



PRSONAS JavaScript SDK

v1.2.131

PRSONAS



This document describes the SDK that can be utilized by external experiences to interact with the PRSONAS system.

Dependencies and file loading order

The PRSONAS JavaScript SDK is designed to run exclusively on the PRSONAS Viewer application (PRSONAS_VIEWER.exe). Currently the SDK is made up of 3 modules: SDK, App and Avatar. The SDK module provides the base functionality that all the other modules depend upon and must be loaded first. The App module must be loaded BEFORE the Avatar module since the latter is dependent on it. A typical HTML `<script>` include sequence would look similar to this:

```
<script src="prsonas-sdk.js"></script>
<script src="prsonas-app.js"></script>
<script src="prsonas-avatar.js"></script>
```

The SDK is coded in plain vanilla JavaScript and does not require any external dependencies.

Globals

When the HTML page is loaded there will be a global variable for each of the modules that have been included. These variables are fully qualified with the “com.prsonas” namespace:

com.prsonas.sdk – SDK module
com.prsonas.sdk.app – App module
com.prsonas.sdk.avatar – Avatar module.

Here is a jQuery example of using these variables:

```
$(function(){
    let avatar = com.prsonas.sdk.avatar;
    let languages = api.avatar.getSupportedLanguages().data;
    for(let index = 0; index < languages.length; index++) {
        console.log(languages[index]);
    }
})
```



```
});
```

Events

The PRSONAS SDK utilizes the publish-subscribe pattern for handling events. Clients can subscribe to an event with a method that will be called when an event is raised. Events are module specific and are defined in the “eventTypes” property. Below is an example of registering a callback that will be invoked when the Avatar recognizes a spoken phrase via the microphone:

```
function handlePhrase(e){  
    if (e === null) return;  
    console.log("Phrase recognized:", e);  
}  
  
com.prsonas.sdk.subscribe(com.prsonas.sdk.avatar.eventTypes.PHRASE_RECOGNIZED,  
handlePhrase);
```

An even can also be unsubscribed from using the unsubscribe method:

```
com.prsonas.sdk.unsubscribe(com.prsonas.sdk.avatar.eventTypes.PHRASE_RECOGNIZED,  
handlePhrase);
```

Data passed into event handlers have the following structure:

```
{  
    "module": "<the module raising event>",  
    "type": "<the name of the event>",  
    "value": "<the overall value of the operation>",  
    "properties": "<an object containing additional data for the event>"  
}
```

For example, an event object for a recognized phrase will look like this:



```
{
  "module": "AVATAR",
  "type": "PHRASE-RECOGNIZED",
  "value": "Where can I park my car?",
  "properties": {
    "ANSWER": {
      "text": "Parking",
      "metadata": {
        "system_metadata_qna_edited_manually": "true",
        "action": "location_parking",
        "confidence": "0.9153"
      }
    }
  }
}
```

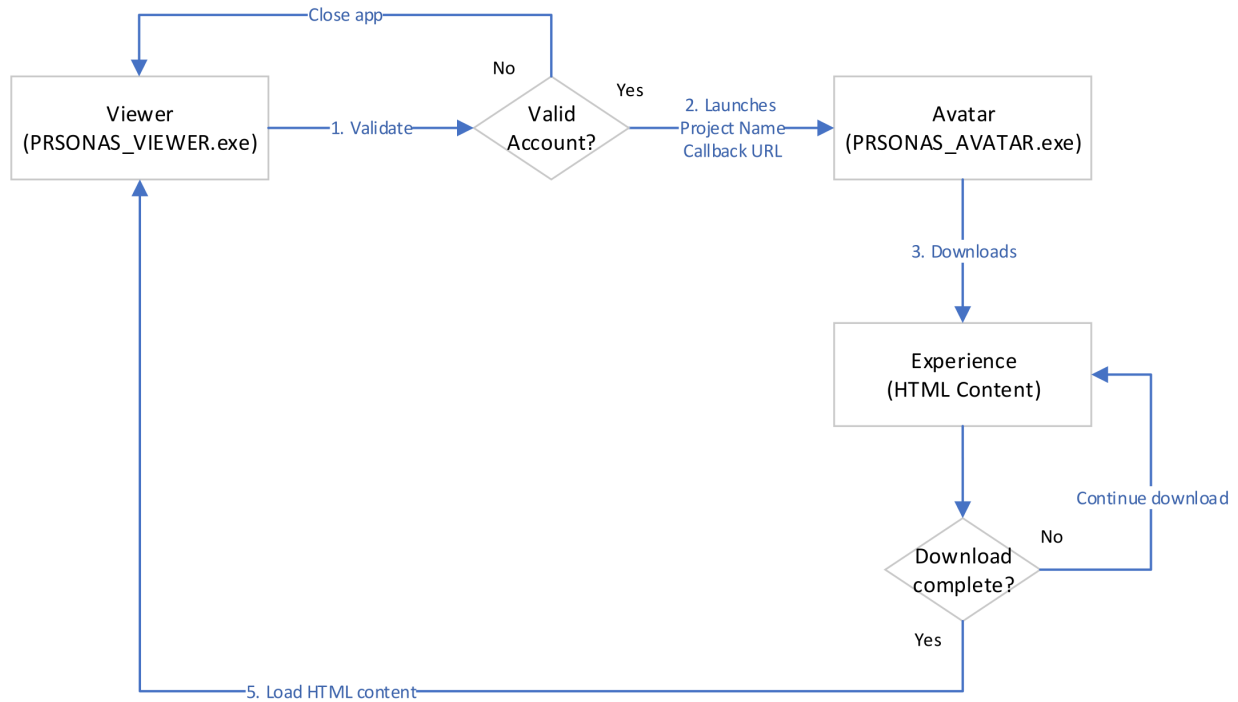
Note: Events are handled asynchronously on separate threads. Be sure to reference the global entry points (i.e. `com.prsonas.sdk`, `com.prsonas.avatar` etc.) in the callback handler.

PRSONAS Startup Sequence

A PRSONAS install is made of up of three components:

- **PRSONAS_AVATAR.exe** – This application displays the PRSONAS character (Avatar) and responds to events sent to it from the Viewer application.
- **PRSONAS_VIEWER.exe** – The Viewer is responsible for displaying the project specific content (Experience) and facilitates communication between the Avatar and the HTML based Experience. The Viewer supports transparency and is most frequently used as an overlay over the Avatar.
- **Experience** – The Experience is the HTML based content displayed in the Viewer application. The Experience is the primary integration point for PRSONAS clients.

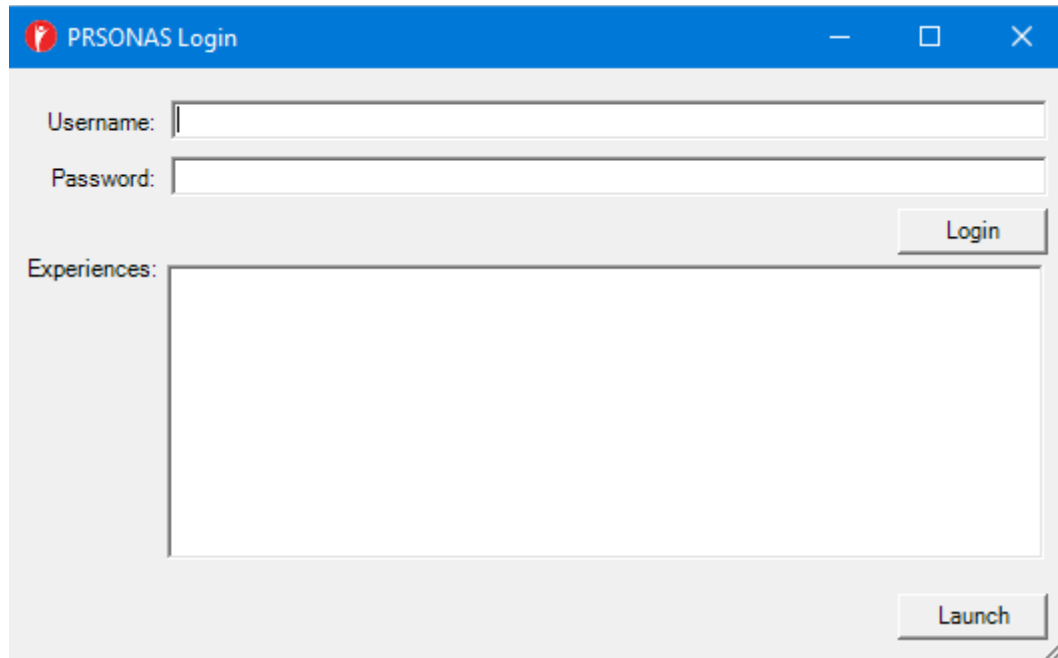
Below is a process diagram that illustrates the startup sequence for a PRSONAS installation:



The PRSONAS_VIEWER.exe (Viewer) is the first application that is started and as such requires a valid username, password and experience supplied by PRSONAS. During startup, the Viewer looks for the account username, password and experience in the following order:

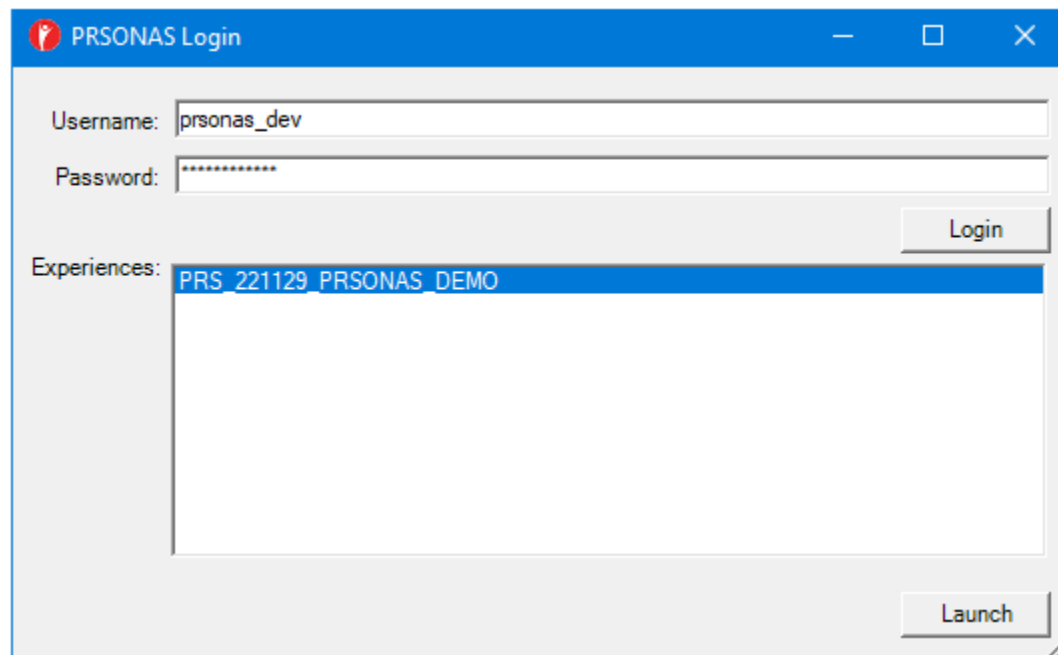
1. Username, password and experience command line parameters.
2. Username, password and experience properties set in the PRSONAS_VIEWER.exe.config file. (NOT RECOMMENDED).
3. _prsonas_username, _prsonas_password and _prsonas_experience environment variables. (Preferred means of storing credentials).
4. Cached credentials saved from the interactive login screen.

If the Viewer does not find credentials in the above locations it will present a login screen:



The screenshot shows the PRSONAS Login window. It has a blue title bar with the PRSONAS logo and the text "PRSONAS Login". Below the title bar, there are two input fields: "Username:" and "Password:". To the right of the "Password:" field is a "Login" button. Below these fields is a large empty box labeled "Experiences:". At the bottom right of the window is a "Launch" button.

Enter the username and password in the supplied fields and click on "Login". If the credentials are valid then a list of available experiences will be displayed in the "Experiences" list box:



The screenshot shows the PRSONAS Login window after successful login. The "Username:" field now contains "prsonas_dev" and the "Password:" field contains a masked password "*****". The "Experiences:" list box now contains one item: "PRS_221129_PRSONAS_DEMO", which is highlighted with a blue background. The "Login" and "Launch" buttons remain visible.

Selecting an experience and clicking the "Launch" button will initiate the loading sequence. Credentials will be stored in an encrypted cache and will automatically be loaded the next time the Viewer is executed so the login screen is only displayed once. To reset credentials



use set the clearAuthCache command line parameter to Y. See the “Viewer Command Line Options” section below for more details.

Viewer Command Line Options

The PRSONAS_VIEWER.exe supports the following command line parameters:

top=<screen position in pixels> - The Y coordinate of the top left corner of the Viewer display.

left=<screen position in pixels> - The X coordinate of the top left corner of the Viewer display.

width=<screen dimension in pixels> - The width of the Viewer display.

height=<screen dimension in pixels> - The height of the Viewer display.

forcefocus=<Y|N> - If set to 'N' the Viewer will not attempt to capture focus. Default is 'Y'.

minimized=<Y|N> - If set to 'Y' the Viewer window will be minimized on startup. Default is 'N'.

url=<URL> - The URL of the size the browser will display.

webroot=<Local drive path> - The root folder for the internal web server. Required if the Viewer is hosting the content instead of an external web server.

clearAuthCache=<Y|N> - Set to 'Y' to clear the cached credentials and display the login screen. Commonly used to switch experiences on an installation.

username=<username provided by PRSONAS>

password=<password provided by PRSONAS>

experience=<experience name provided by PRSONAS>

SDK Class Hierarchy

The JavaScript class hierarchy is available in jsDoc format. See the [documentation/index.html](#) file for more details.