

Sample Application Tutorial

Sample application demonstrates the following features:

1. Login through Captivate Prime Authentication
2. Getting Refresh Token and Access Token
3. Retrieving courses using access token
4. Playing course in embeddable player
5. Closing embeddable player

1.i. Pre-Requisite to run the sample application:

1.i.i Installing WAMP and hosting Sample Application:

To run the Sample Application, the folder (PrimeApp) containing Sample Application source files needs to be hosted (**Place this folder inside [www](#) directory of your WAMP installation directory**). To run it locally, any localhost servers viz. WAMP or XAMPP or any other http servers can be used. Here we are using WAMP as a local server.

Install WAMP and make sure apache, PHP and mysql servers are running.
App is suitable in chrome browser.

1.i.ii. Update Constants Object field in constants.js file:

Check constants.js file of the Sample Application where variable “Constants” is defined.

```
var Constants =
{
    THRESHOLD_BUFFER_TIME_MS : (12 * 60 * 60 * 1000) ,

    //update APP_SECRET field
    APP_SECRET : "bf908d1f-6a90-4fd3-8c17-a04840bba1d6",
    //update this field
    APP_ID : "5595bd7e-d5b7-46f4-a631-af39991d12c8",
    PRIME_SERVER_ADDR : "https://captivateprime.adobe.com/",

    MY_APP_HOSTED_SERVER_BASE_ADDR :
    "http://localhost:8085/", //update this field

    SCOPE : "learner:read,learner:write",
    OCODE_DATA : "statel",
    SOURCE_FOLDER : "PrimeApp/",
    MAIN_PAGE : "oauthredirect.html",
    START_PAGE : "index.html",
    API_VERSION: "primeapi/v2/",
    DEFAULT_LO_TYPE : "course"
};
```

1.i.iii. Create an Application in Integration Admin App:

➔ Create an application in Integration Admin App.

The following points are to be noted as the application is going to be hosted in local server for the purpose of demonstration:

- **Redirect_uri** domain : <http://localhost:8085>
**The port is 8085 which should also be the port on the local machine where WAMP server is running.*
- **Learner possible Scope** : **learner:read,learner:write**

1.ii. Running Sample Application:

Once the server is up, open <http://localhost:8085/index.html> in your chrome Browser.

Enter your email id linked to your Captivate Prime account and select the account of your preference in the next step.

Logging in to your captivate prime account by entering credentials will sign in the user into the Sample application.

1.iii. Sample Application - Code Walkthrough

A. Login through Captivate Prime Authentication

Login through captivate prime authentication requires client id, redirect_uri, accountId and email Id as demonstrated in the sample application. This will redirect to the Captivate Prime Login Page.

```
function redirecttoCPLogin(index)
{

    let ac = accounts[index];

    window.location =

    `https://captivateprime.adobe.com/oauth/o/authorize
    client_id= <application_id>
    &redirect_uri=<redirect_uri>
    &state=<dummy_data>
    &scope=learner:read,learner:write
    &response_type=CODE
    &account=account.id
    &email=emailid`

};
```

After successful login, get the CODE from the redirect url.

B. Getting Refresh Token and Access Token

```
//Getting refresh token
function getRefreshTokenAndFetchData()
{
    var code = getUrlParameter('code');
    var data = {};
    //code received after redirection from the server
    data.code = code;
    //your application secret
    data.client_id = Constants.APP_ID; //your application id
    data.client_secret = Constants.APP_SECRET;
    $.post({
        type: "POST",
        url: "https://captivateprime.adobe.com/oauth/token"
        data: data,
        success: function(data){
            invokeAPIs(); //retrieve resources after successful
            //invocation of refresh and access token
        }
    })
    .fail(function(response) {
        log('Error: ' + response.responseText);
    });
}

//Getting access token
function getAccessTokenAndFetchData()
{
    var data ={};
    data.refresh_token = CLIENT_CACHE.getRefreshToken();
    data.client_id = Constants.APP_ID; //your application id
    data.client_secret = Constants.APP_SECRET; //your
    application secret
    $.post({
        type: "POST",
        url:
        "https://captivateprime.adobe.com/oauth/token/refresh",
        data: data,
        success: function(data){
            invokeAPIs(); //retrieve resources after successful
            //invocation of refresh and access token
        }
    })
    .fail(function(response) {
        console.log("Exception caught while fetching access
        token");
    });
}

}
```

Here data contains OAuth details in the format of

```
{
  "access_token": "95b93757d41d945109d73a4782804d99",
  "refresh_token": "c64ef82e15b90283d146aa720fab57f9",
  "user_role": "learner",
  "account_id": "1234",
  "user_id": "5764678",
  "expires_in": 604799
}
```

Since we have access_token, we can fetch other resources using invokeApi() method.

C. Retrieving courses using access token

Resources can be fetched using access_token as exemplified in the code below:

```
function getCourses()
{
  $.ajax({
    url:
"https://captivateprime.adobe.com/primeapi/v2/learningObjects?include=enrollment,instances.loResources&filter.loTypes=course",
    crossDomain : true,
    dataType : "json",
    beforeSend: function(xhr) {
      xhr.setRequestHeader('Authorization', 'oauth
        '+access_token);

      xhr.setRequestHeader('Accept', 'application/vnd.api+json');
    },
    success: function(lData){
      //lData contains enrolled courses
      jQuery.each(lData.data,function(index,obj){
        //iterate over data
      });
      jQuery.each(lData.included,function(index,obj){
        //iterate over include data
      }
    );
  },
  error: function (errormessage) {
    console.log("error: " + errormessage);
  }
}
);
}
```

D. Playing course in embeddable player

Using Captivate Prime embeddable player in custom application is quite easy. Please look at the code snippet below for its usage:

```
function playCourseUsingEmbeddablePlayer(courseId, iframeId, width, height)
{
    //Just get the embeddable player url by supplying course id
    and access token as shown below

    var url =
    'https://captivateprime.adobe.com/app/player?lo_id=' +
    courseId + '&access_token='+access_token;

    //courseId here is in the format of "course:123456" i.e. v2
    api learningObject format

    //Assign attributes to your iframe where embeddable player
    needs to be fit

    $(iframeId).attr("height", height);

    $(iframeId).attr("width", width);

    $(iframeId).attr("src", url);

    $(iframeId).attr("style","display:block");
}
```

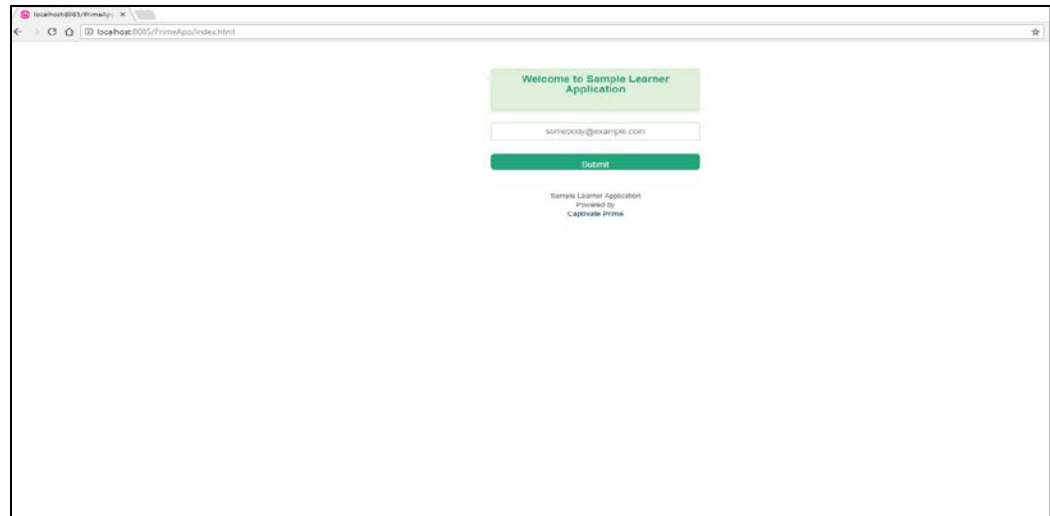
E. Closing embeddable player

Closing an embeddable player is as easy as handling the closing event of iframe. Please see the code snippet below:

```
window.addEventListener("message", function closePlayer(){
    if(event.data === "status:close"){
        $("#samePagePlayerDisplayIframeId").attr('src', 'about:blank');
        $("#samePagePlayerDisplayIframeId").hide();
    }
});
```

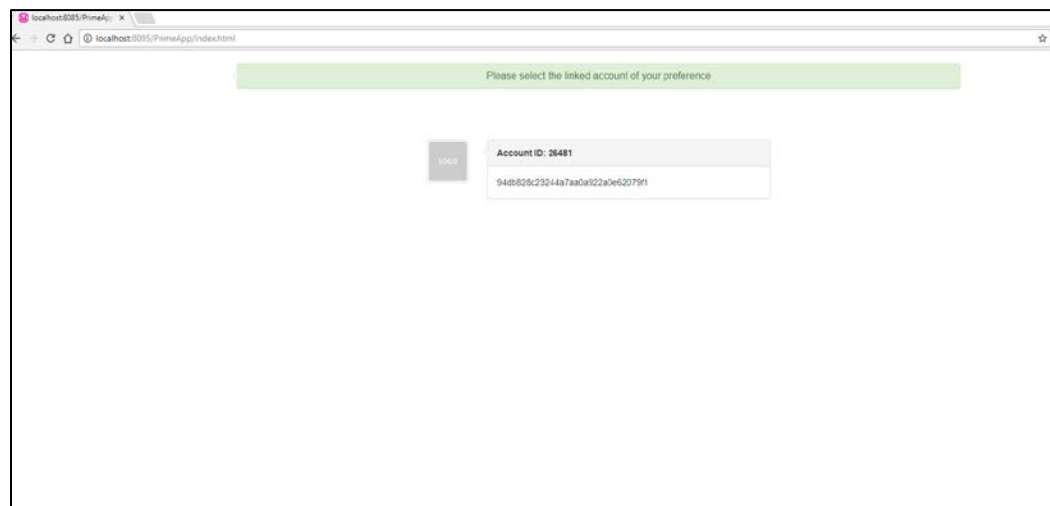
Sample Application Snapshots:

Login Page



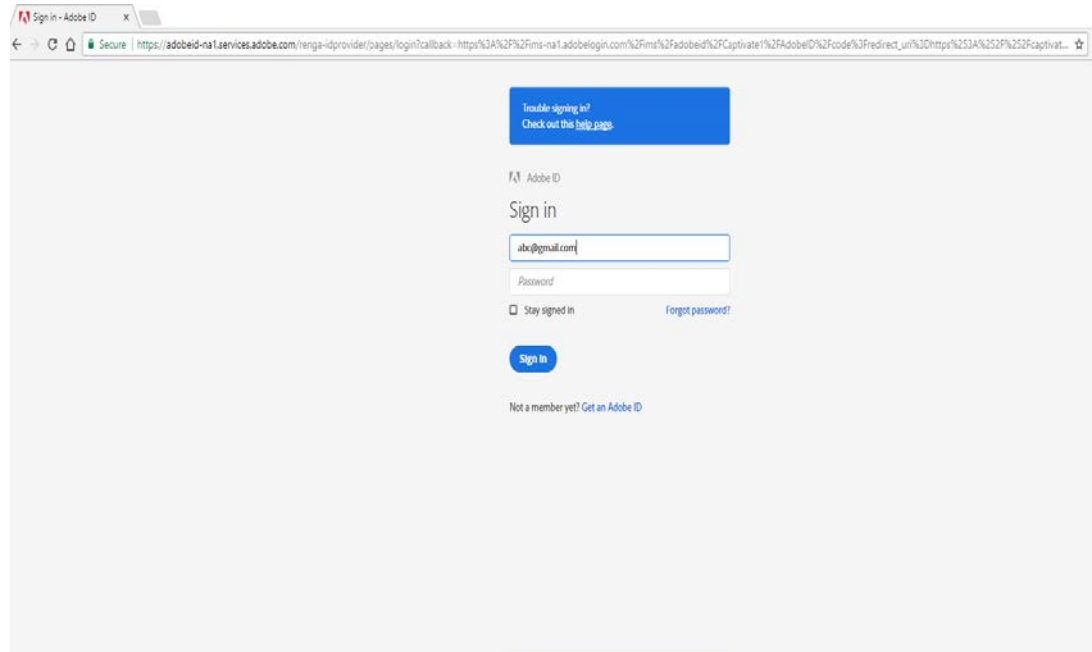
A screenshot of a web browser displaying the login page of the 'Sample Learner Application'. The browser's address bar shows 'localhost:5015/PrimeApp/index.html'. The page features a green header with the text 'Welcome to Sample Learner Application'. Below this is a text input field containing the email address 'somebody@example.com'. A green 'Submit' button is positioned below the input field. At the bottom of the page, it says 'Sample Learner Application' and 'Powered by C:\App\Prime'.

Account Selection Page:



A screenshot of a web browser displaying the account selection page of the 'Sample Learner Application'. The browser's address bar shows 'localhost:5015/PrimeApp/index.html'. A green banner at the top of the page reads 'Please select the linked account of your preference'. Below this banner, there is a table with two columns. The first column contains a grey button labeled 'Logout'. The second column contains a table with two rows: the first row is 'Account ID: 26481' and the second row is '94b82b23244a7aa0a922a0e62079f1'.

Captivate Prime Authentication Page:



Sample Application Home Page After Authentication:

