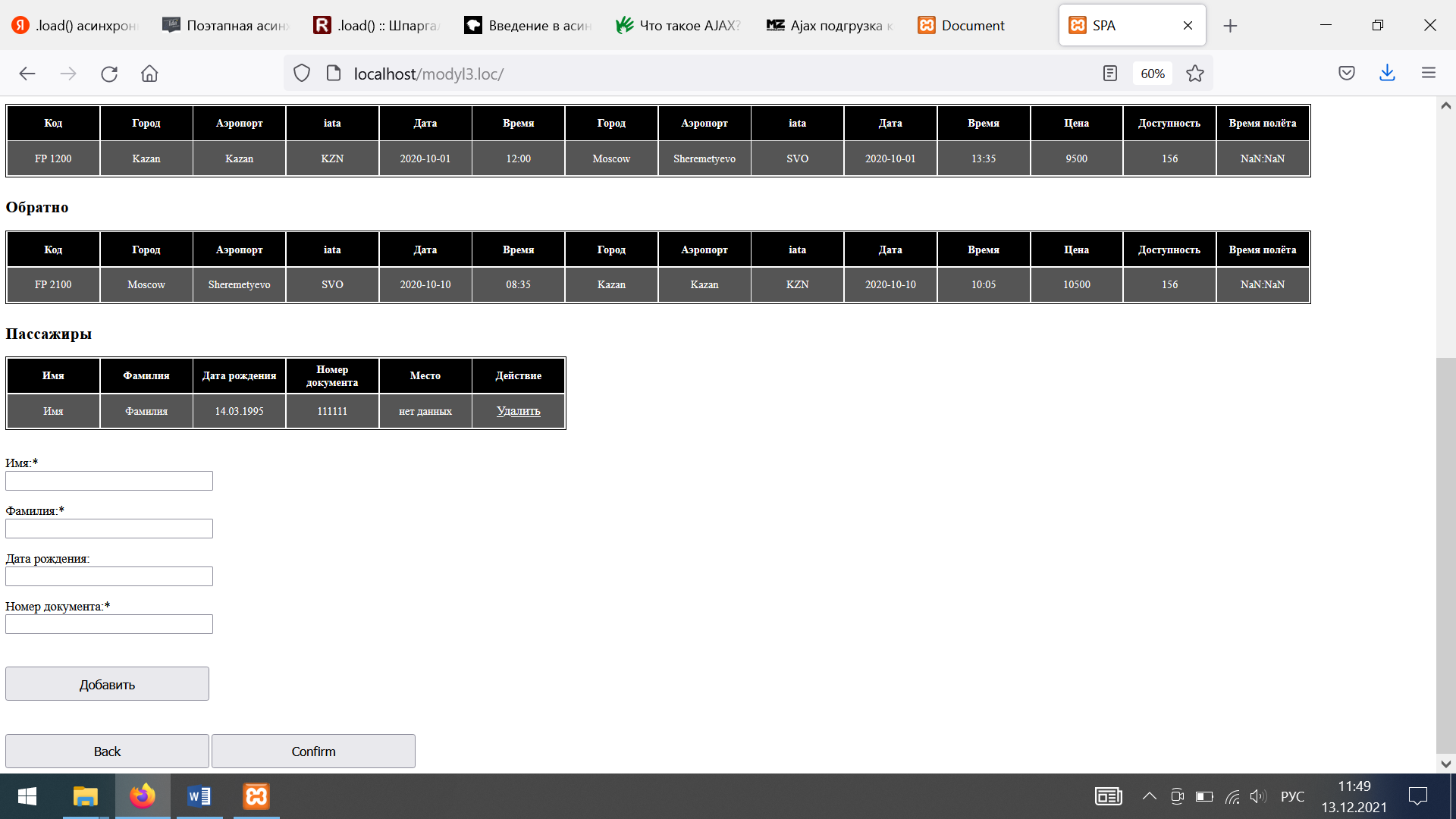


**Экран с оформлением бронирования**

Экран оформления бронирования – на данном экране пользователь должен видеть информацию о выбранных рейсах, а также иметь возможность добавить пассажиров. **Обратите внимание на обязательные поля**

Нажатие на кнопку “Back” позволяет вернуться назад (на экран с найденными рейсами).

Нажатие на кнопку “Confirm” создает бронирование и позволяет перейти на экран управления бронированием.



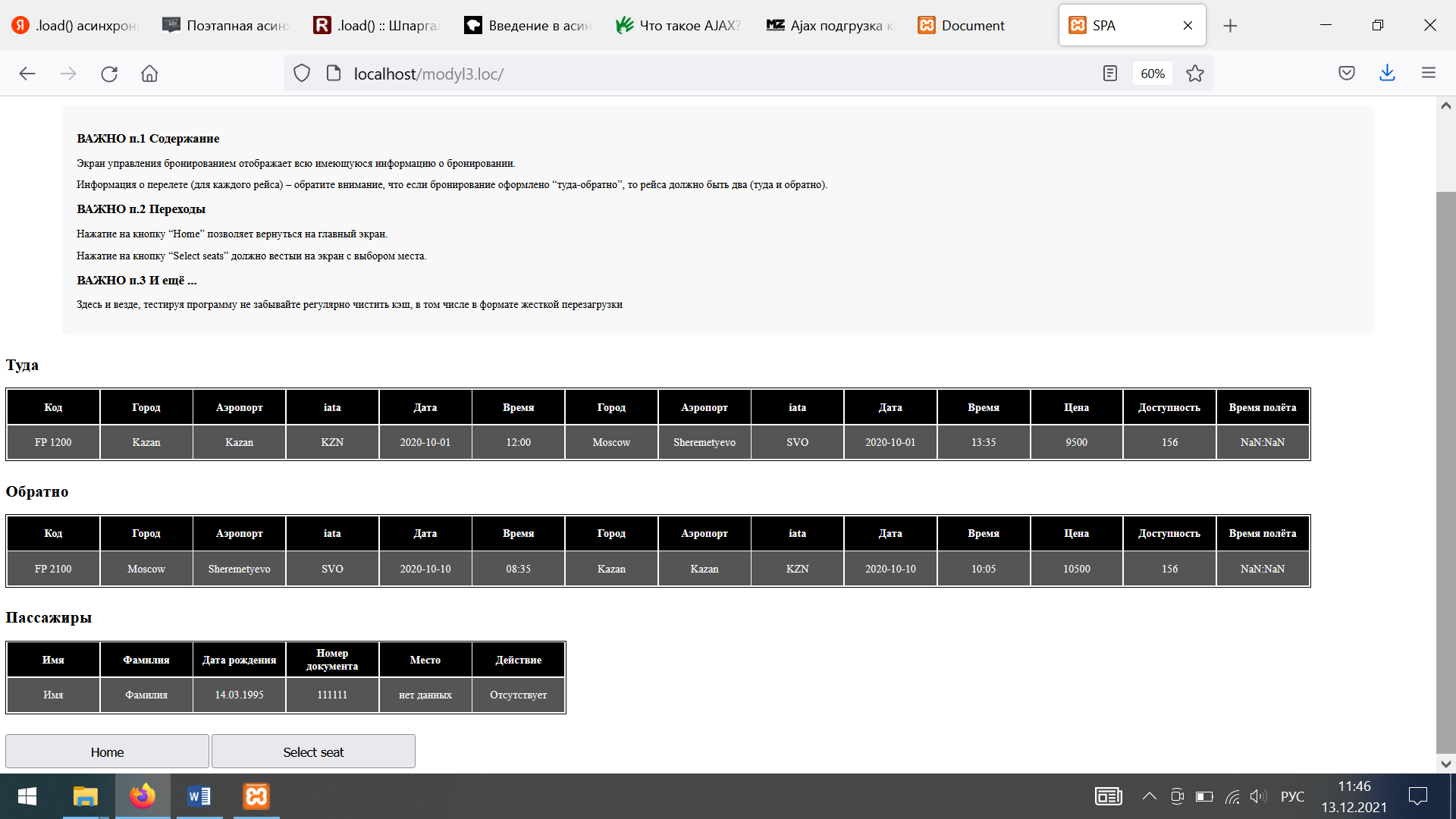
**Экран управления бронированием**

Экран управления бронированием отображает всю имеющуюся информацию о бронировании.

Информация о перелете (для каждого рейса) – обратите внимание, что если бронирование оформлено “туда-обратно”, то рейса должно быть два (туда и обратно).

Нажатие на кнопку “Home” позволяет вернуться на главный экран.

Нажатие на кнопку “Select seats” должно вести на экран с выбором места.



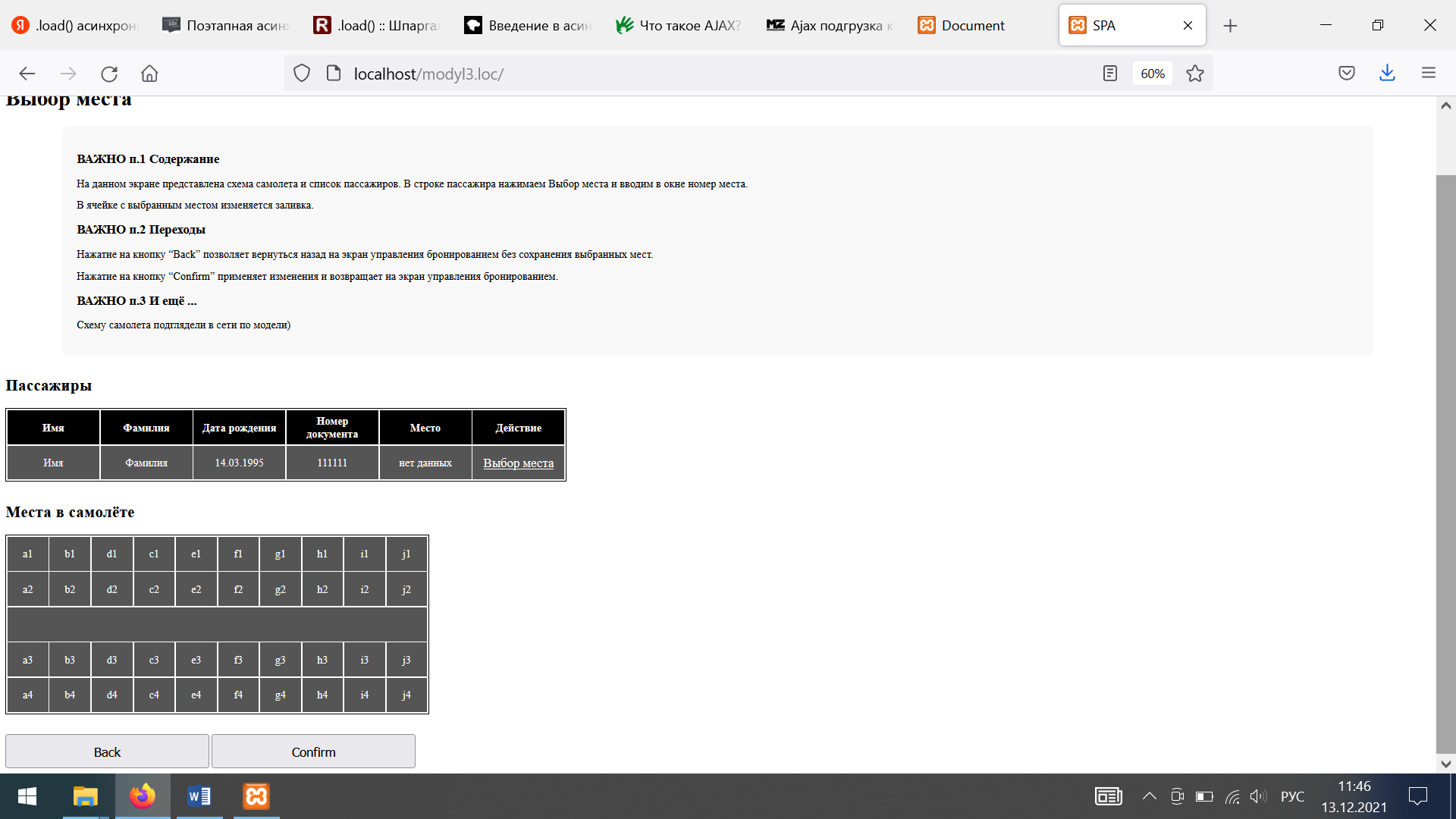
**Экран выбора места**

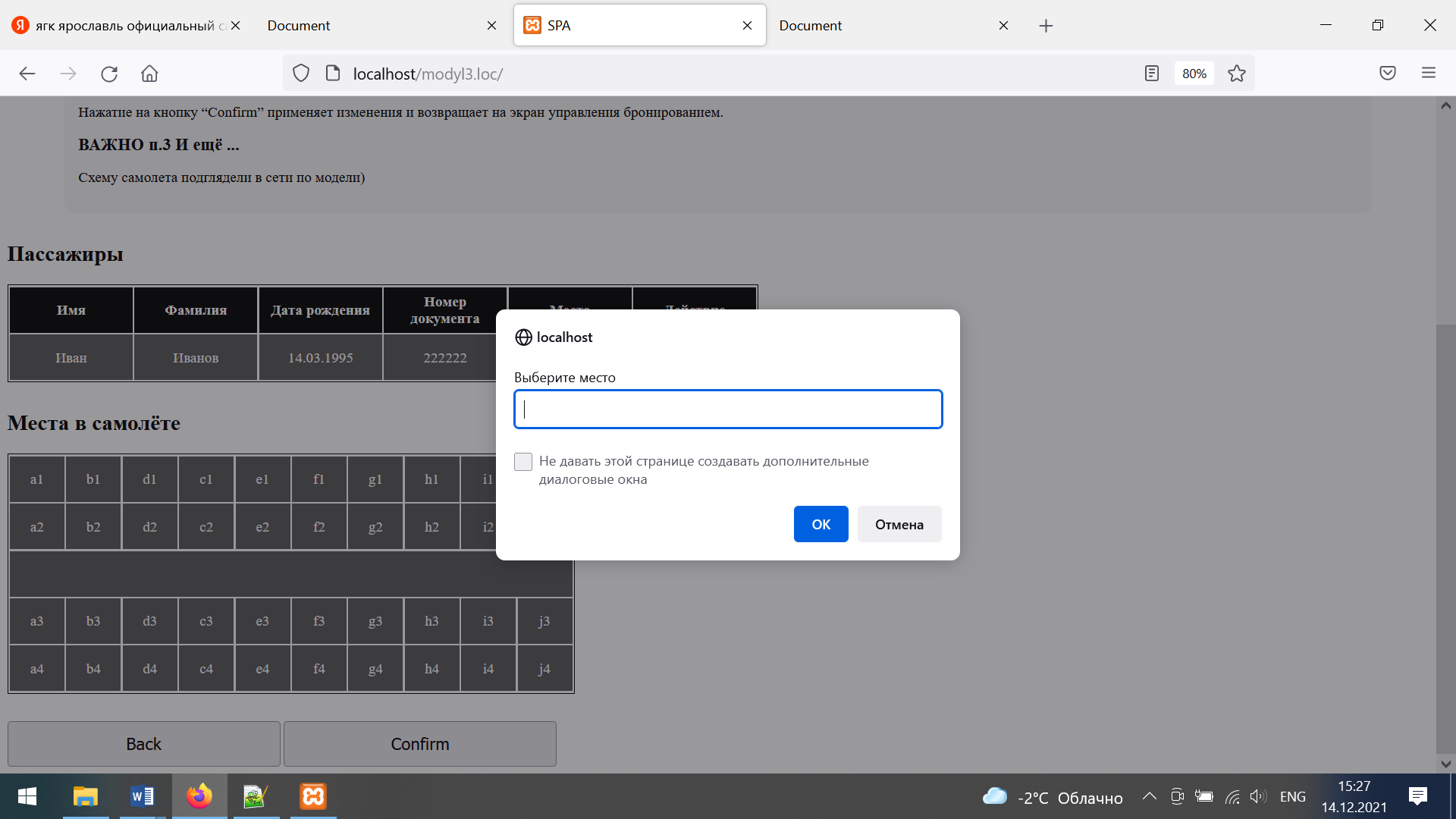
На данном экране представлена схема самолета и список пассажиров. В строке пассажира нажимаем Выбор места и вводим в окне номер места.

В ячейке с выбранным местом изменяется заливка.

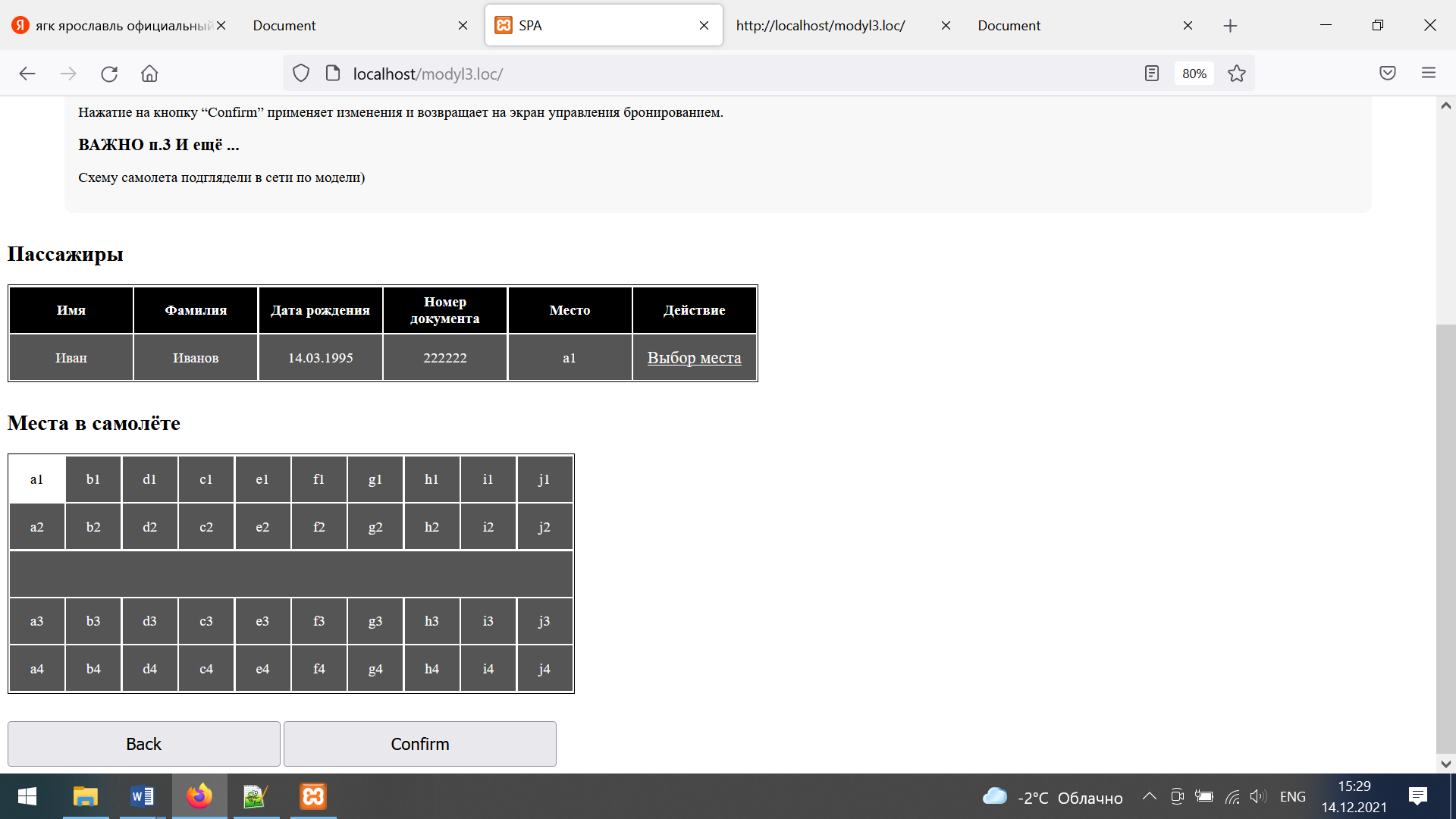
Нажатие на кнопку “Back” позволяет вернуться назад на экран управления бронированием.

Нажатие на кнопку “Confirm” применяет изменения и возвращает на экран управления бронированием.





**\*\*\*страница выбора места в самолете находится в папке с заданием, можно попробовать его использовать.**



**Экран личного кабинета**

На экране с профилем пользователь должен увидеть информацию о себе: ФИО, количество рейсов и информацию, введенную при регистрации.

Кроме этого информацию об избранных, предстоящих и прошедших рейсах.

Нажатие на “Logout” сбрасывает сессию и отправляет на любой экран по вашему усмотрению.

