



Identié

HOW TO START AUTHENTICATING WITH NOPASS™

Identié™™, Inc.
3035 Turtle Brooke
Clearwater, Florida 33761 USA

www.Identite.us

VERSION: 0.3.0
Dated: 19 February 2020

Trademark Notice: Product or corporate names may be trademarks or registered trademarks, and are only used for identification and explanation, without intent to infringe.

Copyright Notice:

Copyright © 2020 Identité™, Inc. All rights reserved.

Permission to copy for internal use only is granted to Identité™. This document may not be reproduced or distributed in whole or in part in any form outside of Identité™. without prior written permission from Identité™, Inc.

CONTENTS

CONTENTS..... III

1. HOW TO START..... 1

2. REALIZATION OF THE API TO INTEGRATE THE NOPASS™ SERVER ON YOUR SITE: 4

3. STEPS FOR USER AUTHENTICATION VIA NOPASS™..... 10

4. STEPS FOR USER REGISTRATION ON NOPASS™ 12

5. STEPS FOR DELETE USER IN NOPASS™ 14

6. STEPS FOR UPDATE USER’S INFORMATION IN NOPASS™ 16

1. HOW TO START

Changing the UI (web site):

1. Adding an interaction button to utilize authentication via Identite.
For example, adding a button for choosing a new authorization type (see Image 1).
2. Allocating a section to show the OTP (picture and digits) received from Identite.
This area should be located on the log in page (see Image 2). When the user selects authentication via Identite, the website server-side will send a request to Identite server. After that, the Identite server generates and sends an OTP (picture and digits) to the website server-side to be displayed in the said area. This picture is a PNG file 8-bit (PNG-8) that is coded in base64 format.
3. (Optional) Allocating a section for the authentication counter. (Expiration time of the OTP).
When the user selects authentication via Identite, the website server-side will send a request to Identite server. After that, the Identite server generates and sends an OTP (picture and digits) to the website server-side to be displayed in the said area. This OTP comes with an expiration time that can be added in the section beneath the OTP.
4. Add functions to call registration and authentication of the user in Identite (see “Realize next API for integration with NoPass™ server on your site” and “Steps to user registration in the NoPass™ system” sections)

[Demo](#) [Home](#) [Privacy](#)[Create Account](#) [Login](#)**PRESHOP**

Shop with us

Welcome!

Sign in to PreShop or [create an account](#).[Next](#)☐ Stay signed in

Using a public or shared device?

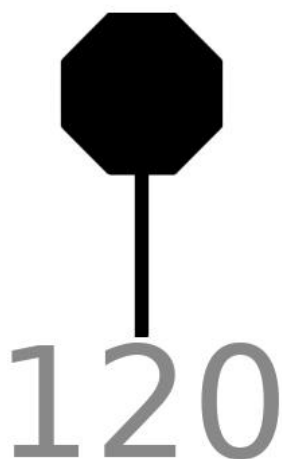
Uncheck to protect your account.

Figure 1 – Demo site login page

[Demo](#) [Home](#) [Privacy](#)[Create Account](#) [Login](#)**PRESHOP**

Shop with us

Welcome!

Sign in to PreShop or [create an account](#).

Picture changes in: 26

☐ Stay signed in

Using a public or shared device?

Uncheck to protect your account.

Figure 2 – Authentication process with NoPass™

2. REALIZATION OF THE API TO INTEGRATE THE NoPASS™ SERVER ON YOUR SITE:

Connection scheme should be https.

Scheme: **PortalUrl/_api_**

PortalUrl is the fully qualified URL of the portal, which is using Identite to authenticate its users.

PortalUrl is string type and length ≤ 2048 .

Example, https://MyWebSite/ or https://WebPortals/Portal1/.

api - api route that should be realized.

Relative Url		/api/PortalCommunication/ConfirmPreRegistration
Description		There is validated the administrator user name of your site.
Request	Content type	application/json-patch+json
	Body	<pre>{ // parameter with length <= 64 and case sensitive. // This parameter should be validated that // this user is the administrator user of your site. "adminId": "string", // parameter with length <= 8 bytes and none negative digits. // This number is coming from server and used for the // random challenge between the NoPass™ authentication //server and the portal "r": 0 }</pre>
Response	Status	200 in case everything is ok. Other in case something is wrong with the parameters during validation
	Content type	text/plain
	Body	<pre>{ // parameter with length <= 36 and case sensitive. // This value is from request body. Have to verified that // it is valid administrator name. "adminId": "string", // parameter which is defined by administrator for registration portal (case sensitive, digits, alphabetic). "sCode": "string", // parameter with length <= 8 bytes and none negative digits. // This number is equal r (from input model) + 1 "r": 0 }</pre>
Example		https://MyWebSite/api/PortalCommunication/ConfirmPreRegistration

Relative URL		/api/PortalCommunication/ConfirmRegistration
Description		Validation of the unique identification number and settings of your site.
Request	Content type	application/json-patch+json
	Body	<pre>{ // settings of the portal in a string format "settings": "string", // type parameter with length <= 256. This value is coming from the Identite server. // Save this value to use for registration and authentication of users in the Identite system. "portalId": "string", // type parameter with length <= 256. This value is coming from the Identite server. // Save this value to authenticate requests against Server API with Bearer scheme. "authToken": "string" }</pre>
	Status	200 in case of all is ok. Other in case of something wrong with parameters during validation
Response	Content type	text/plain
	Body	<pre>{ // parameter which is defined by administrator for registration portal (case sensitive, digits, alphabetic). "sCode": "string" }</pre>
Example		https://MyWebSite/api/PortalCommunication/ConfirmRegistration

Relative URL		/api/PortalCommunication/ConfirmUserRegistration
Description		There is directed to the installation of the application. Use the link provided to show the QR code and store the OTP for using it in future.
Request	Content type	application/json
	Body	<pre>{ // parameter with random characters. // Secure random generated value used to validate the registration. // Upon registration the portal will be asked to enter the registration code, // there it will be typed in. "otp": "string" }</pre>
	Status	200 in case of all is ok. Other in case of something wrong
Response	Content type	application/json
	Body	No answer needed
Example		https://MyWebSite/api/PortalCommunication/ConfirmUserRegistration

Relative URL		/api/PortalCommunication/UpdatePicture
Description		This needs to receive updated image from the NoPass™ system. This image should be showed on a web-page for user to user can compare it with picture on his (her) phone.
Request	Content type	application/json-patch+json
	Body	<pre>{ // Information about the current authentication session provided by the server. "authId": "string", // parameter with PNG file 8-bit (PNG-8) // that is coded in base64 format picture need to be showed on a web-page. // User has to check number on this image with on his (her) phone during authorize process. "image": "string", // Login of the user. It's matter for login via social network to make portal know the user, which was chosen on the social page "userId": "string", // parameter showed expiration period in milliseconds for this image. // Can be used to show count down indicator. "nextChange": 0 }</pre>
Response	Status	200 in case of all is ok. Other in case of something wrong
	Content type	application/json
	Body	No answer needed
Example		https://MyWebSite/api/PortalCommunication/UpdatePicture

Relative Url		/api/PortalCommunication/AuthorizedUser
Description		This needs to receive authentication information from the NoPass™ system.
Request	Content type	application/json-patch+json
	Body	<pre>{ // parameter used to identification the authentication session. // Information about the current authentication session provided by the server. "authId": "string", // parameter showed authentication status. // True: authentication was successful. // False: authentication was not successful. "isAuthorized": true, // A text message that describes the reason for the denial of the authentication. // This message is only shown for an unsuccessful attempt. "reason": "string" }</pre>
Response	Status	200 in case of all is ok. Other in case of something wrong
	Content type	application/json
	Body	No answer needed

Example		https://MyWebSite/api/PortalCommunication/AuthorizedUser
Relative Url		/api/PortalCommunication/ValidateUserRegistration
Description		This needs to validate user's data when the user tried to register from external services (social networks)
Request	Content type	application/json-patch+json
	Body	<pre>{ // Secure random generated value used to validate the registration. // Needs to understand which registration process is about "otp": "string", // First name of the registering user "givenName": "string", // Surname of the registering user "surName": "string", // Phone number of the registering user "phoneNumber": "string", // Email of the registering user "email": "string", // Login of the registering user which will be used later during authorization "login": "string", // link to the profile image "profileImageUrl": "string" }</pre>
Response	Status	200 in case of all is ok. Other in case of something wrong
	Content type	application/json
	Body	<i>// True if provided information is acceptable for the portal. False if no</i> bool
Example		https://MyWebSite/api/PortalCommunication/ValidateUserRegistration

Relative Url		/api/PortalCommunication/DeleteUser
Description		This needs to indicate that the user account in NoPass™ was deleted
Request	Content type	application/json-patch+json
	Body	<pre>{ // Login of the user which is deleted "userId": "string", // Id of the portal from where this user was deleted "portalId": "string", }</pre>
Response	Status	200 in case of all is ok. Other in case of something wrong
	Content type	application/json
	Body	No answer needed
Example		https://MyWebSite/api/PortalCommunication/DeleteUser

Relative Url		/api/PortalCommunication/UpdateUser
Description		This needs to verify possibility of update user's information
Request	Content type	application/json-patch+json
	Body	<pre> { // Login of the user which is deleted "userId": "string", // Id of the portal from where this user was deleted "portalId": "string", // Collection of fields and the actions which needs to be done related to it. If no // actions are required exact action, please skip it "updates": { // The name of field "GivenName": { // new value of the field. Null if no changes required for the value "newValue": "string", // true if user wants to forbid to store information of this field on NoPass™ side. // false if user wants to allow to store information of this field on NoPass™ side. // null if user do not want to change existing behavior related to forbid/allow // store information "forbiddenStore": true }, "SurName": { "newValue": "string", "forbiddenStore": true }, "PhoneNumber": { "newValue": "string", "forbiddenStore": true }, "Email": { "newValue": "string", "forbiddenStore": true }, "ProfileImageUrl": { "newValue": "string", "forbiddenStore": true }, "Locale": { "newValue": "string", "forbiddenStore": true }, "Login": { "newValue": "string", "forbiddenStore": true } } } </pre>

		}
Response	Status	200 in case of all is ok. Other in case of something wrong
	Content type	application/json
	Body	<i>// null in case of no error or constraints from portal side. Error message in case // of something is wrong</i> "string"
	Example	https://MyWebSite/api/PortalCommunication/DeleteUser

3. STEPS FOR USER AUTHENTICATION VIA NoPASS™

When the user chooses authentication via the Identite, the portal should do preform steps describe below.

Connection scheme should be https.

NoPass™_server_api is the NoPass™ system URL:

For example, https://NoPass™.company.com

1. When the user chooses authentication via the Identite, the portal must send a POST request to the Identite server to start the authentication. Response on this request will be an authentication id, which is needed later for correlating authentication information with the user.

URL		NoPass™_server_api/api/UserAuthentication/RequestAuthorization
Description		There is request about authorization user.
Request	Content type	application/json-patch+json
	Authentication	Bearer %AuthToken% <i>//authToken is received from server during registration (see description for /api/PortalCommunication/ConfirmRegistration endpoint for details)</i>
	Body	{ <i>//parameter with length <= 256.</i> <i>// This value should be received and stored after portal registration complete successfully.</i> <i>// A unique identification number assigned to an individual portal</i> <i>// by the authentication server.</i> "portalId": "string", <i>// parameter with length <= 36 and case sensitive.</i> <i>// This is the user's login on the portal (web-site).</i> "userId": "string", <i>// This parameter is not required. It is defines which type of social network should be used to authenticate user against.</i> <i>// 0 = Google, 1 = Facebook</i> "social": int }
Response	Status	200 in case of all is ok. Other in case of something wrong
	Content type	text/plain
	Body	{ <i>//List of errors. Empty if there is no errors.</i> "errors": [

		<pre> { //Code of error. "code": "string", //Message about what's wrong. "message": "string" }], "result": { // parameter to identification the authentication session. // Correlate this value with this login session for this user and remember. // Information about the current authentication session provided by the server. "authId": "string", // parameter with PNG file 8-bit (PNG-8) that is coded in base64 format. // This image should be showed on a web-page for user to // user can compare it with picture on his (her) phone. "image": "string", // It is the time period in milliseconds, where the picture will be changed. // Can be used to show count down indicator. "nextChange": 0, // Link to be used to authenticate user against social network "loginUrl": "string" } } </pre>
Example	https://NoPass™.company.com/api/UserAuthentication/RequestAuthorization	

2. Update login session information in-time after receiving updated data:
 - 2.1 The Idente system calls **PortalUrl/api/PortalCommunication/UpdatePicture** on your site with new information.
You can use web sockets to inform user in-time.
 - 2.2 The Idente system calls **PortalUrl/api/PortalCommunication/AuthorizedUser** on your site to inform about finalized authorization.
3. If image expiration time outs and no information about finalized authorization is received, user should be informed about it. You can inform the user immediately or set some time, which you can wait for information to arrive (updated picture or authentication status) from the Idente system.

4. STEPS FOR USER REGISTRATION ON NoPass™

When the user chooses registration on the Identite system, the portal should perform the steps described below.

Connection type should be https.

NoPass™_server_api is the NoPass™ system URL:

For example, https://NoPass™.company.com

When the user chooses registration via the NoPass™ system, the portal have to send POST request to NoPass™ server for starting. Response on this request will provide link that directs to the installation of the application and OTP for validate the registration in future.

URL		NoPass™_server_api/api/UserRegistration/PreRegisterUser
Description		There is request about registration user.
Request	Content type	application/json-patch+json
	Authentication	Bearer %AuthToken% <i>//authToken is received from server during registration (see description for /api/PortalCommunication/ConfirmRegistration endpoint for details)</i>
	Body	<pre> { //parameter with length <= 256. // This value should be received and stored after portal registration complete // successfully. // A unique identification number assigned to an individual portal // by the authentication server. "portalId": "string", // parameter with length <= 36 and case sensitive. // The login of the user, will be used for logging in. // Should be empty in case of registration is going via social network "userId": "string", // It is remote IP address of client machine. "clientIP": "string", // Absolute url where the user should be redirected // back after registration in NoPass™ will be completed. "redirectUrl": "string", // The name of social network from which the registration may proceed. Google // only now supported // Should be empty in case of registration without social network "socialNetwork": "string", </pre>

		<pre>// Additional information about user. Skip in case of via social network registration is chosen "data": { // First name of the registering account. "givenName": "string", // Surname of the registering account. "surName": "string", // Phone number of the registering account. "phoneNumber": "string", // Email of the registering account. "email": "string", // Url to the picture for the registering account. "profileImageUrl": "string", // locale of the registering account. "locale": "string" }</pre>
Response	Status	200 in case of all is ok. Other in case of something
	Content type	text/plain
	Body	<pre>{ //List of errors. Empty if there is no errors. "errors": [{ //Code of error. "code": "string", //Message about what's wrong. "message": "string" }], "result": { // Secure random generated number used to validate the registration. // Upon registration the portal will be asked to put this registration code in some of the requests "otp": "string", // parameter with length <= 2048. // Link that directs to the installation of the application. // The user should be redirected to this link "registerLink": "string" } }</pre>
	Example	https://NoPass™.company.com/api/UserRegistration/PreRegisterUser

5. STEPS FOR DELETE USER IN NoPASS™

When the user chooses delete account on the Identite system, the portal should perform the steps described below.

Connection type should be https.

NoPass™_server_api is the NoPass™ system URL:

For example, <https://NoPass™.company.com>

When the user chooses delete account action in the NoPass™ system, the portal have to send POST request to NoPass™ server for starting.

URL		NoPass™_server_api/api/UserRegistration/DeleteInitialPortal
Description		There is request about delete account in NoPass™.
Request	Content type	application/json-patch+json
	Authentication	Bearer %AuthToken% <i>//authToken is received from server during registration (see description for /api/PortalCommunication/ConfirmRegistration endpoint for details)</i>
	Body	<pre>{ //parameter with length <= 256. // This value should be received and stored after portal registration complete successfully. // A unique identification number assigned to an individual portal // by the authentication server. "portalId": "string", // parameter with length <= 36 and case sensitive. // The login of the user, will be used for logging in. "userId": "string" }</pre>
Response	Status	200 in case of all is ok. Other in case of something
	Content type	text/plain
	Body	<pre>{ //List of errors. Empty if there is no errors. "errors": [{ //Code of error. "code": "string", //Message about what's wrong. "message": "string" }], //Id of deletion session. "result": "string" }</pre>

Example	https://NoPass™.company.com/api/UserRegistration/DeleteInitialPortal
----------------	---

6. STEPS FOR UPDATE USER'S INFORMATION IN NoPass™

When the user chooses update any of information of account on the Identite system, the portal should perform the steps described below.

Connection type should be https.

NoPass™_server_api is the NoPass™ system URL:

For example, <https://NoPass™.company.com>

When the user chooses update account action in the NoPass™ system, the portal have to send POST request to NoPass™ server for starting.

URL		NoPass™_server_api/api/UserRegistration/UpdateInitialPortal
Description		There is request about update account in NoPass™.
Request	Content type	application/json-patch+json
	Authentication	Bearer %AuthToken% <i>//authToken is received from server during registration (see description for /api/PortalCommunication/ConfirmRegistration endpoint for details)</i>
	Body	<pre>{ // parameter with length <= 36 and case sensitive. // The login of the user, will be used for logging in. "userId": "string", //parameter with length <= 256. // This value should be received and stored after portal registration complete // successfully. // A unique identification number assigned to an individual portal // by the authentication server. "portalId": "string", // Collection of fields and the actions which needs to be done related to it. If no // actions are required exact action, please skip it "updates": { // The name of field "GivenName": { // new value of the field. Null if no changes required for the value "newValue": "string", // true if user wants to forbid to store information of this field on NoPass™ side. // false if user wants to allow to store information of this field on NoPass™ side. // null if user do not want to change existing behavior related to forbid/allow // store information "forbiddenStore": true }, "SurName": { "newValue": "string", "forbiddenStore": true } } }</pre>

		<pre> "PhoneNumber": { "newValue": "string", "forbiddenStore": true }, "Email": { "newValue": "string", "forbiddenStore": true }, "ProfileImageUrl": { "newValue": "string", "forbiddenStore": true }, "Locale": { "newValue": "string", "forbiddenStore": true }, >Login": { "newValue": "string", "forbiddenStore": true } } </pre>
Response	Status	200 in case of all is ok. Other in case of something
	Content type	text/plain
	Body	<pre> { //List of errors. Empty if there is no errors. "errors": [{ //Code of error. "code": "string", //Message about what's wrong. "message": "string" }], //Id of updating session. "result": "string" } </pre>
	Example	https://NoPass™.company.com/api/UserRegistration/UpdateInitialPortal