**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук  
Департамент программной инженерии

|  |  |
| --- | --- |
| СОГЛАСОВАНО  Заместитель декана по учебно-методической работе  доцент департамента больших данных и информационного поиска факультета компьютерных наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ И. Ю. Самоненко «\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2020 г. | УТВЕРЖДАЮ  Академический руководитель образовательной программы «Программная инженерия» профессор департамента программной инженерии, канд. техн. наук  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В. В. Шилов «\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2020 г. |

|  |  |
| --- | --- |
| ***Подп. и дата*** |  |
| ***Инв. № дубл.*** |  |
| ***Взам. инв. №*** |  |
| ***Подп. и дата*** |  |
| ***Инв. № подл*** | RU.17701729.04.01-01 51 01-1-ЛУ |

**WEB ПРИЛОЖЕНИЕ ДЛЯ СОЗДАНИЯ ГЕНЕАЛОГИЧЕСКОГО ДРЕВА**

**Программа и методика испытаний**

**ЛИСТ УТВЕРЖДЕНИЯ**

**RU.17701729.04.01-01 51 01-1-ЛУ**

Исполнитель  
студент группы БПИ173  
\_\_\_\_\_\_\_\_\_\_ / Переплетчиков А. И. /  
«\_\_» \_\_\_\_\_\_\_\_\_\_\_ 2020 г.

**Москва 2020**

УТВЕРЖДЕНRU.17701729.04.01-01 51 01-1-ЛУ

|  |  |
| --- | --- |
| Подп. и дата |  |
| Инв. № дубл. |  |
| Взам. инв. № |  |
| Подп. и дата |  |
| Инв. № подл | RU.17701729.04.01-01 51 01-1-ЛУ |

**WEB ПРИЛОЖЕНИЕ ДЛЯ СОЗДАНИЯ ГЕНЕАЛОГИЧЕСКОГО ДРЕВА**

**Программа и методика испытаний**

**RU.17701729.04.01-01 51 01-1**

**Листов 24**

**Москва 2019**

**Содержание**

[**1. Объект испытаний** 2](#_Toc37547328)

[**1.1. Наименование программы** 2](#_Toc37547329)

[**1.2. Область применения** 3](#_Toc37547330)

[**1.3. Обозначение испытуемой программы** 3](#_Toc37547331)

[**2. Цель испытаний** 4](#_Toc37547332)

[**3. Требования к программе** 5](#_Toc37547333)

[**3.1. Требования к функциональным характеристикам** 5](#_Toc37547334)

[**3.1.1. Требования к составу выполняемых функций** 5](#_Toc37547335)

[**3.1.2. Требования к организации входных данных** 5](#_Toc37547336)

[**3.1.3. Требования к организации выходных данных** 5](#_Toc37547337)

[**3.2. Требования к надежности** 5](#_Toc37547338)

[**3.3. Требования к интерфейсу** 6](#_Toc37547339)

[**4. Требования к программной документации** 7](#_Toc37547340)

[**5. Средства и порядок испытаний** 8](#_Toc37547341)

[**5.1. Технические средства, используемые во время испытаний** 8](#_Toc37547342)

[**5.2. Программные средства, используемые во время испытаний** 8](#_Toc37547343)

[**5.3. Порядок проведения испытаний** 8](#_Toc37547344)

[**5.4. Условия проведения испытаний** 8](#_Toc37547345)

[**5.4.1. Климатические условия** 8](#_Toc37547346)

[**5.4.2. Требования к численности и квалификации персонала** 8](#_Toc37547347)

[**6. Методы испытаний** 9](#_Toc37547348)

[**6.1. Испытание выполнения требований к программной документации** 9](#_Toc37547349)

[**6.2. Испытание выполнения требований к интерфейсу** 9](#_Toc37547350)

[**6.3. Испытание выполнения требований к функциональным характеристикам** 14](#_Toc37547351)

[**6.3.1. Регистрация и авторизация пользователя** 14](#_Toc37547352)

[**6.3.1. Добавление нового человека в дерево** 15](#_Toc37547353)

[**6.3.1. Изменение существующего человека в дереве** 17](#_Toc37547354)

[**6.3.1. Удаление человека из дерева** 19](#_Toc37547355)

[**6.4. Испытание выполнения требований к надежности** 22](#_Toc37547356)

# **1. Объект испытаний**

## **1.1. Наименование программы**

Наименование программы: «Web приложение для создания генеалогического древа».

## **1.2. Область применения**

Программа может применяться в генеалогических исследованиях, а также помочь в популяризации изучения семейной истории. Информация, которую пользователи будут вносить при помощи данного приложения, может послужить ценной для формирования общего единого архива генеалогических баз данных в России и русскоязычных странах.

## **1.3. Обозначение испытуемой программы**

Наименование темы разработки – «Web приложение для создания генеалогического древа».

# **2. Цель испытаний**

Цель испытаний – проверка соответствия функционала и характеристик программного продукта требованиям к программному продукту, изложенным в документе «Техническое задание» (ГОСТ 19.201-78).

# **3. Требования к программе**

## **3.1. Требования к функциональным характеристикам**

### **3.1.1. Требования к составу выполняемых функций**

Программа должна обеспечивать возможность выполнения следующих функций:

1. регистрация новых пользователей, проверка корректности введенных данных;
2. авторизация пользователей, проверка соответствия данных, введенных при авторизации, с данными одного из зарегистрированных пользователей;
3. визуализация генеалогического древа на странице;
4. создание нового родственника, ввод информации о нем;
5. представление информации о каждом родственнике на отдельной странице;
6. указание родственных связей;
7. редактирование информации об уже имеющемся родственнике;
8. удаление родственника из дерева, проверка допустимости удаления (отсутствие разорванных родственных связей);
9. сохранение информации в облачной базе данных Firebase.

### **3.1.2. Требования к организации входных данных**

Ввод данных происходит посредством стандартных html-форм: текст, число, кнопка, переключатели, поле со списком, загрузка файлов.

### **3.1.3. Требования к организации выходных данных**

Программа должна отображать семейное древо в следующем виде:

* Узлы дерева – карточки с информацией о каждом родственнике (фотография, ФИО, место и дата рождения), при клике на карточку открывается страница с расширенной информацией о человеке.
* Ребра дерева – линии, отображающие родственные связи между людьми в узлах; ребра проводятся между двумя супругами и от родителя к ребенку.

Программа должна вносить изменения в БД в соответствии с введенными пользователем данными.

## **3.2. Требования к надежности**

1. Программа обеспечивает проверку корректности входных данных;
2. Программа обеспечивает проверку выходных данных при авторизации пользователя;
3. Программа не допускает доступа к информации других пользователей;
4. Программа должна обеспечивать потокобезопасность при обращении к общим ресурсам;
5. Программа корректно отображает семейное древо, не допускает ошибок или потерь информации при его рендеринге;
6. Программа не должна допускать аварийного завершения при возникающей во время ее работы исключительной ситуации.

## **3.3. Требования к интерфейсу**

Интерфейс программы имеет два уровня доступа: для неавторизованных и авторизованных пользователей. Интерфейс приложения должен включать в себя следующие элементы взаимодействия:

1. Страница авторизации пользователя (страница, отображающаяся для неавторизированного пользователя по умолчанию) с полями для ввода электронной почты и пароля и ссылкой на страницу регистрации;
2. Страница регистрации нового пользователя со следующими полями:
   1. Электронная почта;
   2. Пароль;
   3. Подтверждение пароля;
   4. Фамилия;
   5. Имя;
   6. Отчество;
   7. Дата рождения;
   8. Пол;
   9. Родной город;
3. Страница с краткой информацией о сайте;
4. Главная страница с генеалогическим деревом. Перемещаться по генеалогическому дереву можно, зажимая курсор и перемещая его в нужную сторону, как на интернет-картах;
5. Боковое меню, раскрывающееся при наведении курсора. Элементы меню:
   1. Добавить человека
   2. Выход
   3. Информация
6. Страница добавления нового человека, содержащая те же поля, что и при регистрации нового пользователя (кроме электронной почты и пароля), а также выбор фото и текстовая область для ввода биографии человека;
7. Страница выбора ближайшего родственника для нового созданного человека и типа родственной связи между ними (родитель/супруг/ребенок);
8. Страница с более подробной информацией о человеке, открывающаяся при клике на карточку человека в дереве;

# **4. Требования к программной документации**

Состав программной документации должен включать в себя следующие компоненты:

1. Техническое задание (ГОСТ 19.201-78)
2. Программа и методика испытаний (ГОСТ 19.301-78)
3. Пояснительная записка (ГОСТ 19.404-79)
4. Руководство оператора (ГОСТ 19.505-79)
5. Текст программы (ГОСТ 19.401-78)

# **5. Средства и порядок испытаний**

## **5.1. Технические средства, используемые во время испытаний**

Для нормального функционирования программы требуется компьютер, оснащенный следующими техническими компонентами:

1. процессор не ниже Intel Pentium/Celeron, AMD K6/Athlon/Duron или совместимый с ними с тактовой частотой не ниже 1 ГГц;
2. 512 Мб ОЗУ или более;
3. жесткий диск с объемом свободной памяти не менее 100 Мб;
4. VGA-совместимые видеоадаптер и монитор с разрешением не ниже 1280х800;
5. клавиатура и мышь.

## **5.2. Программные средства, используемые во время испытаний**

Для нормального функционирования программы требуется компьютер, оснащенный следующими программными компонентами:

1. операционная система Microsoft Windows XP (SP2, SP3) / Vista / 7 / 8 / 8.1 / 10;
2. библиотека Microsoft .NET Framework 4.5 и выше.

## **5.3. Порядок проведения испытаний**

Испытания должны проводиться в следующем порядке:

1. проверка требований к программной документации;
2. проверка требований к интерфейсу;
3. проверка требований к функциональным характеристикам;
4. проверка требований к надежности.

## **5.4. Условия проведения испытаний**

### **5.4.1. Климатические условия**

Климатические условия проведения испытаний программного продукта должны удовлетворять стандартным требованиям к климатическим условиям использования компьютера и использования и хранения соответствующих электронных и бумажных носителей информации.

### **5.4.2. Требования к численности и квалификации персонала**

Для испытаний программы требуется по крайней мере один пользователь.

Требуемая квалификация пользователя программы – оператор ЭВМ с базовыми знаниями в области дифференциальных уравнений и нейронных сетей.

# **6. Методы испытаний**

Испытания проводятся в порядке, указанном в п. 5.3 настоящего документа.

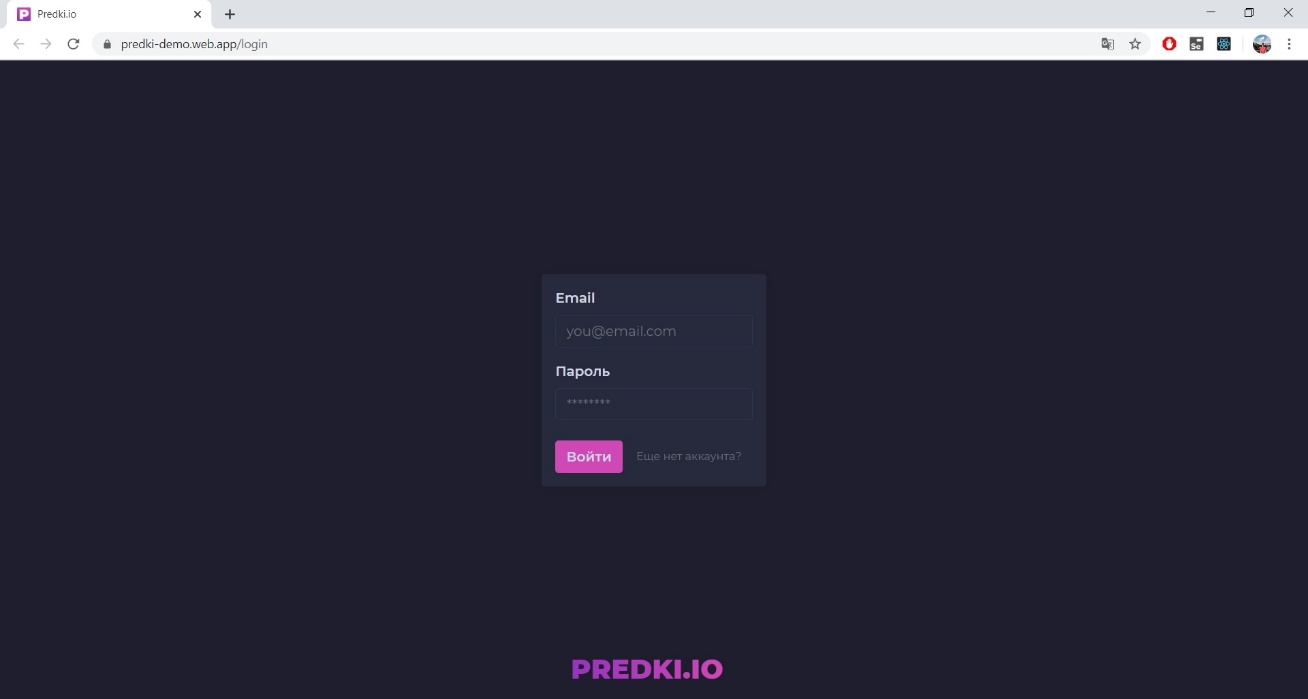
## **6.1. Испытание выполнения требований к программной документации**

Соответствие программной документации требованиям проверяется путем просмотра программной документации вручную.

Путем просмотра выявлено, что программная документация удовлетворяет требованиям.

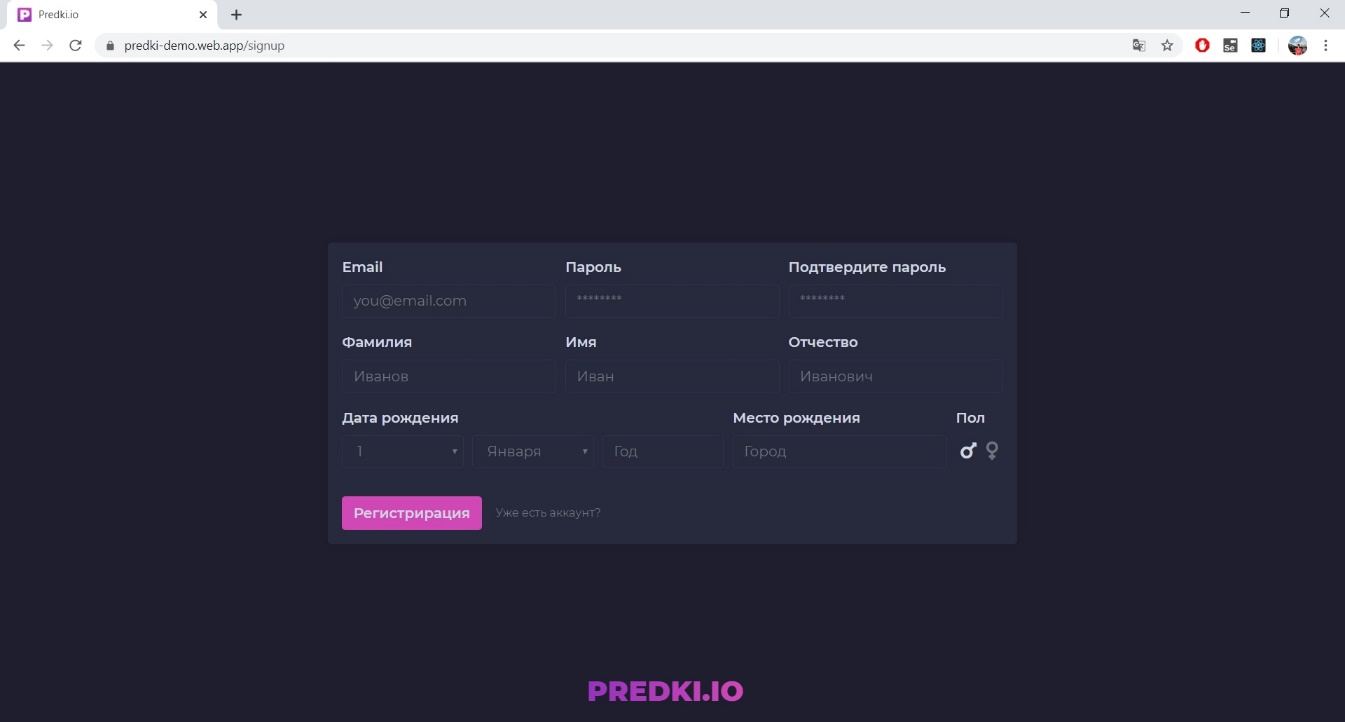
## **6.2. Испытание выполнения требований к интерфейсу**

Страница авторизации пользователя отображается корректно



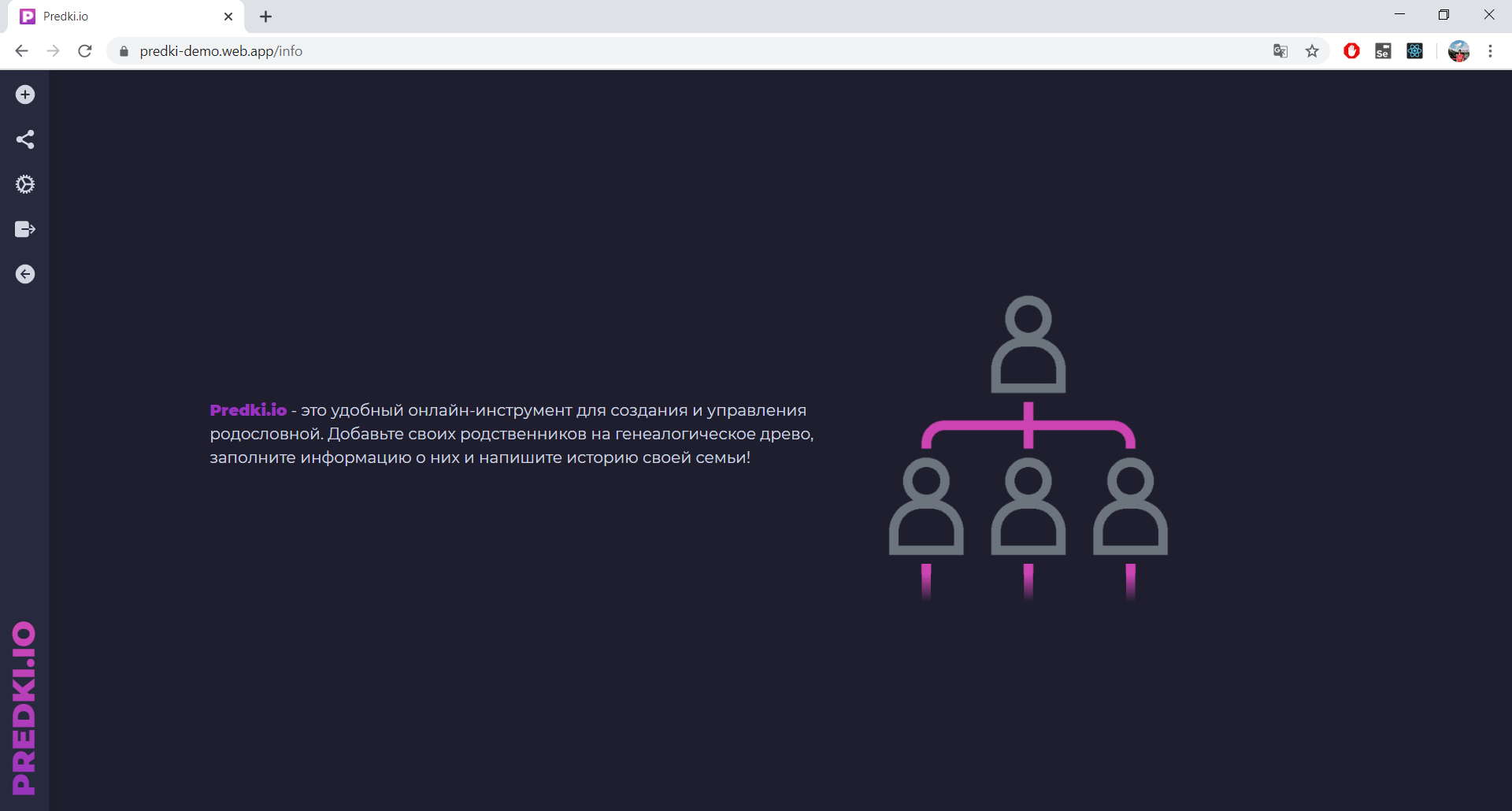
*Рис. 1. Внешний вид страницы авторизации*

Страница регистрации пользователя отображается корректно



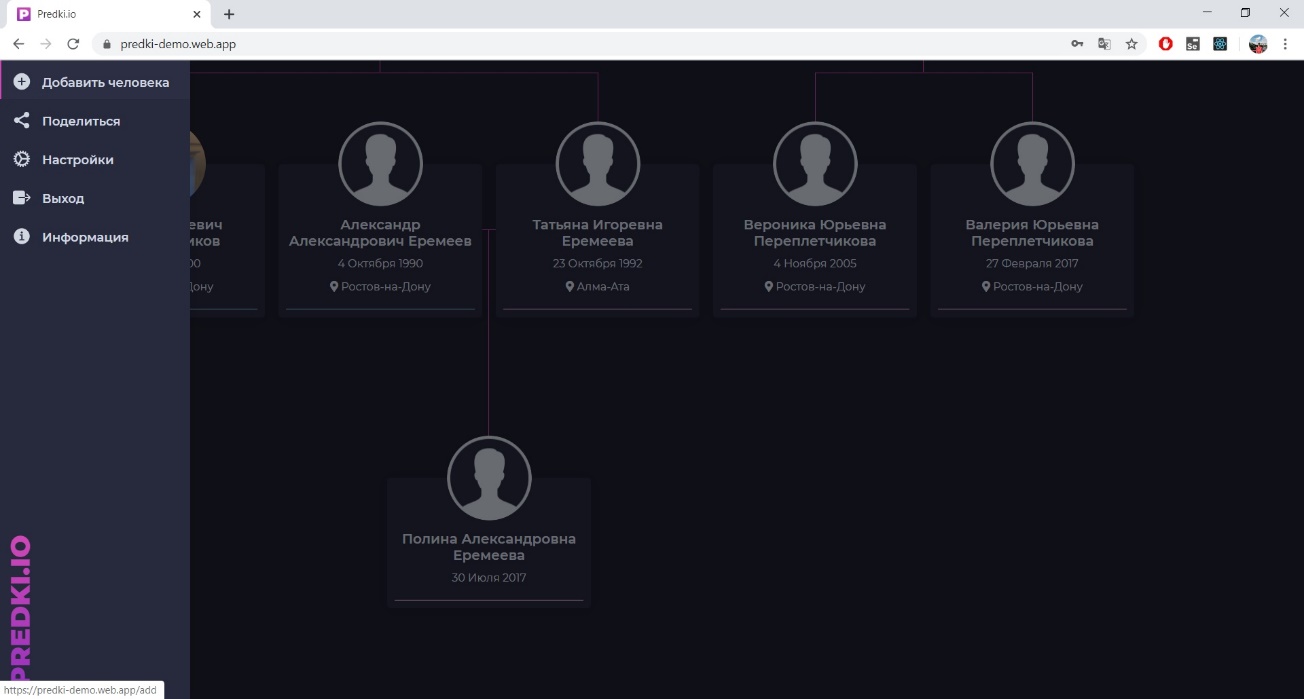
*Рис. 2. Внешний вид страницы регистрации*

Страница с краткой информацией о сайте отображается корректно



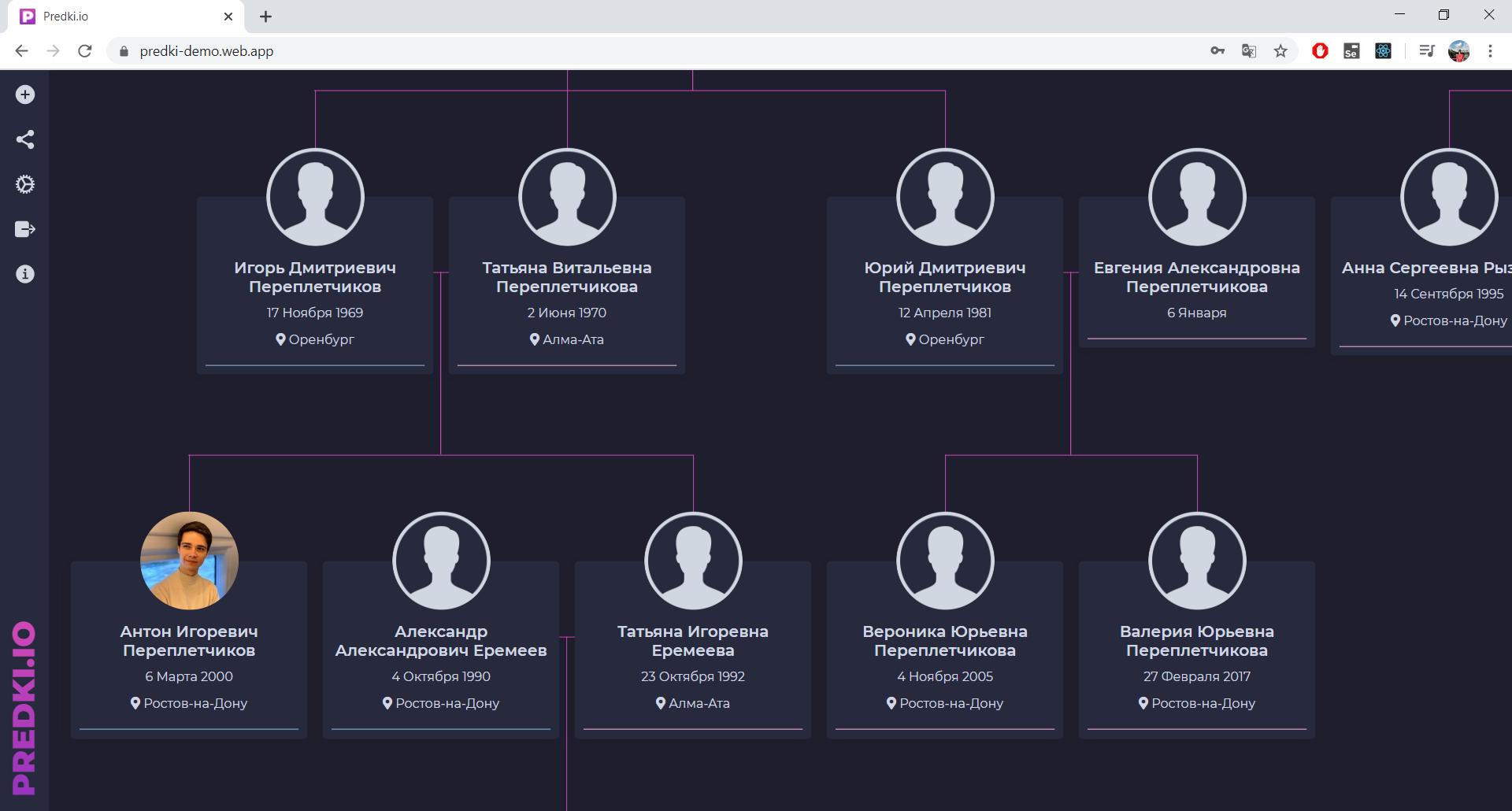
*Рис. 3. Внешний вид информационной страницы*

Боковое меню отображается корректно и раскрывается при наведении на него курсора



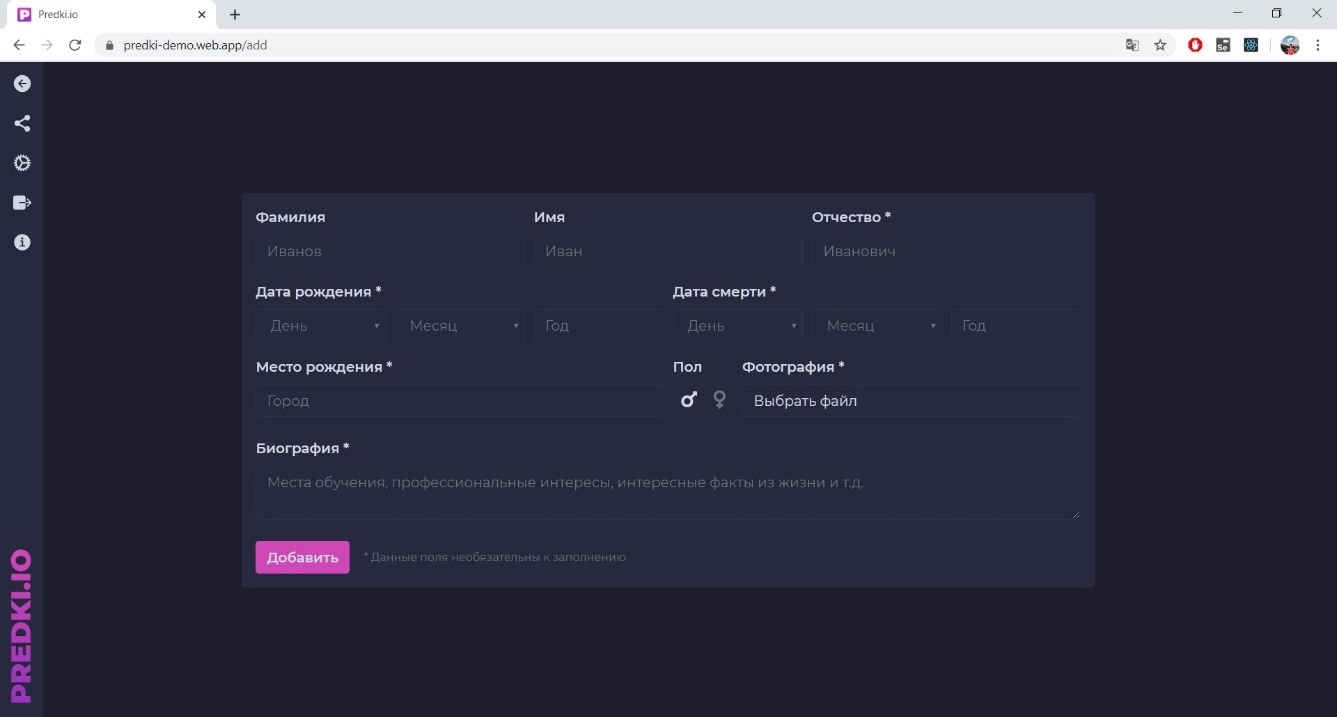
*Рис. 4. Внешний вид бокового меню при наведении на него курсора*

Страница дерева корректно отображается в полном соответствии с предъявляемыми требованиями.



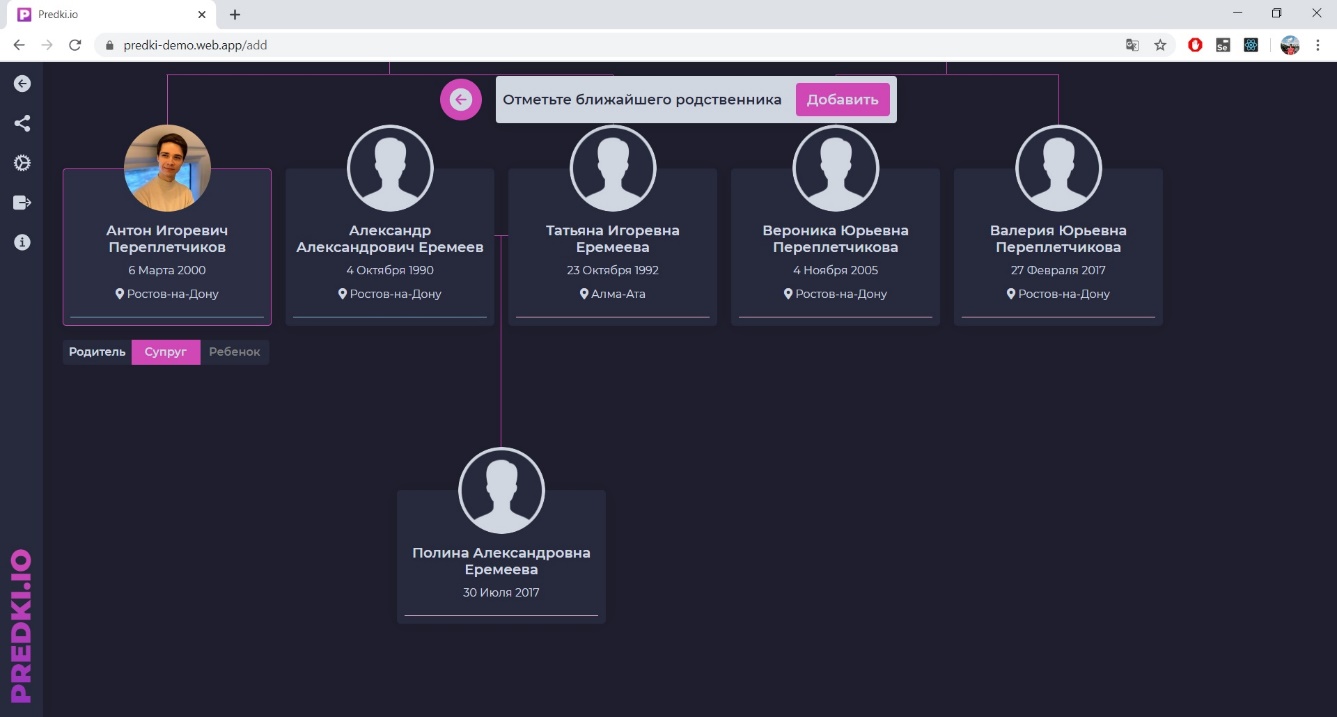
*Рис. 5. Главная страница с деревом*

Страница добавления нового человека отображается корректно



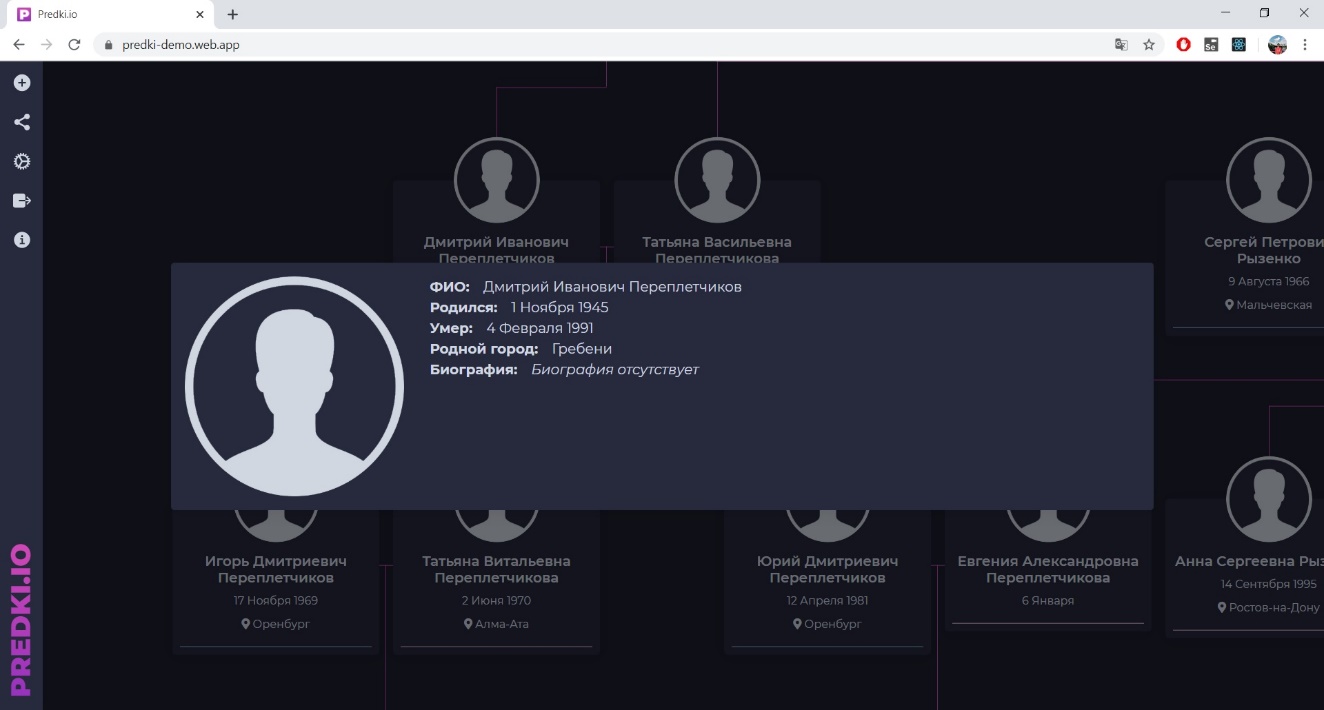
*Рис. 6. Внешний вид страницы добавления нового человека*

Страница выбора родственных связей отображается корректно



*Рис. 7. Внешний вид страницы выбора родственных связей для добавляемого человека*

Персональная страница родственника отображается корректно



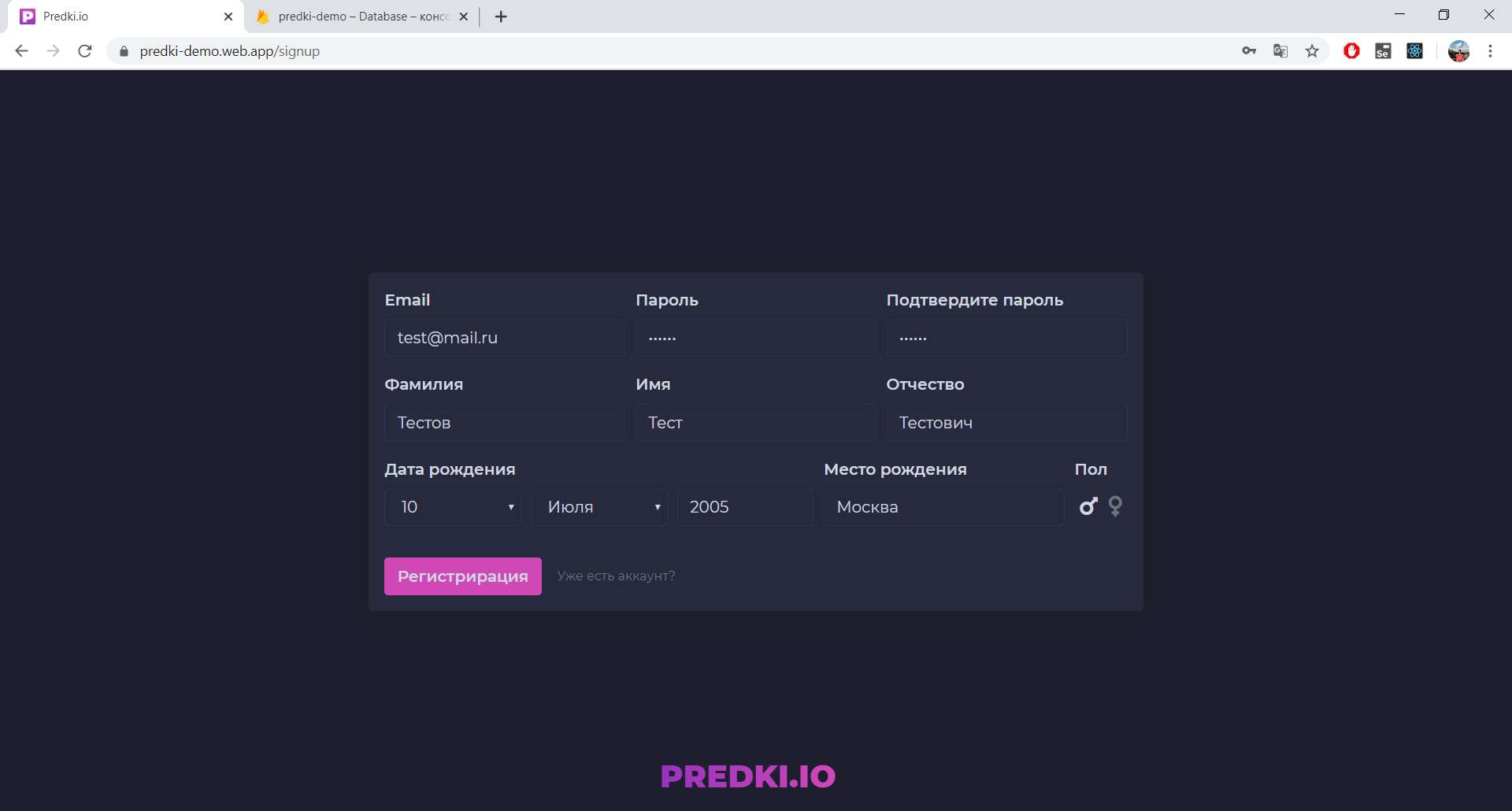
*Рис. 8. Внешний вид страницы человека*

Таким образом, все компоненты интерфейса программы визуально соответствуют предъявляемым требованиям, следовательно, программа соответствует требованиям к интерфейсу.

## **6.3. Испытание выполнения требований к функциональным характеристикам**

### **6.3.1. Регистрация и авторизация пользователя**

Зарегистрируем нового пользователя, введя информацию о нем в форму регистрации.



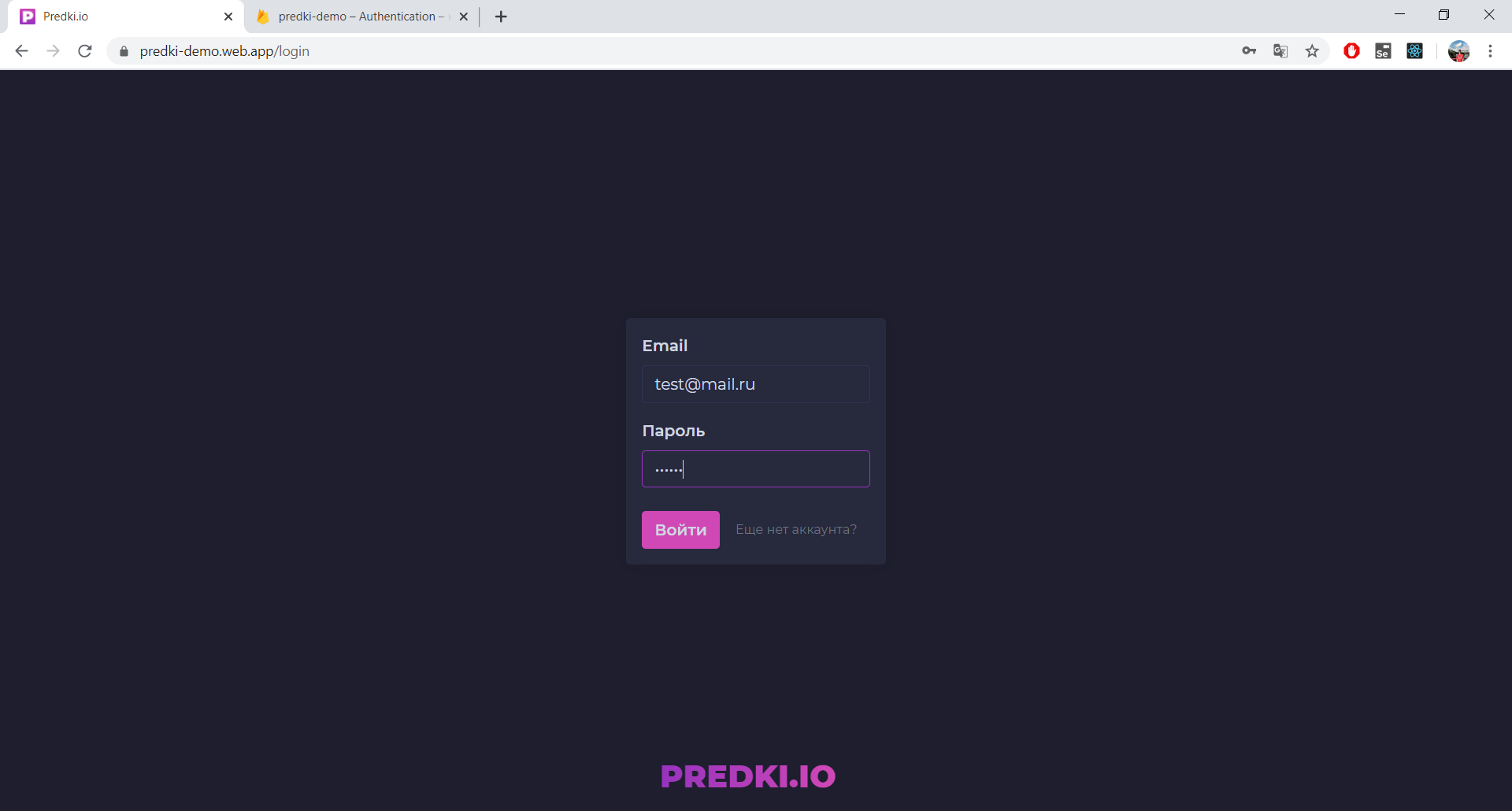
*Рис. 9. Заполнение формы регистрации*

После завершения регистрации в консоли Firebase должен отобразиться новый зарегистрированный пользователь.

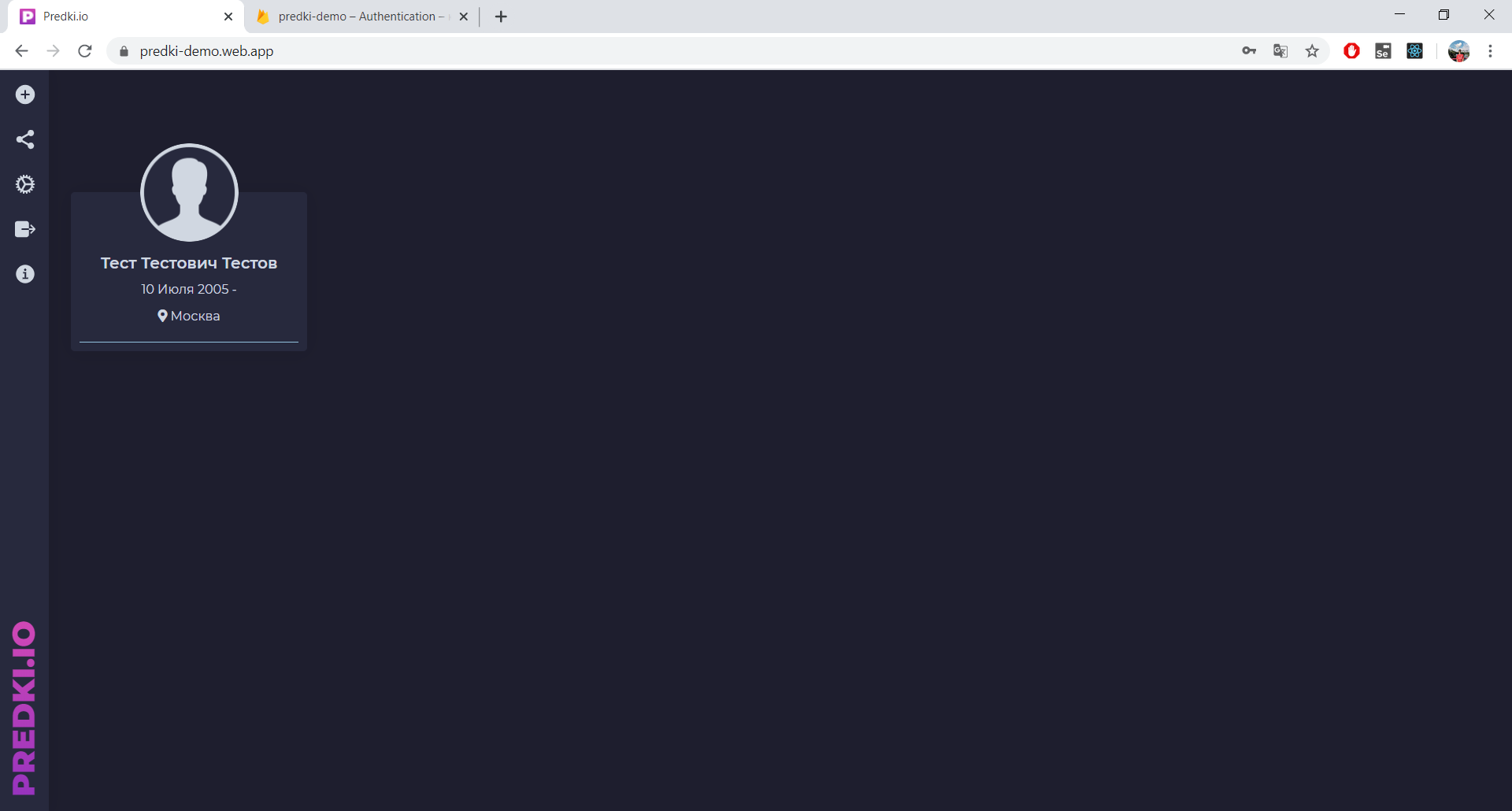


*Рис. 10. Новый пользователь добавлен в БД*

С введенными данными новый пользователь должен иметь возможность войти в систему.



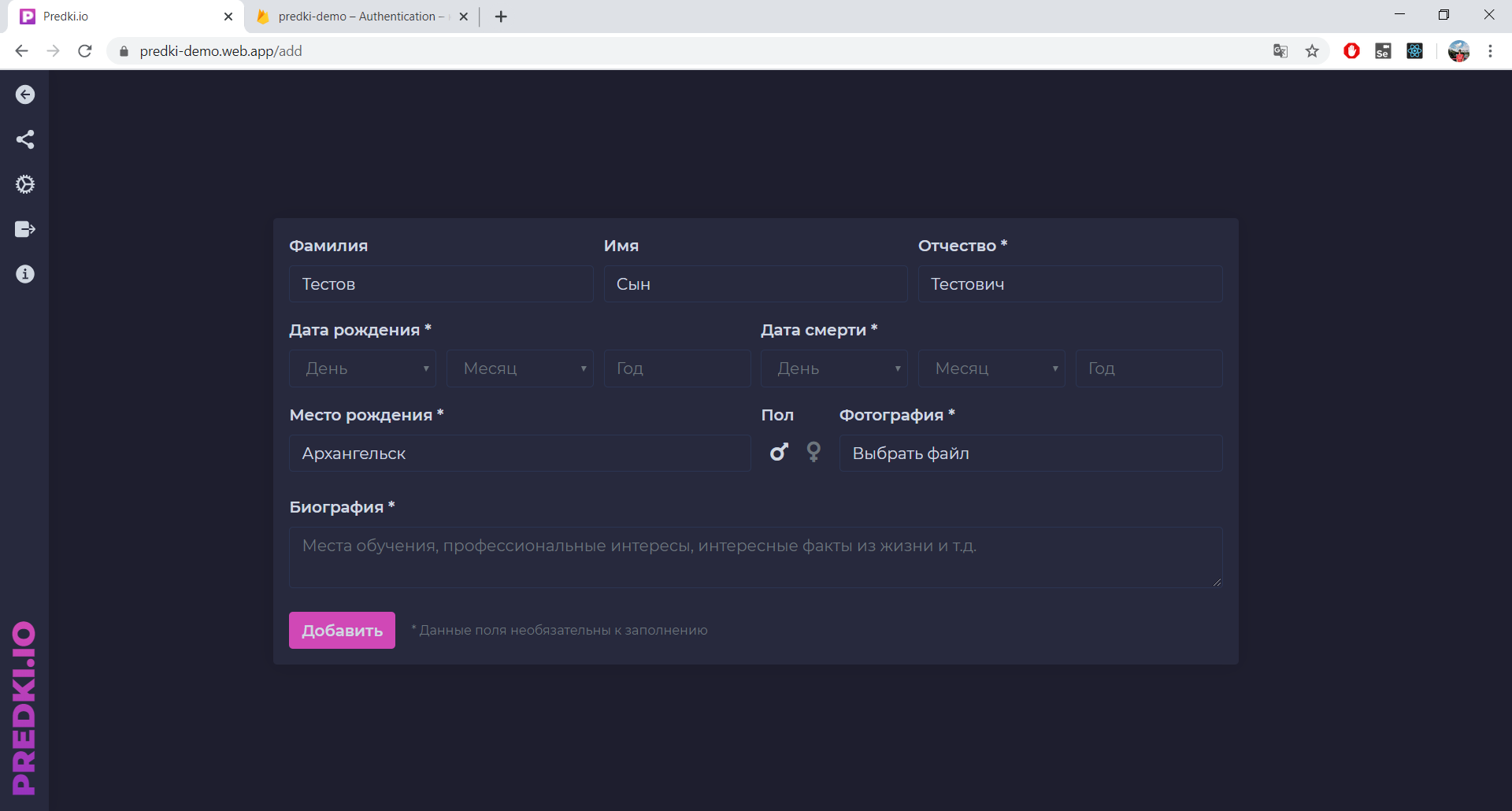
*Рис. 11. Введенные данные подходят для входа в систему*



*Рис. 12. После авторизации отобразилось дерево с одним человеком - пользователем*

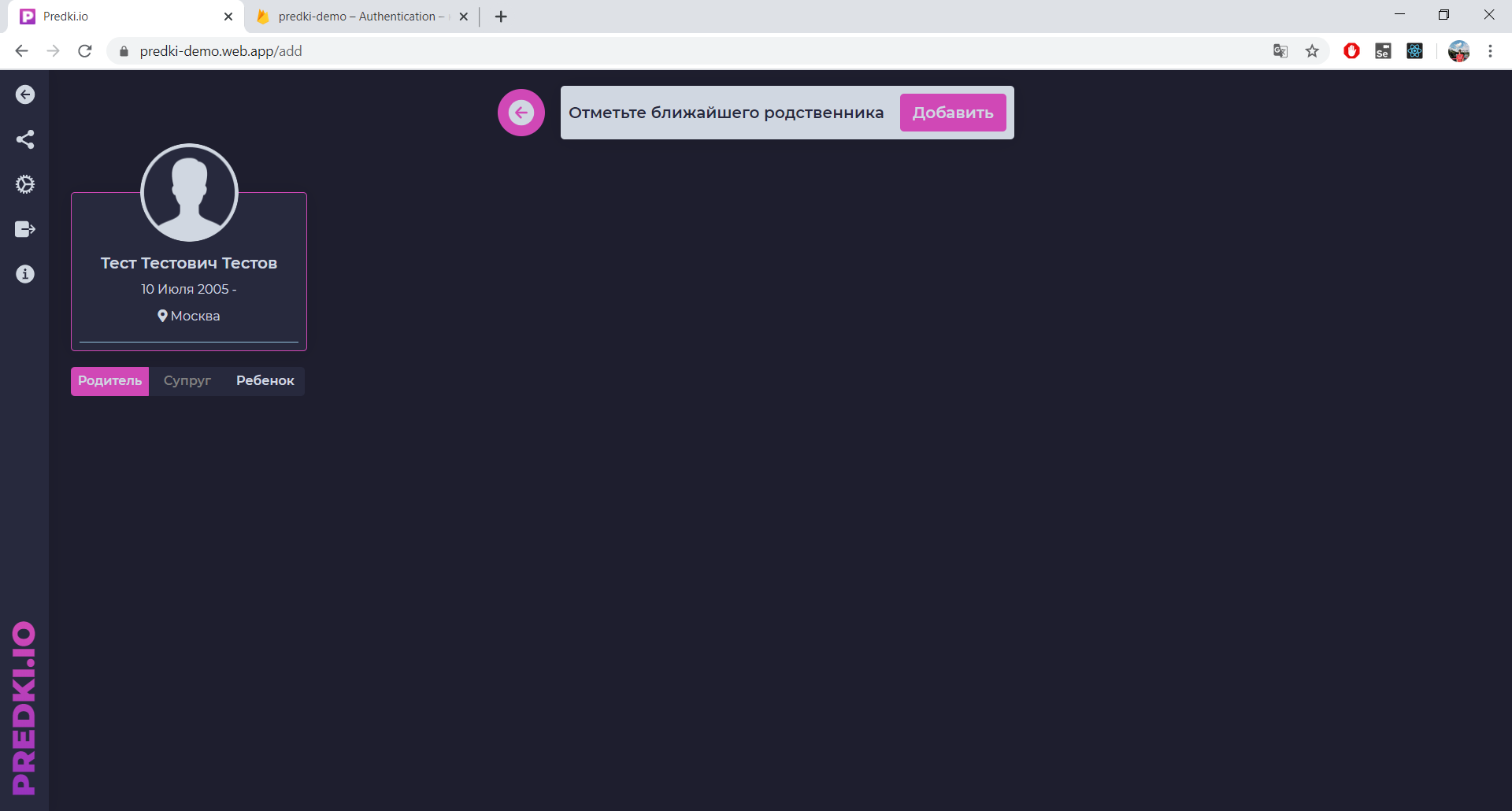
### **6.3.1. Добавление нового человека в дерево**

При нажатии на кнопку «Добавить человека» открывается форма для ввода данных о новом человеке.



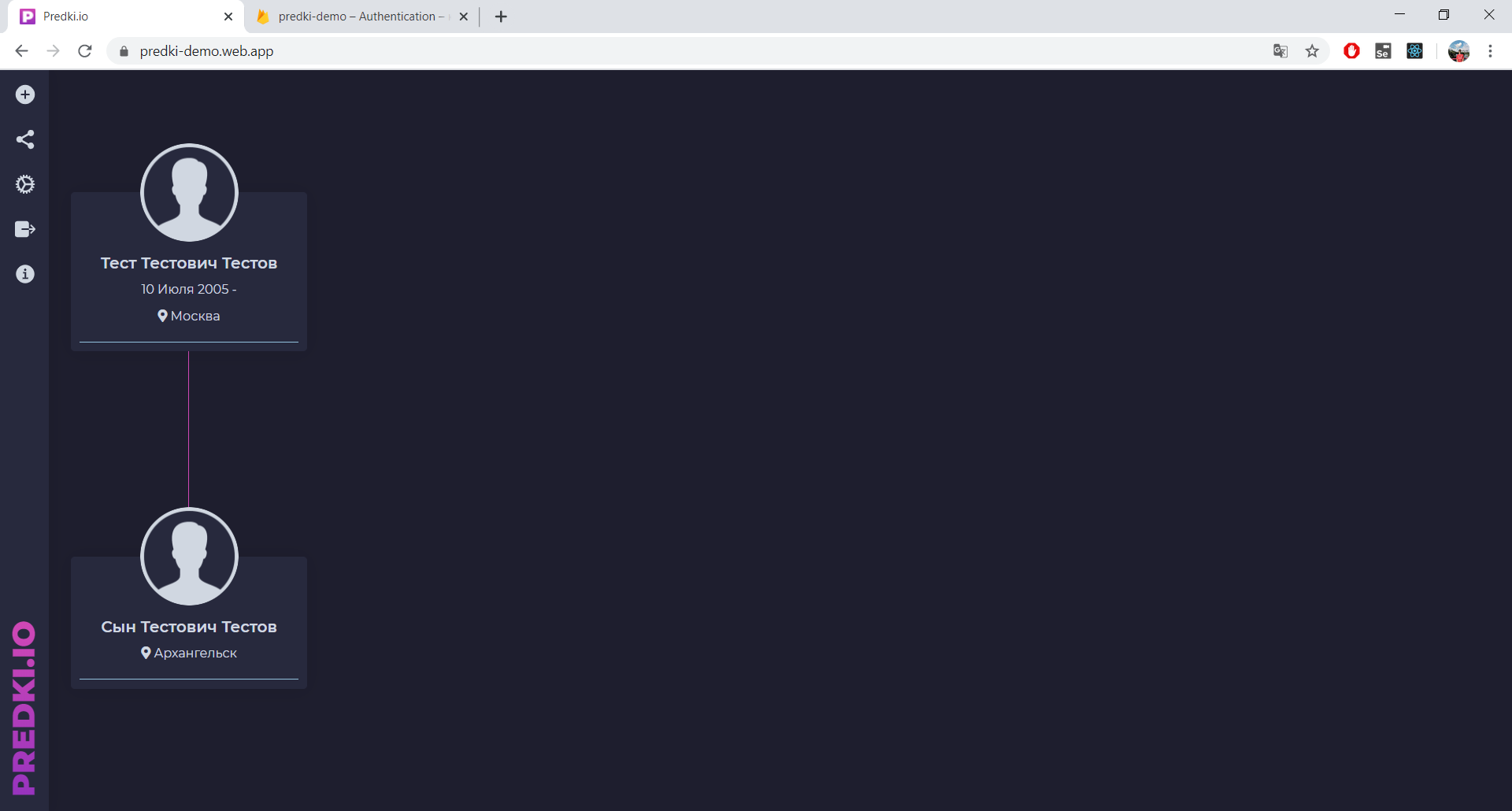
*Рис. 13. Ввод данных о новом человеке*

После нажатия на кнопку «Добавить» открывается дерево, где можно выбрать родственника и тип родственной связи.



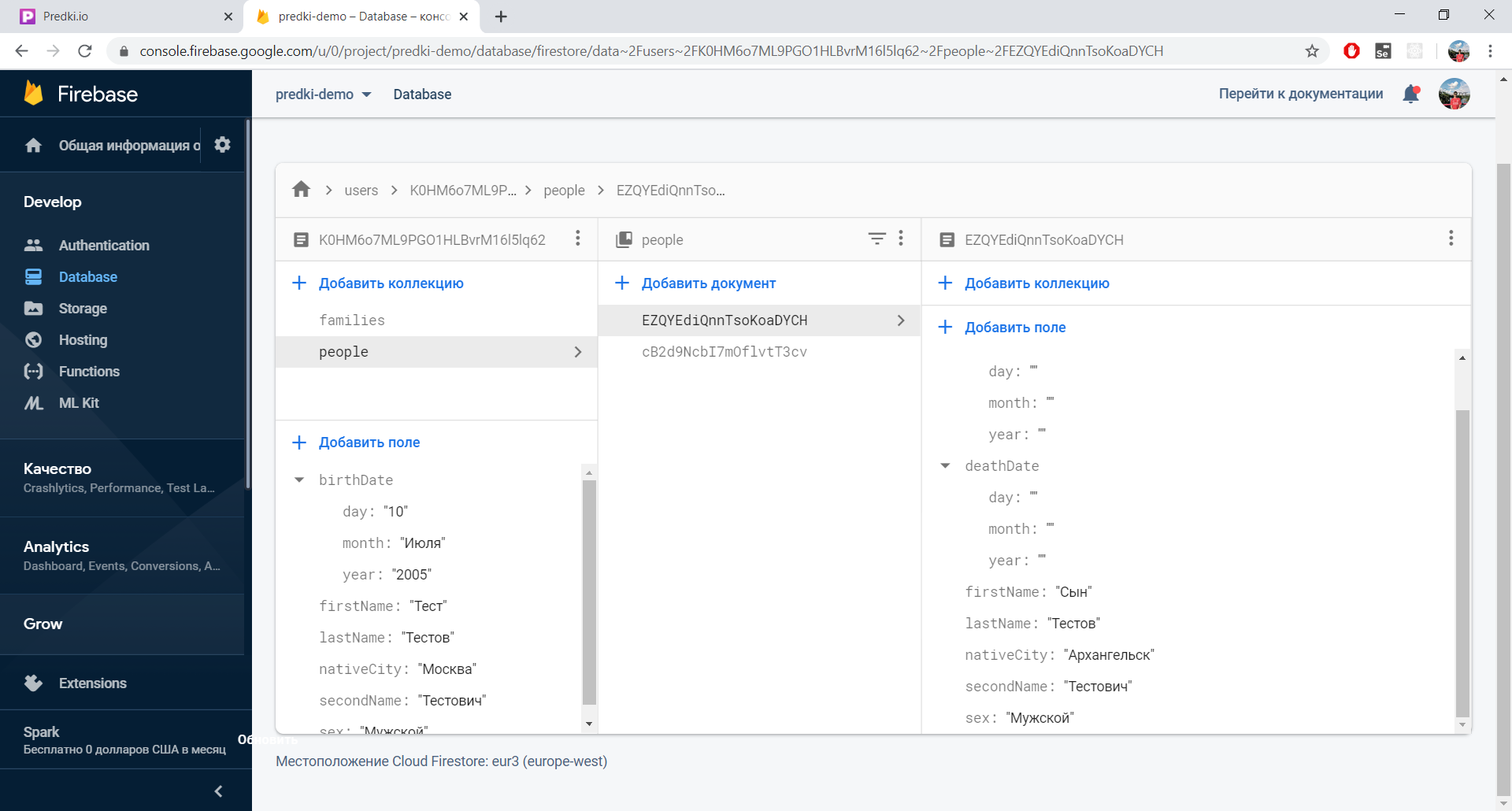
*Рис. 14. Выбор родственника*

После нажатия на кнопку «Добавить» человек корректно добавляется в дерево.



*Рис. 15. Результат добавления*

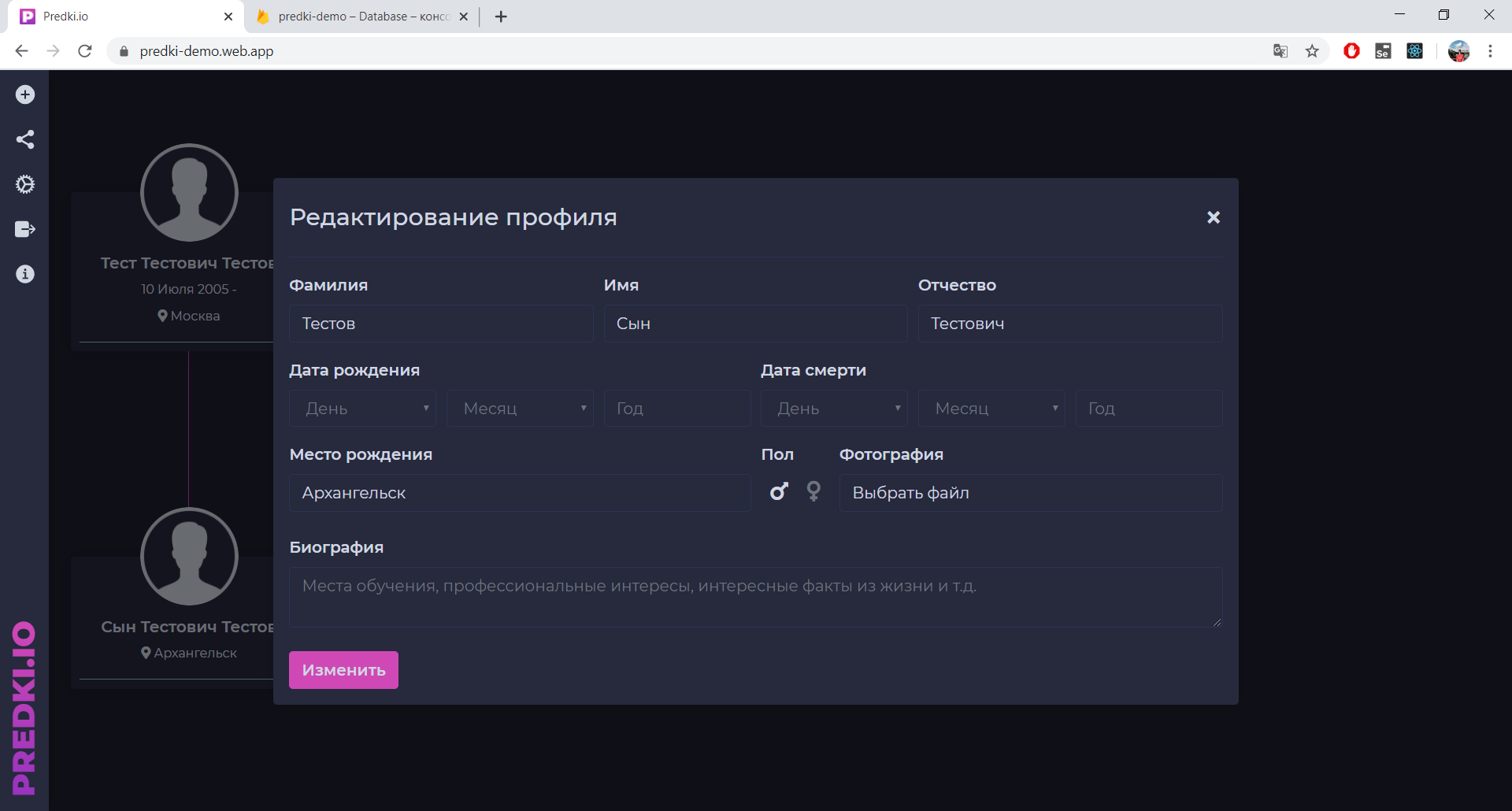
В базе данных также добавляется новый человек и обновляются родственные связи.



*Рис. 16. Обновление в БД*

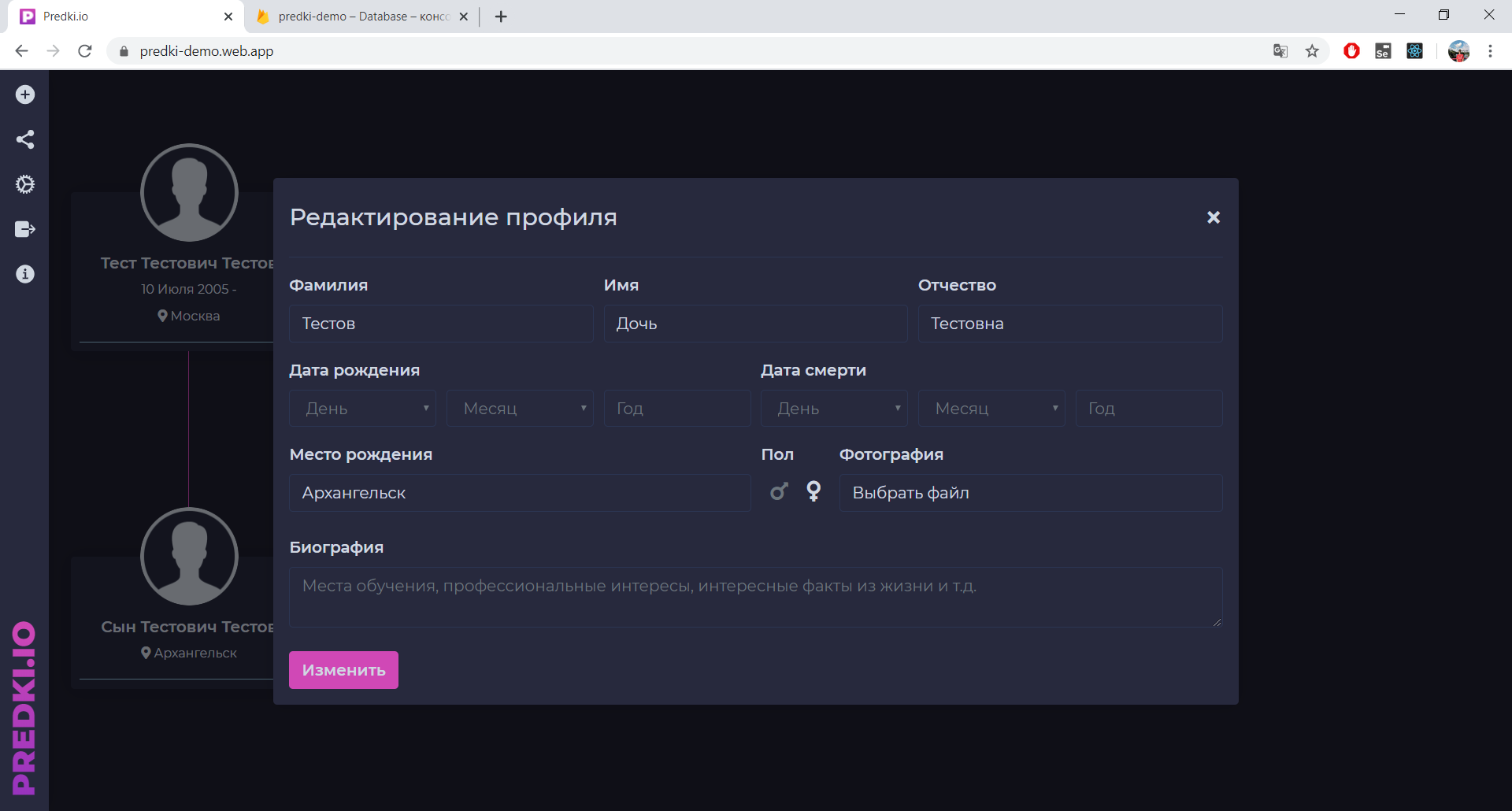
### **6.3.1. Изменение существующего человека в дереве**

При наведении на фотографию человека на его карточке (или пиктограмму, если фотографии нет), фотография приподнимается и под ней оказываются две кнопки: отредактировать (иконка карандаша) и удалить (иконка мусорной корзины). При нажатии на кнопку «отредактировать» открывается окно редактирования, содержащее формы, как в окне добавления, только с уже заполненные данными для данного человека.



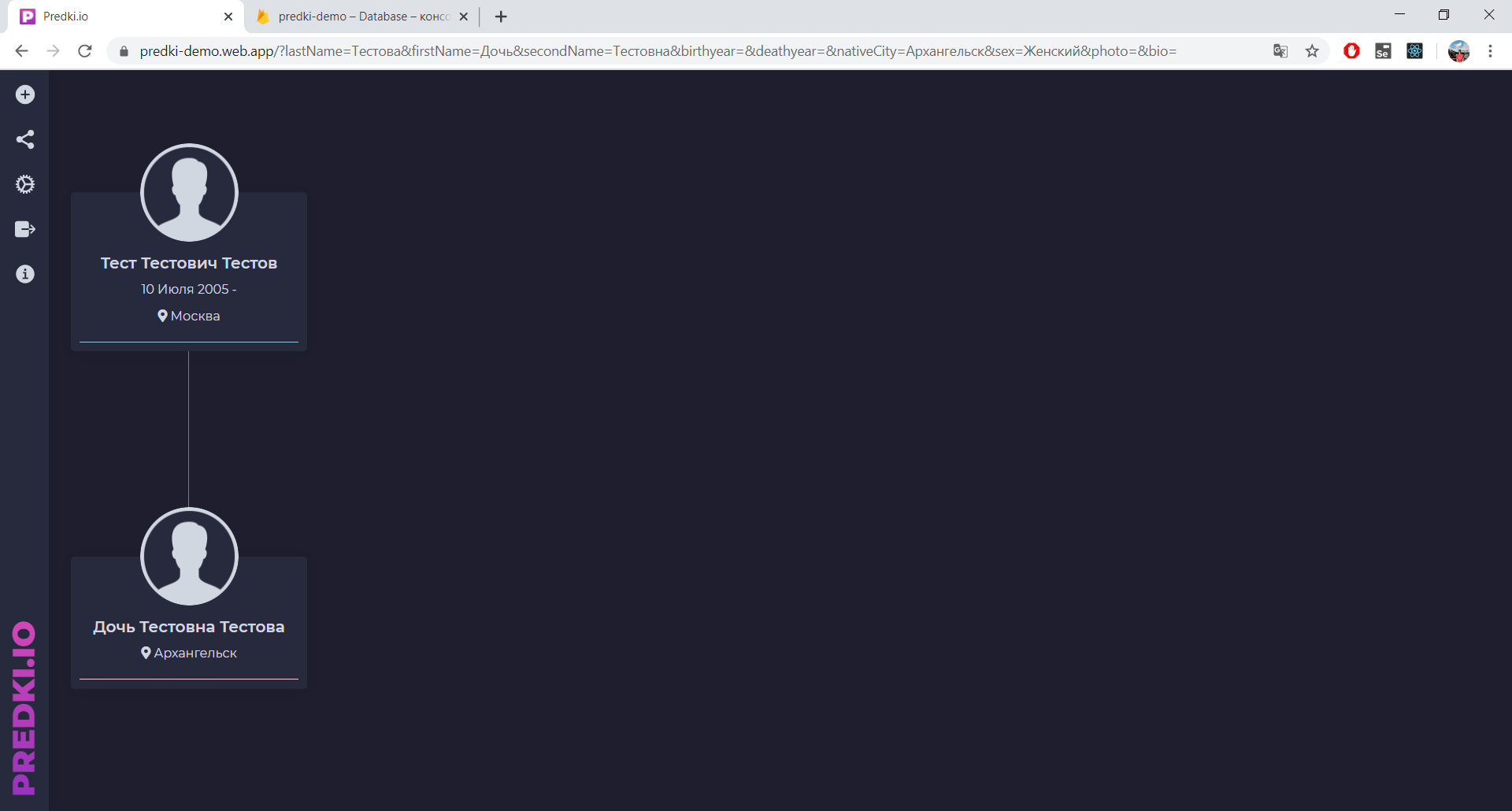
*Рис. 17. Окно редактирования человека*

Значения в формах можно изменять. Если пользователь хочет вернуться к дереву без сохранения изменений, он нажимает на крестик в правой верхней части окна редактирования. Если пользователь хочет сохранить результаты, он нажимает на кнопку «Изменить».

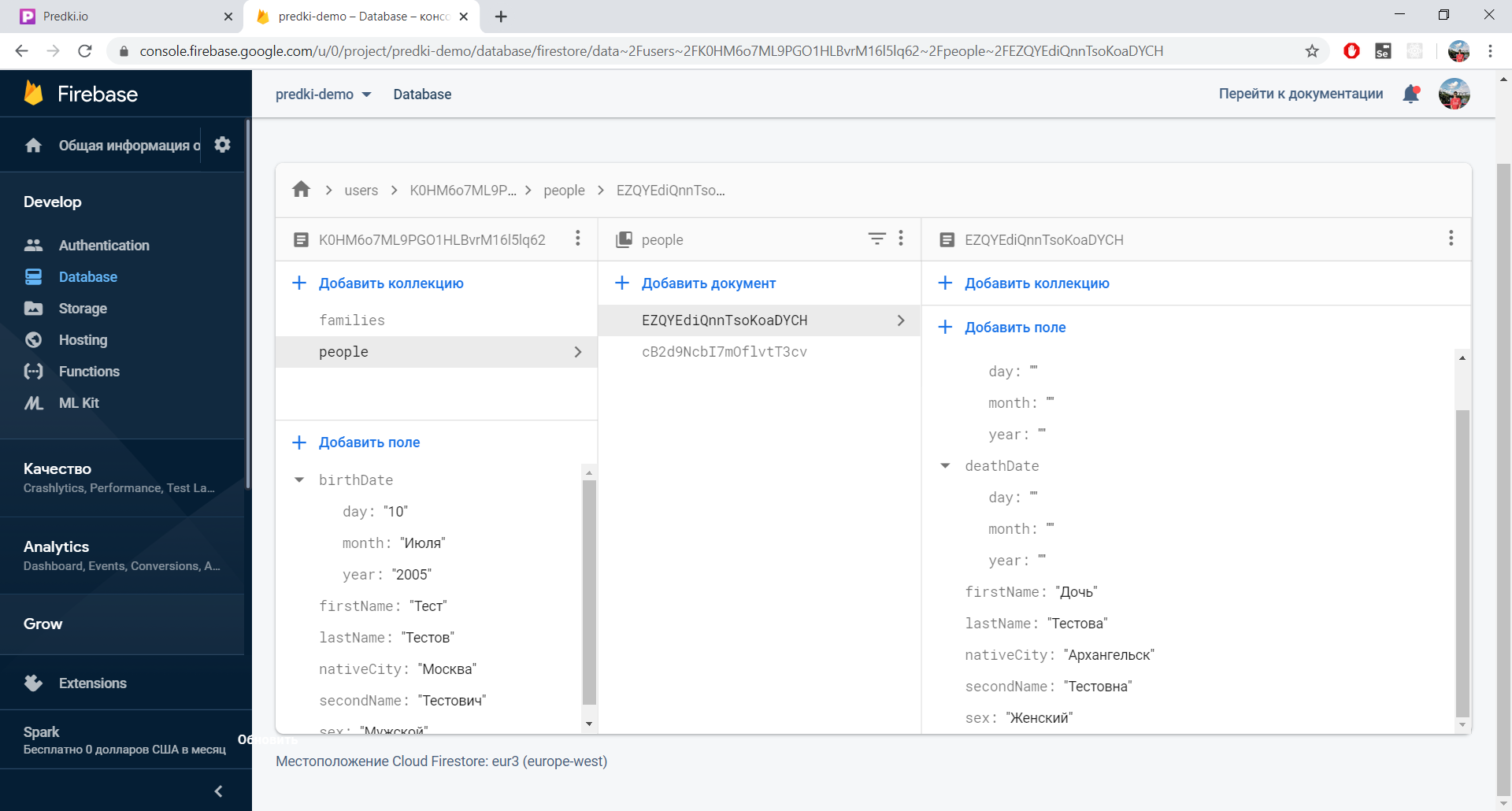


*Рис. 18. Изменение информации*

После внесения изменений результат отображается в дереве и базе данных.



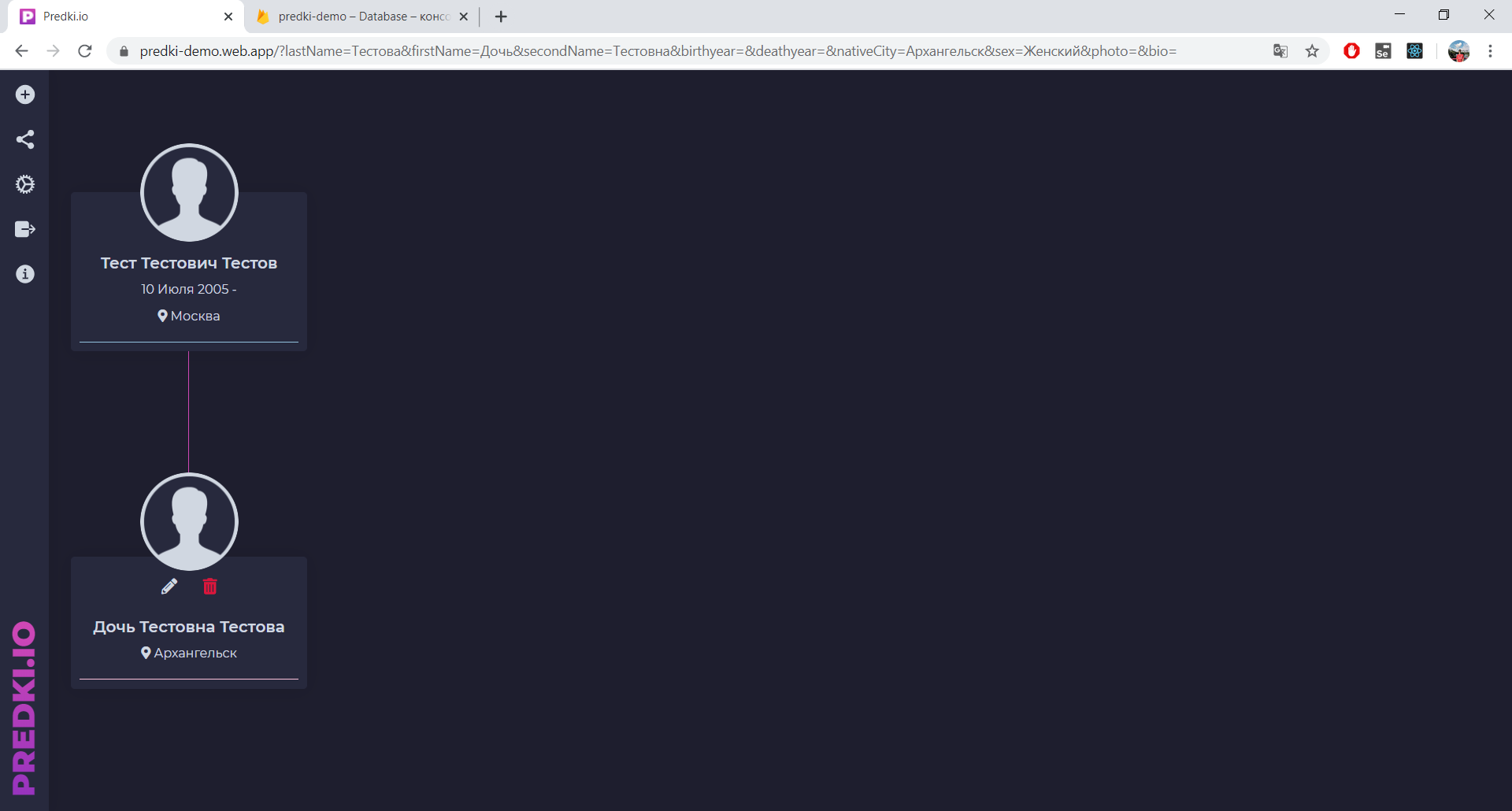
*Рис. 19. Результат изменения данных в дереве*



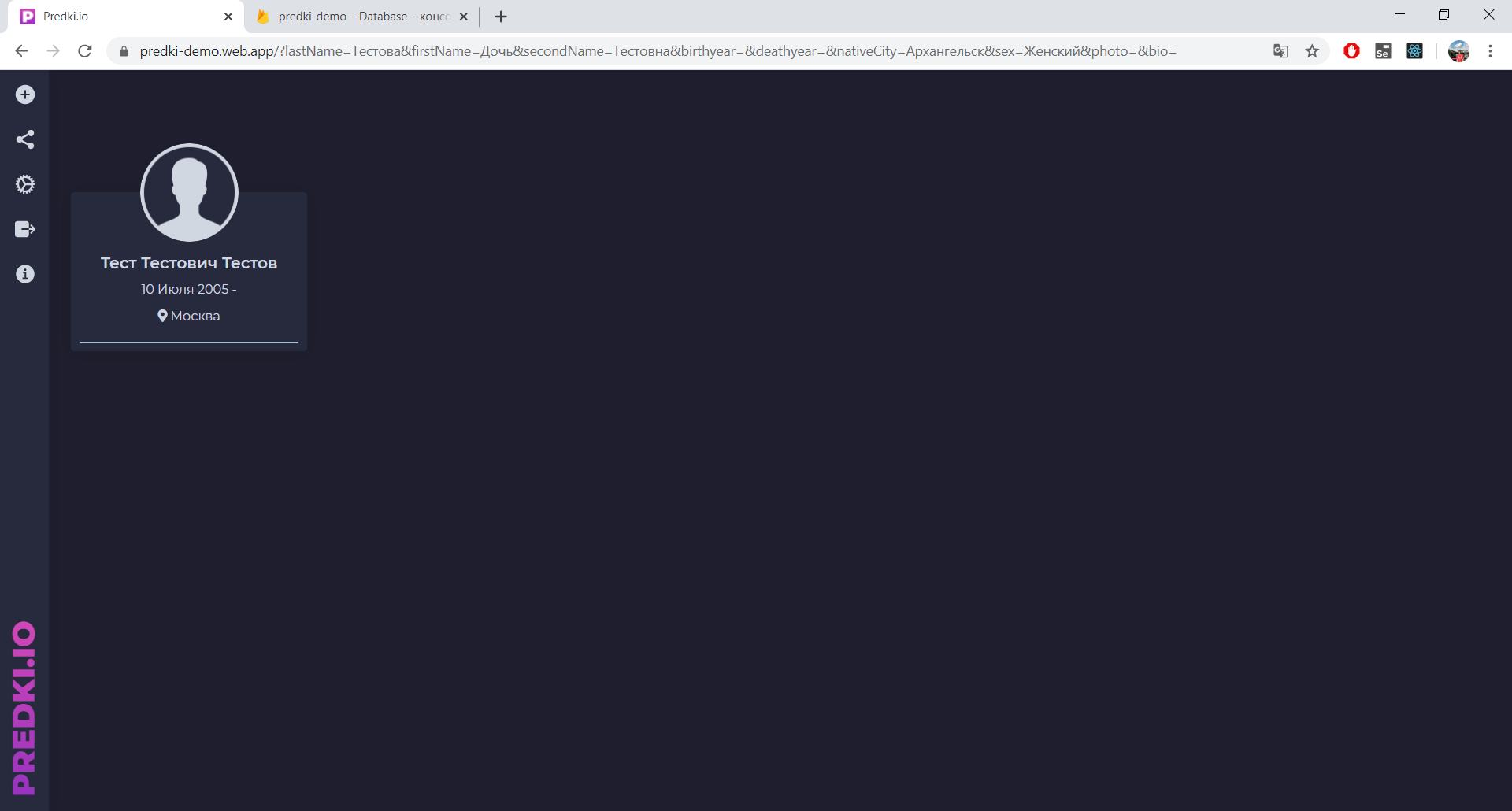
*Рис. 20. Результат изменения данных в БД*

### **6.3.1. Удаление человека из дерева**

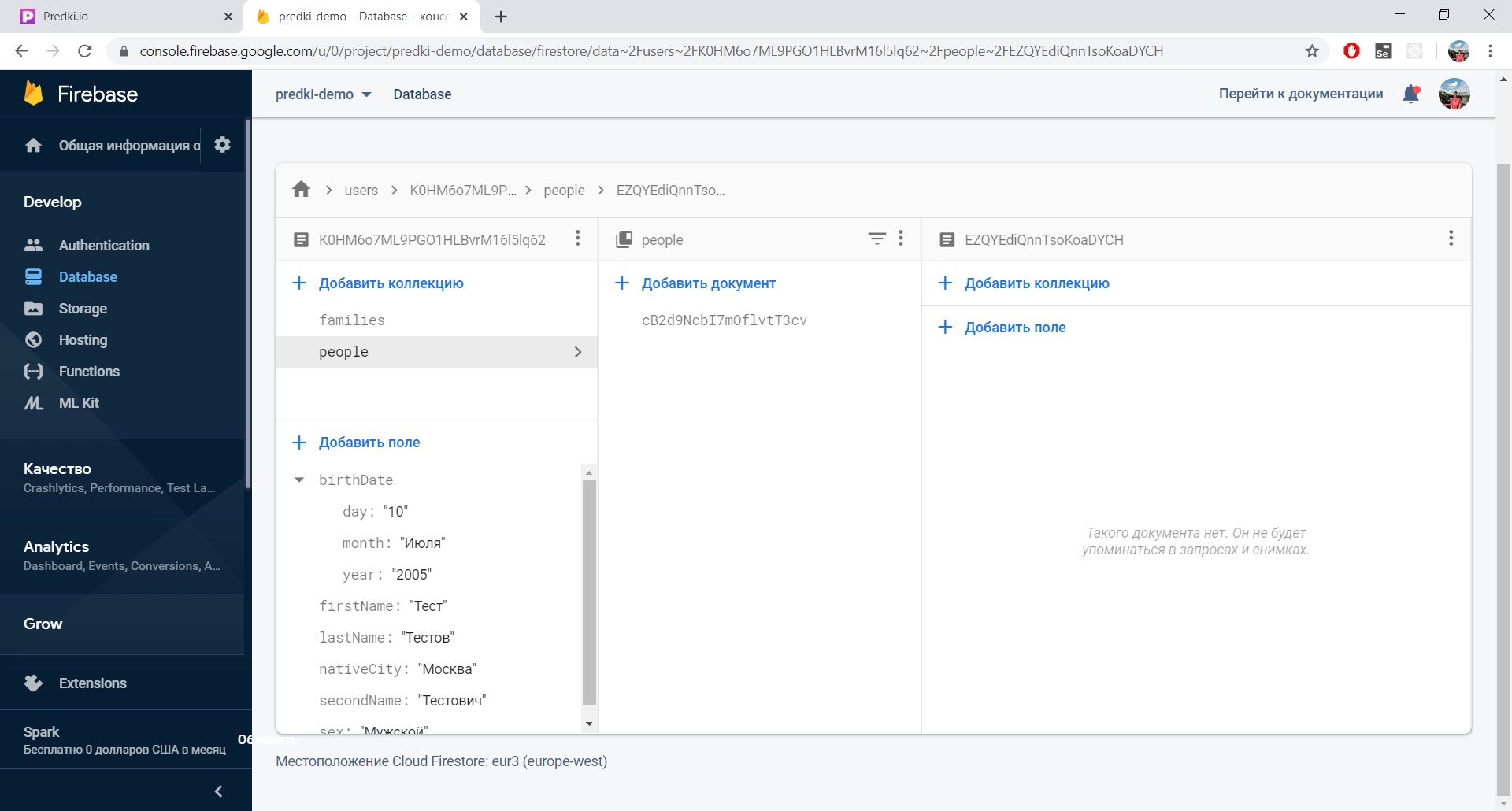
При клике на кнопке «удалить», если это возможно (не разрываются родственные связи так, что одна часть дерева оказывается оторванной от другой), человек удаляется из базы данных, а страница перезагружается с новым деревом без данного человека.



*Рис. 21. Удаление человека из дерева*



*Рис. 22. Дерево после удаления*

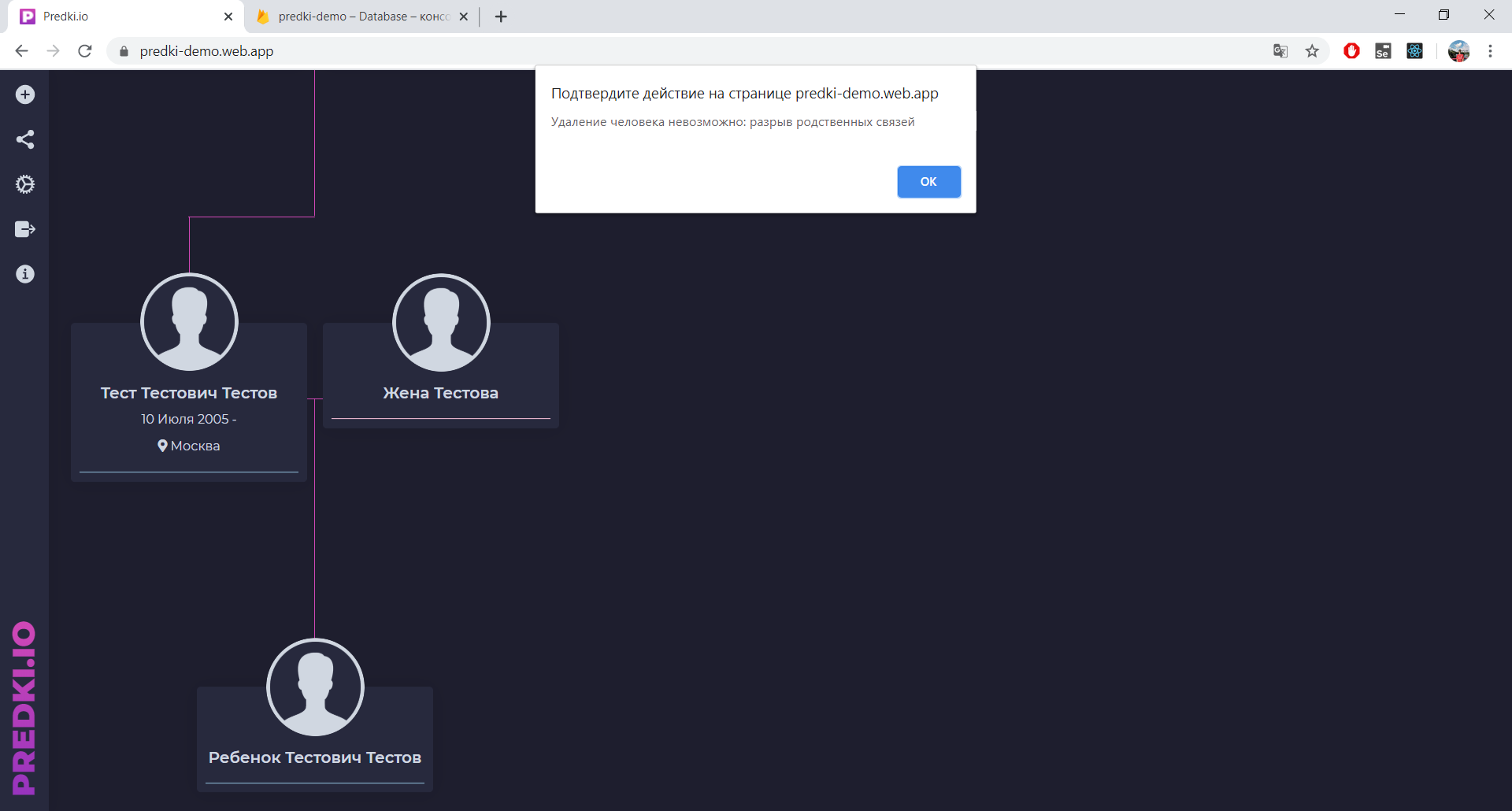


*Рис. 23. Удаление из БД (такого документа больше нет).*

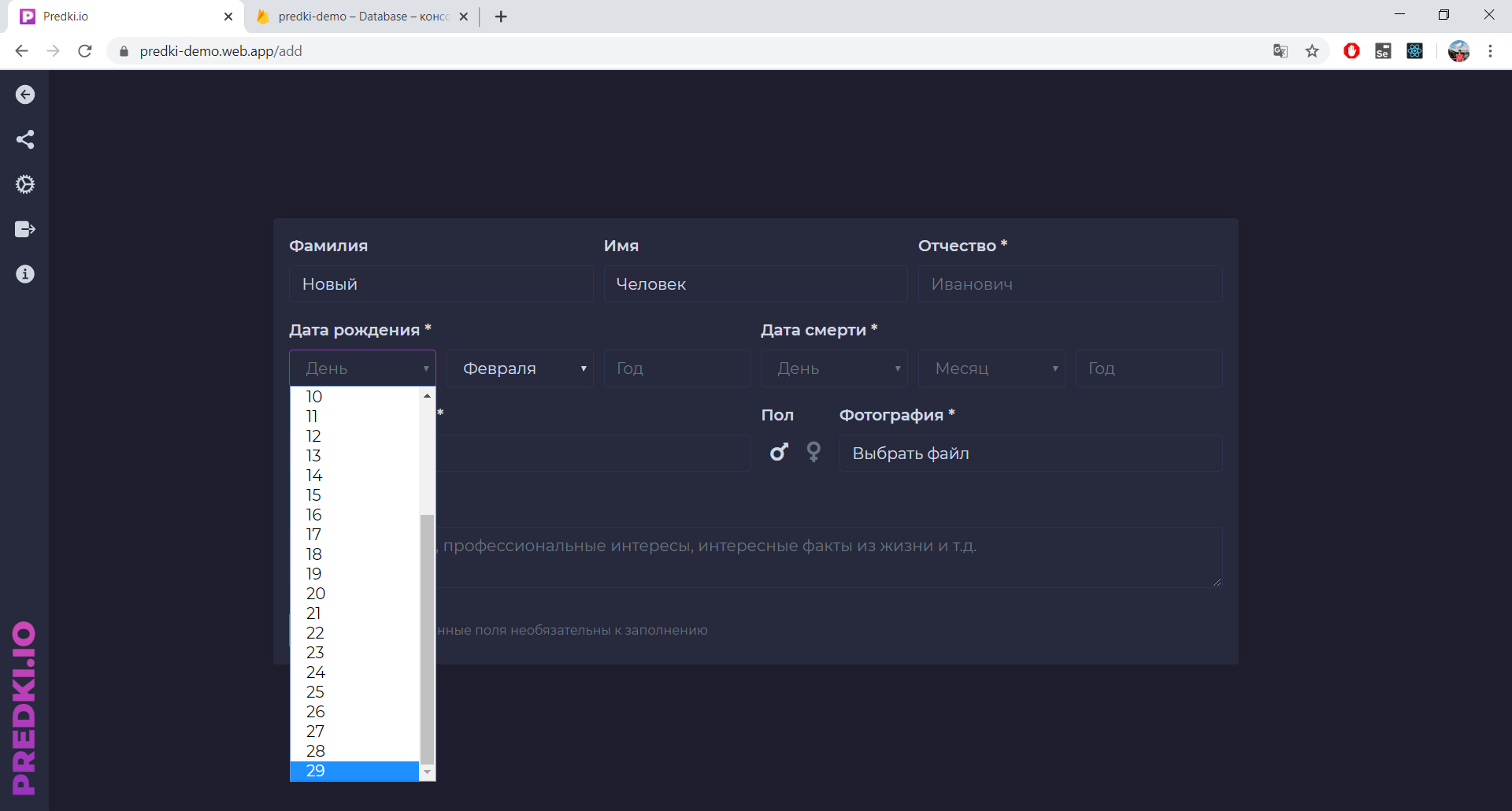
## **6.4. Испытание выполнения требований к надежности**

Программа осуществляет проверку корректности ввода информации:

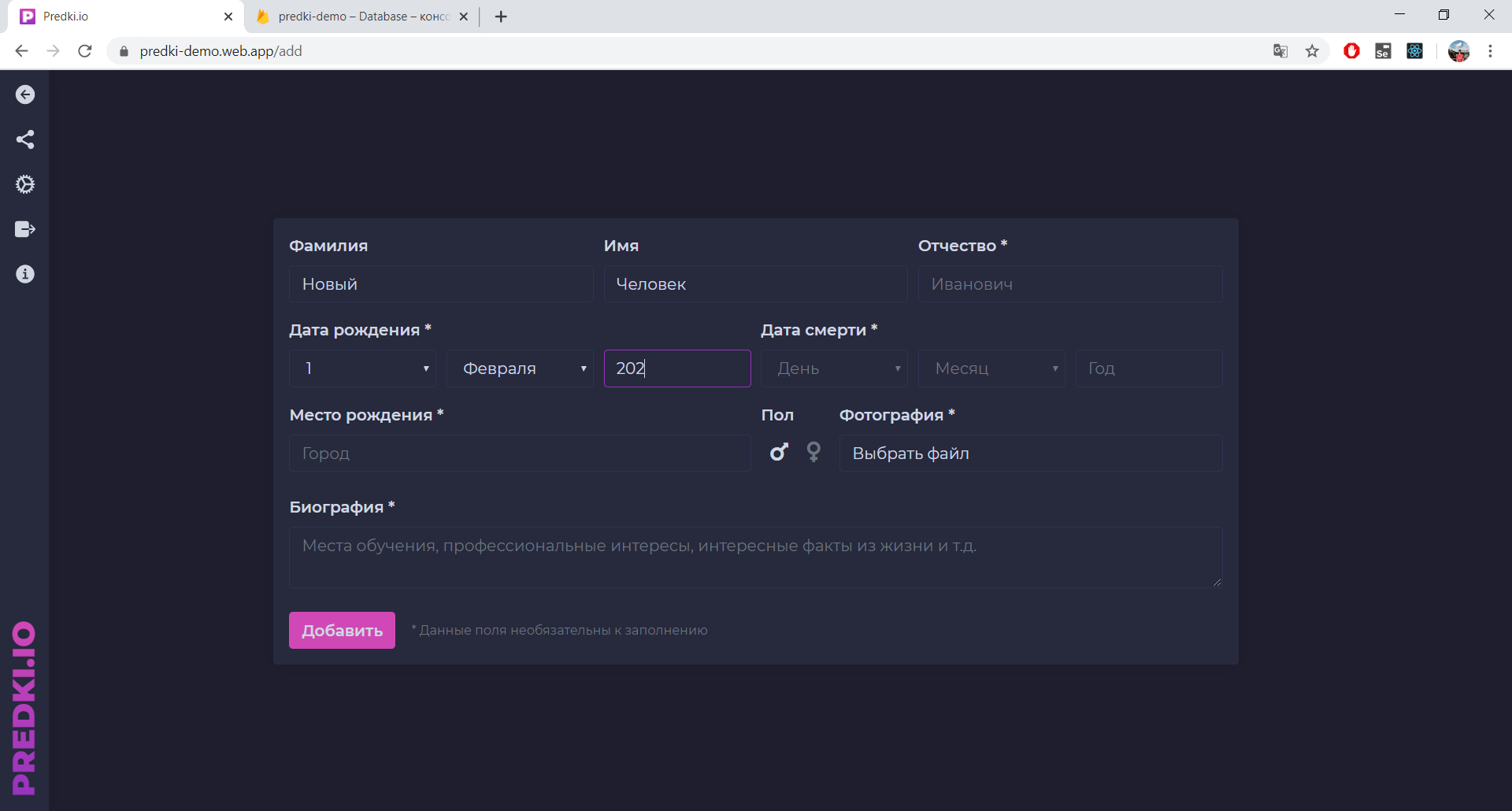
* Соответствие формата электронной почты
* Соответствие пароля и подтверждения пароля
* Корректность введенных дат (количество дней в каждом из месяцев, корректность введенного года)
* Программа сообщает об исключительных ситуациях, возникающих во время ее работы
* Программа не допускает разрыва родственных связей при удалении



*Рис. 24. При попытке удаления Теста Тестовича родственная связь между его родителями и его детьми разрывается, и программа этому препятствует, оповещая пользователя*



*Рис. 25. Программа не дает выбрать 30 или 31 февраля*



*Рис. 26. Программа не дает ввести год рождения >2020*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Лист регистрации изменений | | | | | | | | | |
| Изм. | Номера листов | | | | Всего листов (страниц) в документе | № документа | Входя- щий № сопро- водит. докум. и дата | Под- пись | Дата |
| изменен- ных | заменен- ных | новых | аннулиро- ванных |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |