[Ссылка на макет](https://www.figma.com/file/d8LhhLjMkaTfPvAcYQULNv/Library---students-file?node-id=2601%3A15229&t=NlX22Vv6Rx35bf9N-1)

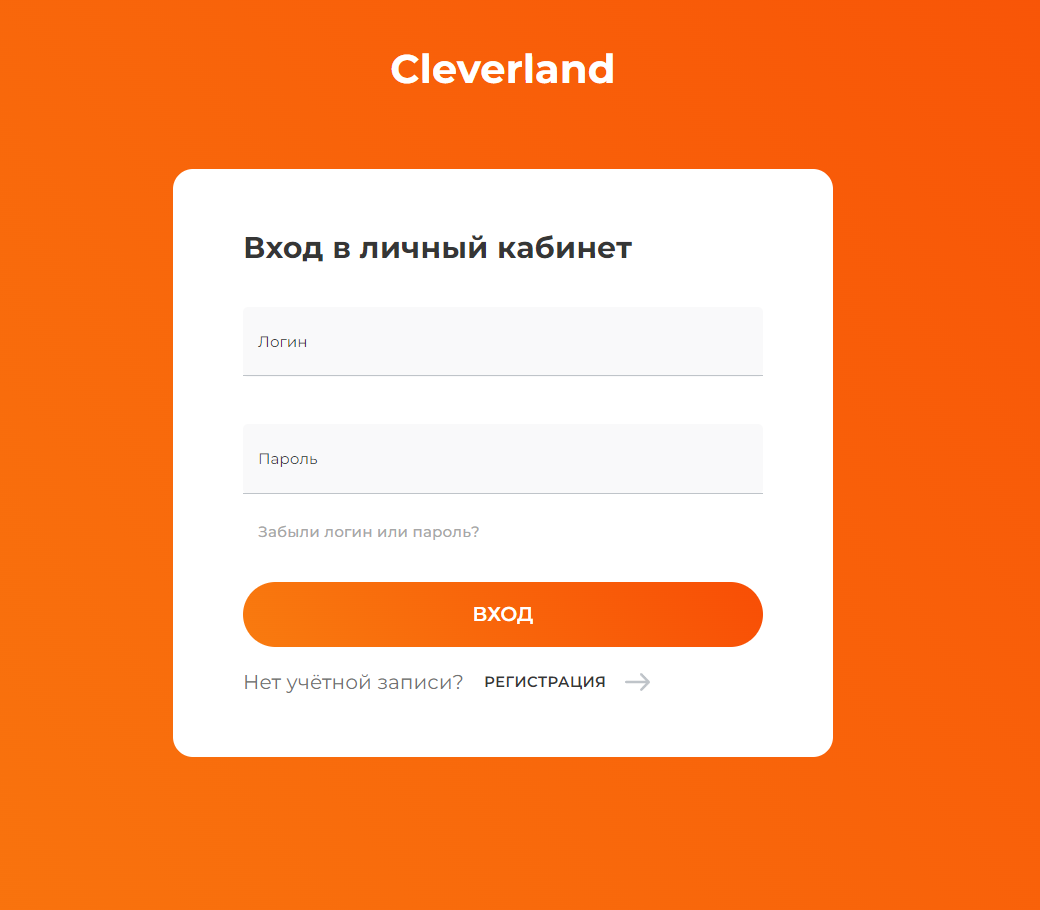
[API](https://docs.google.com/document/d/11xserDnP3F8OvwOQ97mm4vRfqAP9KlLxQeZG0URw08A/edit)

**Регистрация, авторизация, восстановление пароля**

В этом спринте вам необходимо реализовать логику регистрации, авторизации пользователя, а также восстановления пароля.

Для данного функционала нужно подключить библиотеки для работы с формами. Например, [react-hook-form](https://react-hook-form.com/get-started) (в приоритете) или [formik](https://formik.org/).

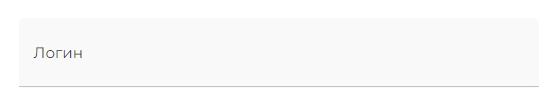
В рамках данного спринта вход в приложение теперь будет происходить через авторизацию. А именно: при загрузке приложения отображается экран авторизации, дальше происходит вход в приложение (либо регистрация и вход, либо восстановление пароля и вход) и переход на главную страницу библиотеки. Далее все запросы после успешного входа идут вместе с полем в headers - “Authorization”: "Bearer {jwtToken}" (jwtToken - токен, полученный из успешного ответа авторизации). Для этого функционала необходимо использовать [axios interceptors](https://axios-http.com/docs/interceptors). Также реализовать функционал разлогина пользователя.

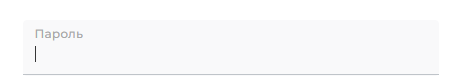
  
 Если пользователь авторизован, при попытке перехода на какой либо из роутов авторизации, происходит редирект на главную страницу. И наоборот, при попытке перехода на главную страницу неавторизованного пользователя, происходит редирект на страницу авторизации.

При отправке любого из запросов отображается лоадер:

**Схожее поведение для всех инпутов**

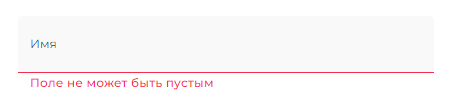
По умолчанию плейсхолдер отображается вертикально по центру инпута. При установке фокуса либо при непустом поле, плейсхолдер смещается вверх инпута.



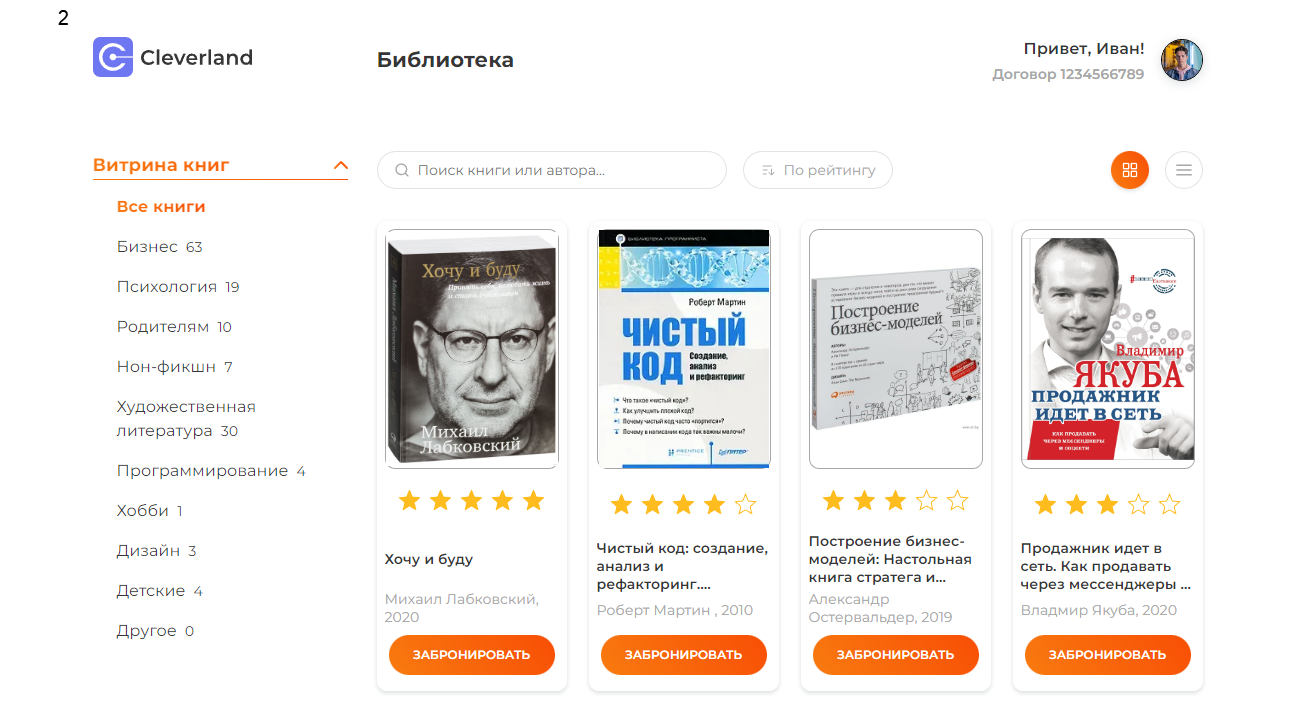
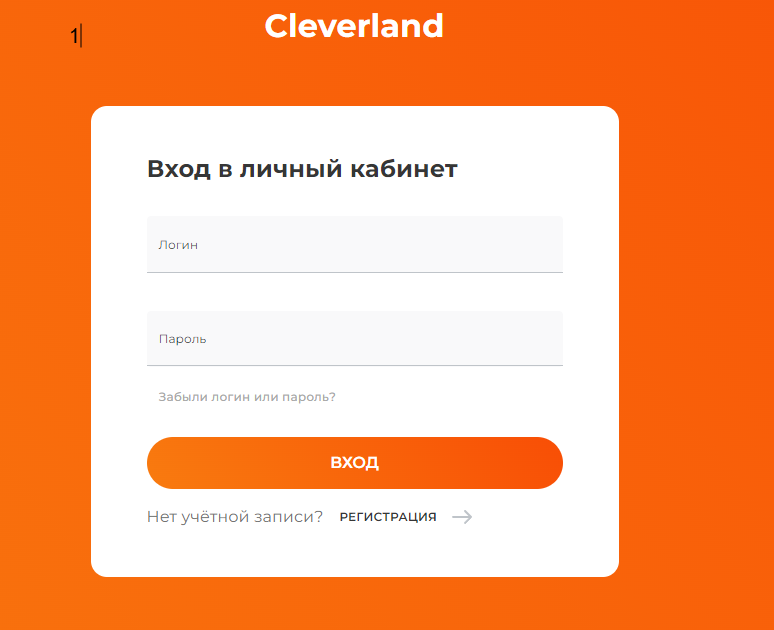


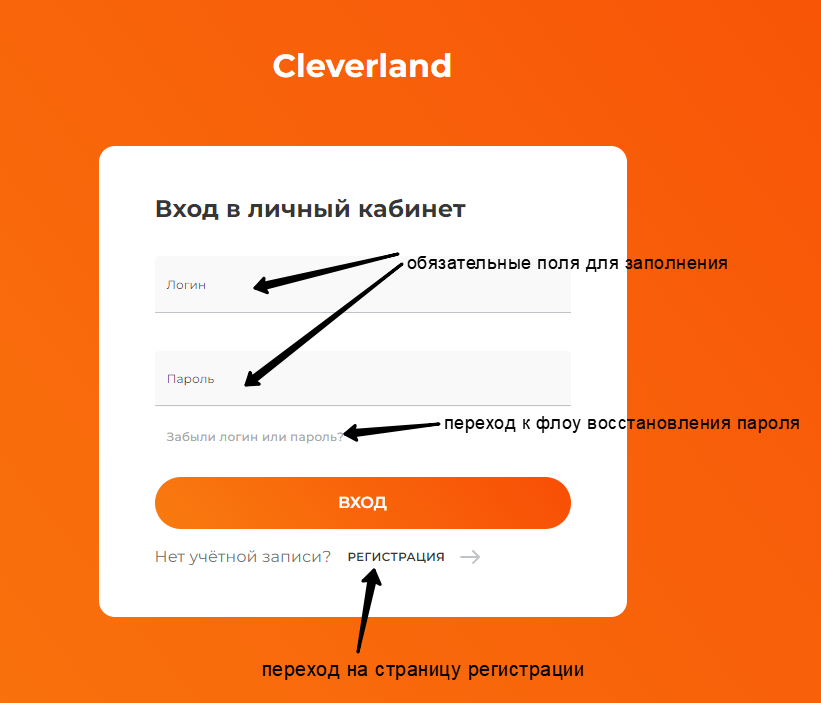
Валидация происходит при потере фокуса либо по нажатию кнопки отправки формы, при этом подсвечивается красным подсказка (либо текст ошибки) и бордер. Отправка формы не выполняется, пока не пройдена валидация.

Если поле не заполнено, при потере фокуса либо по нажатию кнопки отправки формы, отображается текст ошибки “Поле не может быть пустым” (хардкод), инпут подсвечивается красным бордером.



**Авторизация**

При первом входе в приложение для неавторизованного пользователя отображается экран с формой авторизации. После того, как пользователь успешно авторизовался, при последующем открытии приложения происходит редирект сразу на главную страницу.

**Элементы экрана авторизации**:

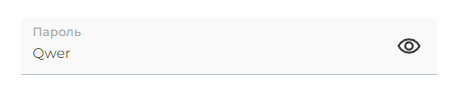
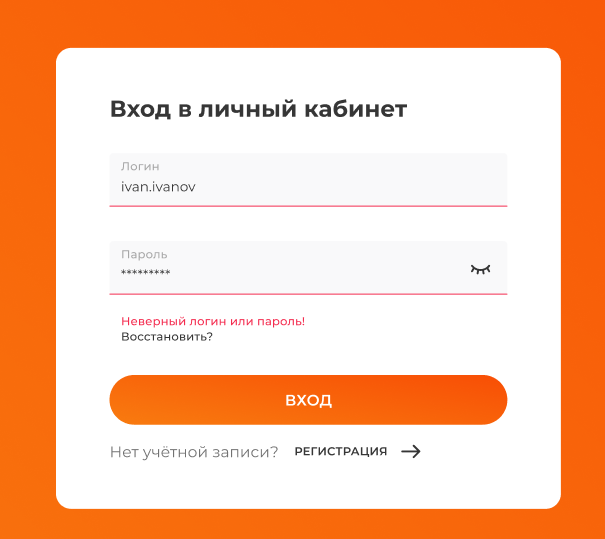
Поля ввода логина и пароля являются обязательными.

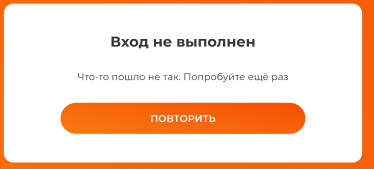
**Поведения инпута ввода пароля:**

* при заполнении по дефолту значение инпута скрывается символами, глазик закрыт;



* при нажатии на глазик, глазик раскрывается, отображается введенное значение;

  
 При нажатии на кнопку “Вход” (введенные данные не валидируются), происходит отправка запроса POST /api/auth/local (все запросы и тело запросов подробно описаны в [api](https://docs.google.com/document/d/11xserDnP3F8OvwOQ97mm4vRfqAP9KlLxQeZG0URw08A/edit)). В успешном ответе приходит токен (поле jwt). Данный токен нужно добавлять в каждый запрос в авторизованной зоне в headers: “Authorization”: "Bearer {jwt}". Далее все запросы без headers: “Authorization”: "Bearer {jwt}" в авторизованной зоне будут падать с ошибкой.   
 При получении **ошибки запроса POST /api/auth/local:**   
- если статус ответа 400 - инпуты подсвечиваются красным бордером, под инпутами текст ошибки (хардкод), под текстом ошибки появляется ссылка “Восстановить” на переход к восстановлению пароля;

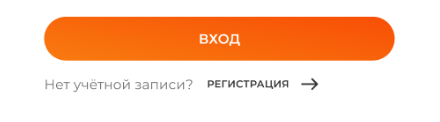
* другие статусы отображается экран:

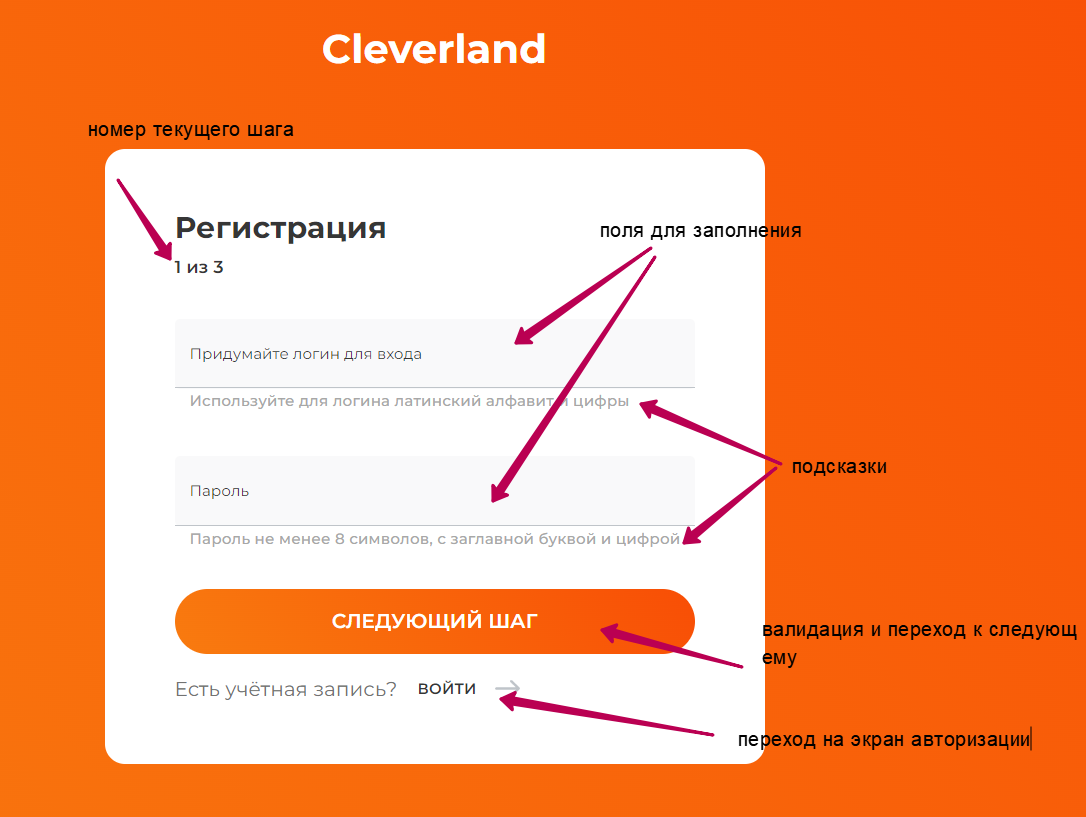
**2. Регистрация**

На экран регистрации попадаем (только для неавторизованных пользователей):

- по прямой ссылке;

- по нажатию на кнопку “регистрация” на экране авторизации.

  
**Элементы экрана регистрации.**



Экран авторизации состоит из 3 шагов, в каждом шаге все поля обязательные. Так же добавлена валидация и маска телефона. Переход к следующим шагам не осуществляется, пока не пройдена валидация.

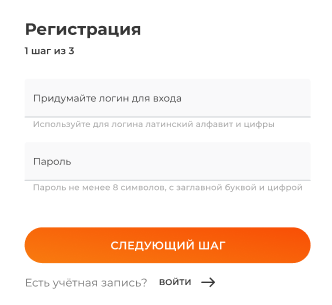
**Требования к валидации:**

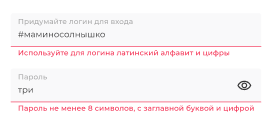
* логин: должен быть латинский алфавит и цифры;
* пароль: должно быть не менее 8 символов, заглавная буква и цифра;
* email: соответствует регулярному выражению для email.

Для поля ввода телефона - **ввод по маске** +375 (xx) xxx-xx-xx, где x - только цифры. Допустимые символы - только цифры. Пользователь начинает вводить с кода оператора, а код +375 (и остальные символы в маске) подставляется автоматически. Код оператора может быть только операторов РБ - 29|33|25|44.

**Логика форм регистраци:**

**На первом шаге** поля ввода логина и пароля подлежат валидации.

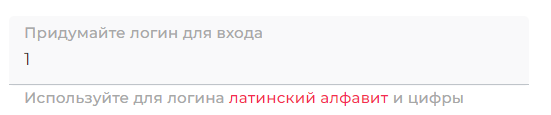


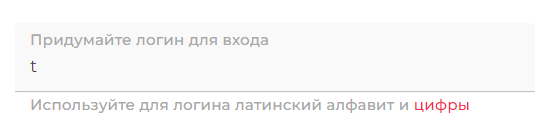
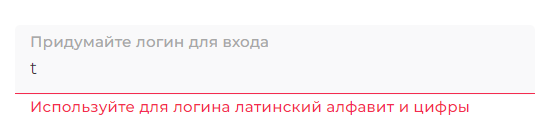


Валидация поля ввода логина.

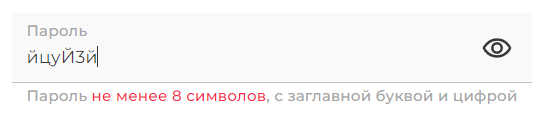
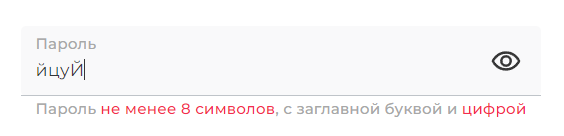
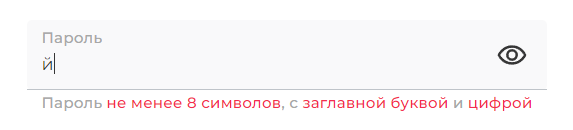
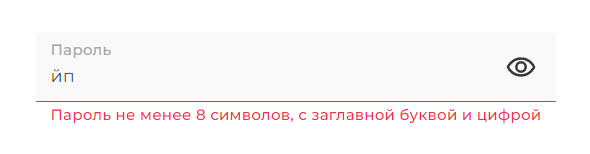
В поле ввода пароля необходимо реализовать валидацию по мере ввода, то есть пока пароль не соответствует требованиям, подсвечивать соответствующее требование:

1. введена только цифра

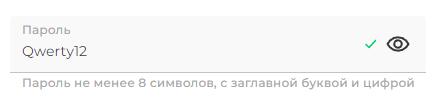


1. введена только буква латинского алфавита ****
2. при потере фокуса либо по нажатию кнопки “следующий шаг”, подсвечиватся полностью красным текст подсказки

Валидация поля ввода пароля.  
В поле ввода пароля необходимо реализовать валидацию по мере ввода, то есть пока пароль не соответствует требованиям, подсвечивать соответствующее требование:

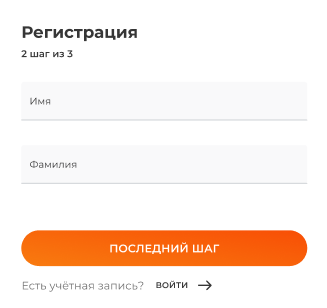
1. введено меньше 8 символом, но есть заглавная буква и цифра  
   
2. введено меньше 8 символов, нету цифры, но есть заглавная буква
3. не соблюдено ни одно требование ****
4. при потере фокуса либо по нажатию кнопки “следующий шаг”, подсвечиватся полностью красным текст подсказки

Если поле ввода пароля прошло валидацию, рядом с глазиком отображается зеленая галочка.

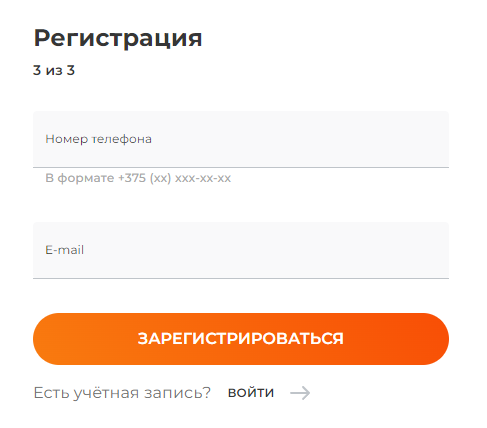


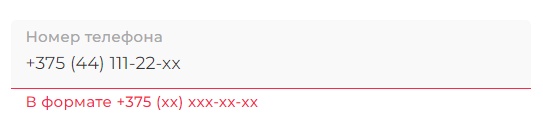
При нажатии на кнопку “следующий шаг” происходит валидация. Если она пройдена - переход на второй шаг.

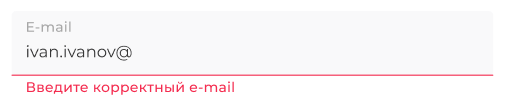
**На втором шаге** инпуты проверяются только на пустое поле.



После заполнения всех полей и нажатия на кнопку “последний шаг” происходит переход к последнему третьему шагу.  
**На третьем последнем шаге** для поля ввода телефона необходимо реализовать маску. Можно реализовать самостоятельно, а также использовать библиотеки, например, [react-text-mask](https://www.npmjs.com/package/react-text-mask).

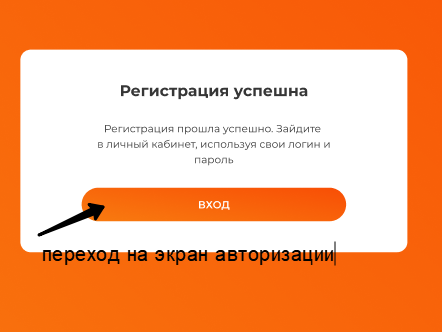


При несоответствии маске номера телефона инпут подсвечивается красным подсказка и бордер:

Для валидации поля ввода email используйте регулярное выражение. При несоответствии email регулярному выражению инпут подсвечивается красным ошибка(хардкод) и бордер.

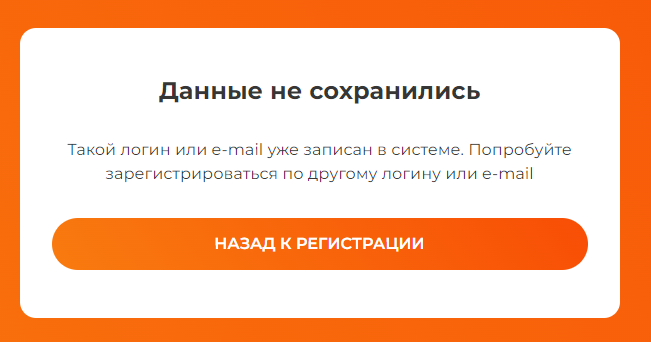
Если все поля прошли валидацию, при нажатии на кнопку “зарегистрироваться”, отправляется запрос POST /api/auth/local/register со всеми данными, что вводили в предыдущих шагах (тело для данного запроса смотрите в [api](https://docs.google.com/document/d/11xserDnP3F8OvwOQ97mm4vRfqAP9KlLxQeZG0URw08A/edit)).

**После успешного ответа запроса** происходит переход к экрану успешной регистрации, с которого по кнопке “вход” попадаем на экран авторизации. Происходит флоу авторизации.

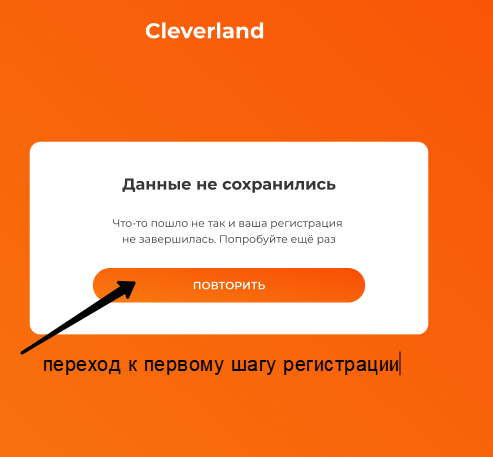


При **получении ошибки запроса POST /api/auth/local/register**

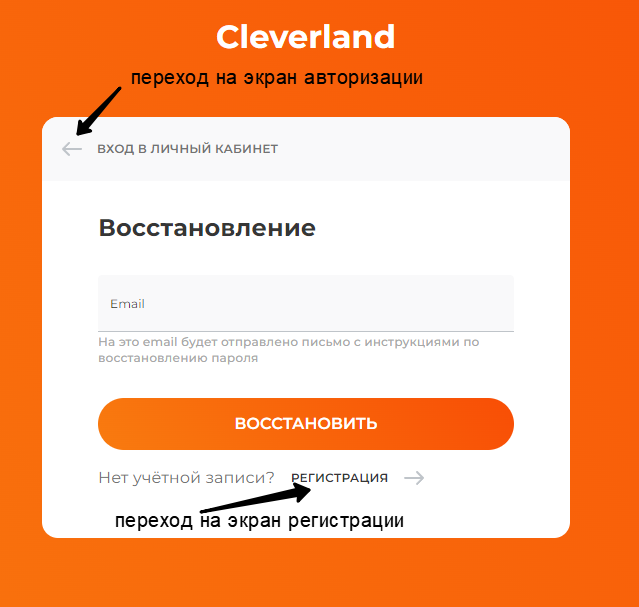
* если статус ответа 400 (это ошибка о том, что такой логин или пароль уже существуют), отображается экран ниже. На этом экране по нажатию на кнопку “назад к регистрации” происходит переход к первому шагу регистрации. Все поля на первом шаге и на последующих очищены.



* остальные ошибки - переход на экран “Данные не сохранились”. На этом экране по нажатию на кнопку “повторить” происходит переход к первому шагу регистрации. Все поля на первом шаге и на последующих очищены.

****

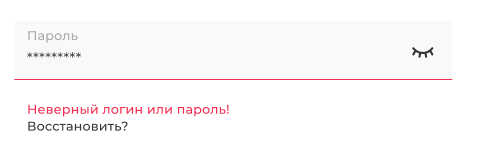
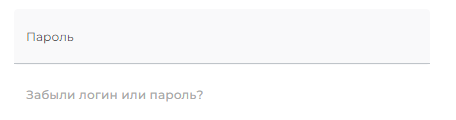
**3. Восстановление пароля**



На экран восстановления пароля попадаем (только для неавторизованных пользователей):

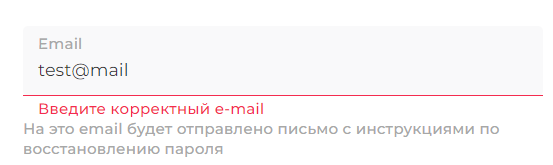
- по прямой ссылке;

- по нажатию по ссылкам “восстановить?” или ”забыли логин или пароль?” на экране авторизации.

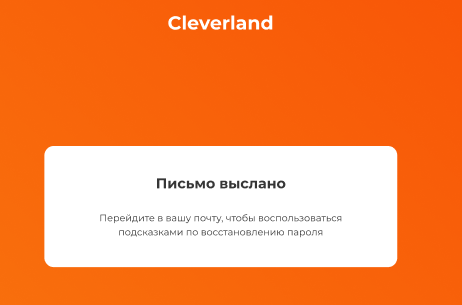


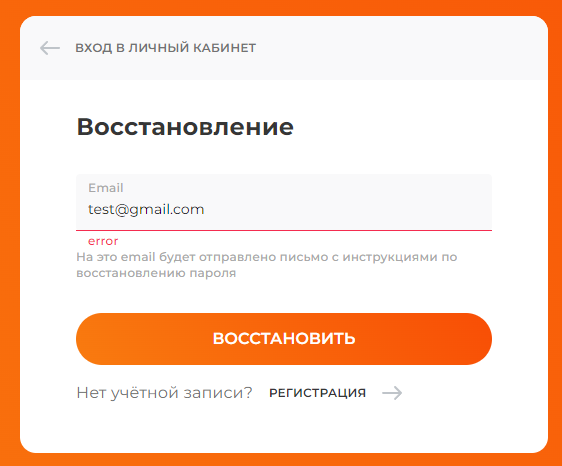
**Логика флоу восстановления пароля:**

В поле ввода email вводим реальный email зарегистрированного пользователя. Инпут валидируется только на пустое поле, а также на соответствие регулярному выражению (аналогично во флоу регистрации).

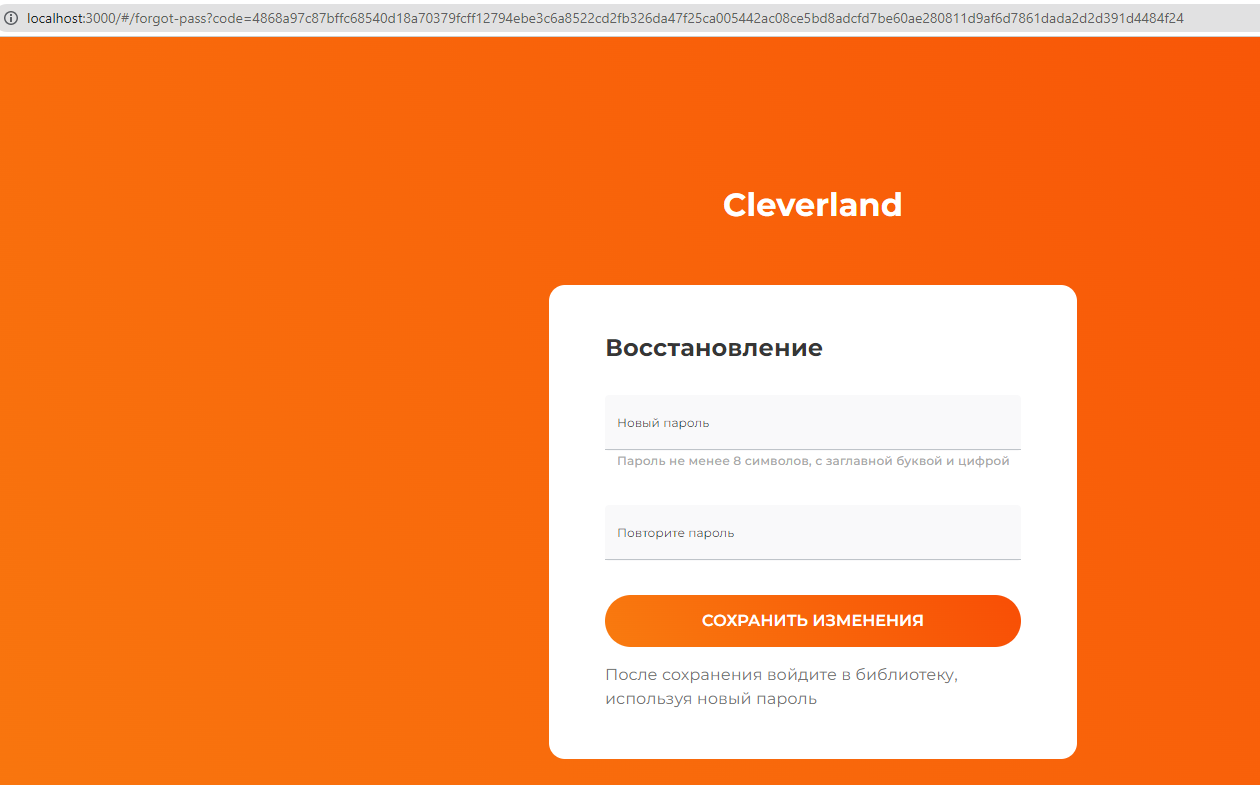


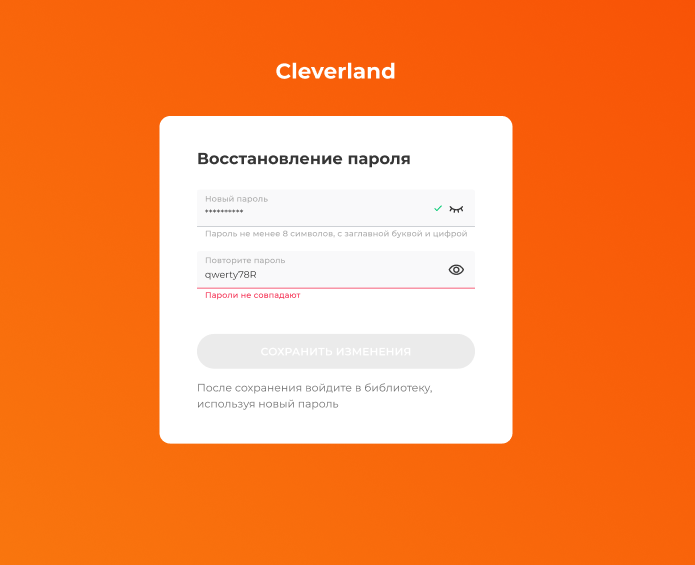
По нажатию на кнопку “восстановить” происходит отправка запроса POST /api/auth/forgot-password. После успешного ответа сервера отображается экран успеха.

  
При получении ошибки в ответе запроса, текст ошибки отображаем под инпутом, например:



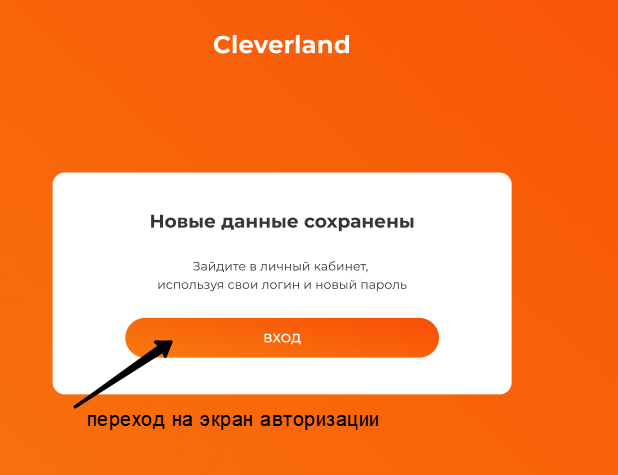
В свою очередь на указанную почту отправляется письмо с ссылкой, содержащую код для сброса пароля, например: <http://localhost:3000/#/forgot-pass?code=d31eebfdc2ebac7a4ac149130be2ef572b4d8451dd987b7c8e6746750c73567f40035eeb236ab9214db9cfdf4db6b46fd4d0cdf77278d647ef7c2b58eabc06e3>. Нажав на данную ссылку, пользователь переходит на страницу ввода нового пароля, отображается следующий экран:



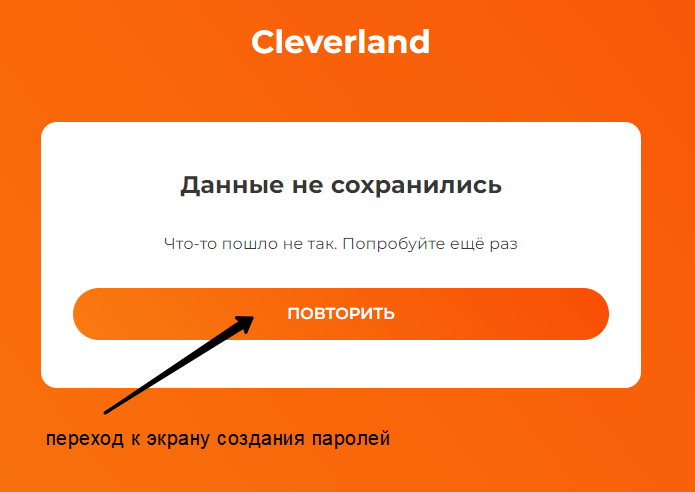
Валидация полей ввода паролей аналогична в форме регистрации. За исключением того, что подсказка и зеленая галочка об успешной валидации отображается только под первым инпутом. Так же добавлена валидация на то чтобы пароли совпадали. При несовпадении паролей второе поле подсвечивается ошибкой, кнопка дизэйблится.  


При нажатии на кнопку сохранить происходит отправка запроса POST /api/auth/reset-password.

В зависимости от ответа сервера отображаем экраны:  
 - при успехе:

\

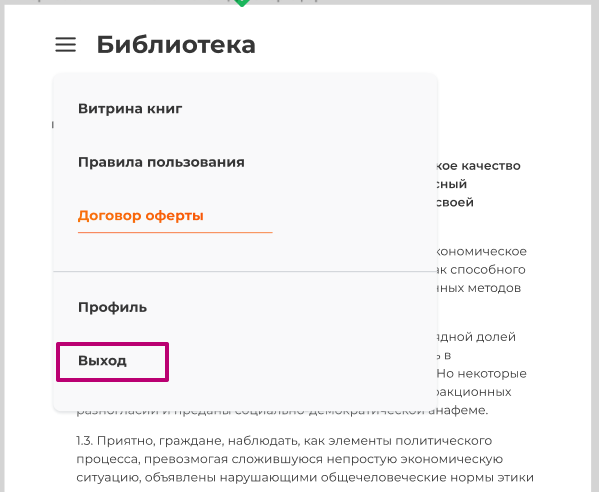
* при ошибке:



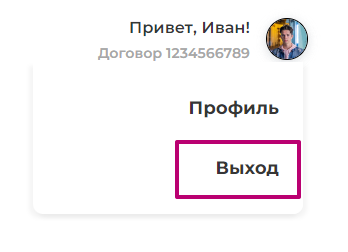
**4. Разлогин**

В авторизованной зоне необходимо реализовать функционал разлогина пользователя по нажатию на кнопки выхода:

* в навигационном меню



* в хэдере



По нажатию на данные кнопки происходит разлогаут пользователя, редирект на страницу авторизации. При попытке перехода на главную страницу неавторизованного пользователя, происходит также редирект на страницу авторизации.

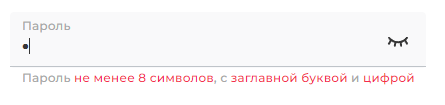
**Тесты**

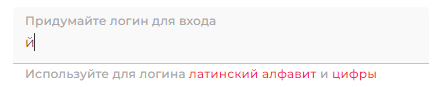
Названия роутов должны совпадать со следующими:

* авторизация - auth,
* регистрация - registration,
* восстановление пароля - forgot-pass.

Атрибут name для всех инпутов задавать строго в соответствии с данными в API. То есть name у инпута, значение которого вы будете передавать в запросе, должен совпадать с названием параметра.

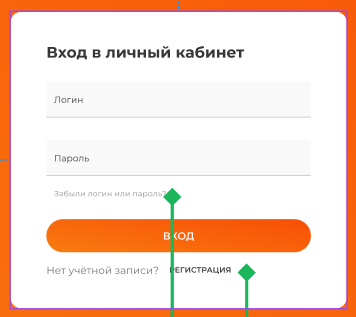
Для корректной проверки валидации полей логина и пароля подсвеченные красным слова в ошибках под инпутами должны быть обернуты дополнительно в span:



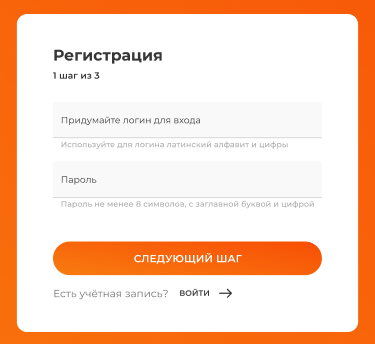


Атрибуты **data-test-id** для следующих компонентов:

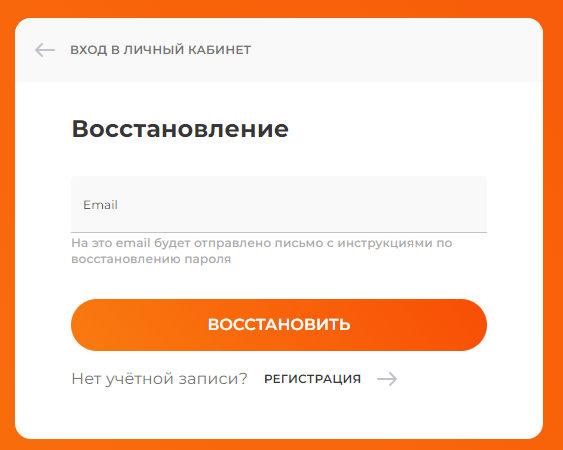
* общая обертка (layout) над всеми компонентами авторизации - auth
* форма авторизации - auth-form (для тэга form, кнопка лежит в форме)



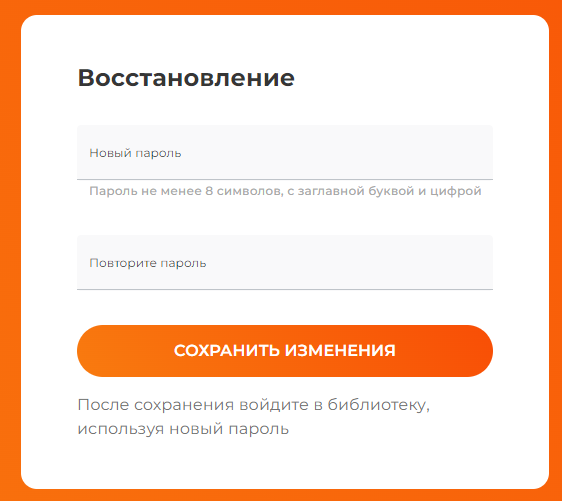
* форма регистрации register-form (для тэга form, все кнопки лежат в форме)



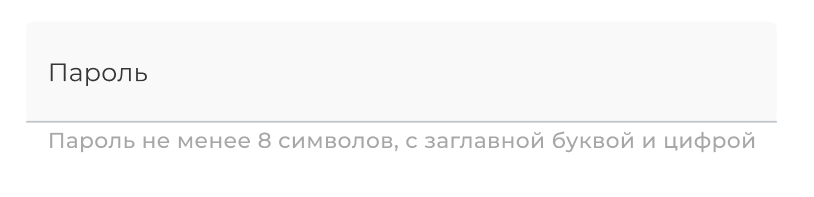
* форма отправки email send-email-form (для тэга form, кнопка лежит в форме)

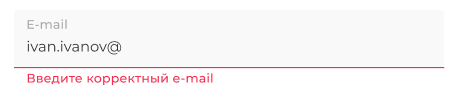


* форма сброса пароля reset-password-form (для тэга form, кнопка лежит в форме)



* подсказка (она же и ошибка) под полем инпута hint

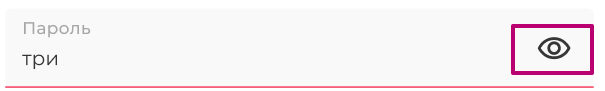




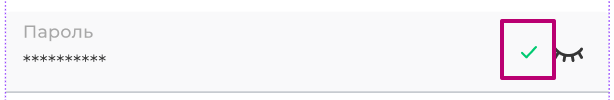
* закрытый глазик eye-closed



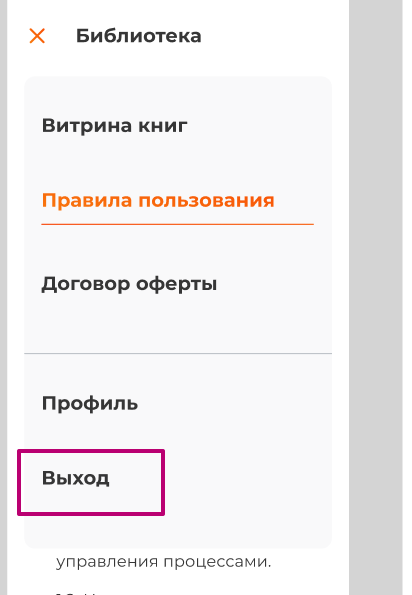
* открытый глазик eye-opened



* галочка checkmark



* кнопка разлогина exit-button из меню



* блок статуса после ответа сервера status-block

