API Workflow: Customer Signup and Login for C2S Implementations

Table of Contents

Summary

SessionM provides several ways for a customer to sign up for or log in to a loyalty program on a client site. Signup and login logic can be implemented directly for a client site or indirectly via a social media site. The bottom line is that a customer profile exists on the SessionM Platform and can be retrieved for updates. This workflow documents the steps that enable these processes for a customer-to-server (C2S) implementation.

One of the primary characteristics of the C2S flow is that it begins for an unauthenticated customer performing an action on a mobile app or web site. The action can then elicit a screen or dialog that provides one of two ways for accessing the platform: logging in or signing up. Once the customer is logged in or signed up, the workflow then uses the access token associated with the login or signup to retrieve the customer's profile.

Use Cases

These use cases are typical for customers signing up or logging in via a web site or mobile app. Once signed in or logged in, they provide personal data that can be used to create a customer profile on the SessionM Platform:

- Customer signs up on a client site using an email and password; data becomes inputs for a new customer profile.
- Customer signs up via a social media site; data becomes inputs for a new customer profile.
- Customer logs in on a client site using an email and password or via a social media site.
- Access token or ID is obtained for C2S transactions.
- Customer retrieves and updates their profile with new data using an access token. Subsequent activities for logged in or signed in customers rely on other SessionM APIs such as those that manage campaigns or offers.

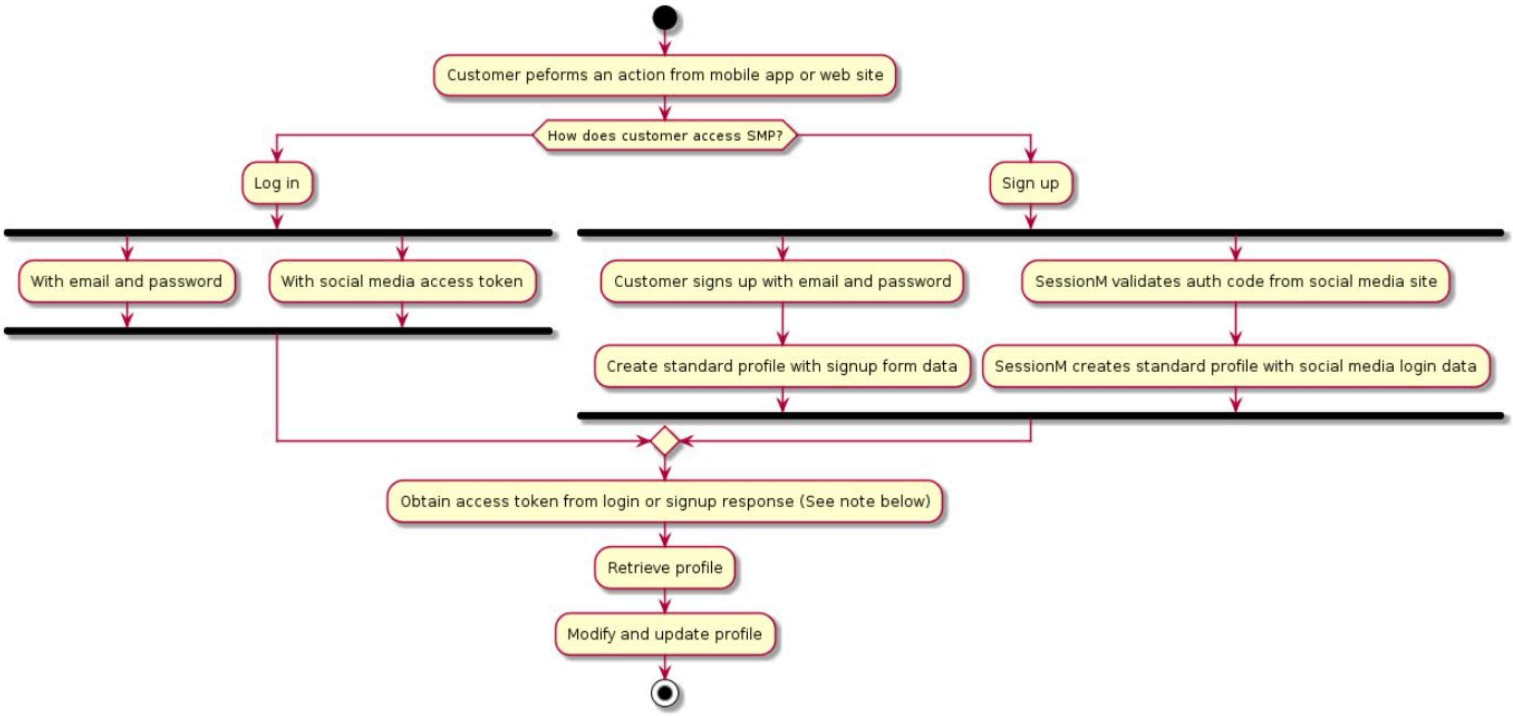
Data Configuration Prerequisites

This workflow presumes the client has performed the following tasks before it can be implemented for both sign-ups and logins:

- Creates and hosts a signup form that customers can access.
  - If signups are through a social media site, the signup form must be integrated with the social login module. This can be done by using the social provider's web SDK.
  - Be sure to update signup form/site when a new social provider is being used, and share the social media site's application ID and credentials with SessionM.
- Sets up server to implement the SessionM C2S APIs.

Step by Step

The workflow for signing up or logging in a customer follows the step-by-step decision tree diagrammed below. The top part of the workflow diagram details the steps associated with logging in or signing up a customer in a C2S implementation:


Customer peforms an action from mobile app or web site
How does customer access SMP?
Log in
Sign up
With email and password
With social media access token
Customer signs up with email and password
SessionM validates auth code from social media site
Create standard profile with signup form data
SessionM creates standard profile with social media login data
Obtain access token from login or signup response (See note below)
Retrieve profile
Modify and update profile

Note: After a customer has signed in, they can opt to log in as captured in the "Log in" step shown in the top diagram.

The sections that follow reflect the decisions and actions detailed in this diagram. Note that throughout this workflow external IDs are featured in the specification of endpoints, but internal IDs can also be used.

When issuing curl commands for platform transactions, adhere to the following syntax:

- Specify: `-H 'Content-Type: application/json' -H 'authorization: Bearer ACCESS_TOKEN'`
- Begin URL with same endpoint + API key: `https://[ENDPOINT]/api/v1/apps/API_KEY`

Step 1: Determine How the Customer Wants to Access the SMP

This workflow begins with determining how the customer plans to access the SessionM Platform.

Since they do not have a profile, they can sign up two different ways:

- Directly, by submitting email and password via a signup form. Proceed to step 2.
- Indirectly, by logging in to a social media site. Proceed to step 3.

Step 2: Create a Customer Profile with Signup Data

If you are following the flow that supports a customer signing up for a customer profile, you can build the signup logic to support two different scenarios.

Creating a Profile with Data from a Social Media Site

You can have SessionM create a customer profile using social media login data. For more information, contact your integration engineer.

Creating a Profile with Data from a Client's Signup Form

You can create a customer profile using signup form data that the customer provided after being authenticated by their email and password. In this step of the workflow a customer accesses a client's UI via a signup button sent to them in an email. The data the customer provides on the signup form becomes a standard customer profile.

For a pure client-side request, specify this endpoint to create a new standard profile with data from the signup form:

```
POST :login_host/:app_ID/accounts.json
```

In the endpoint, identify the *login_host* and *app_ID* (or digital property) for your environment.

A sample *curl* command follows featuring the *user* request object:

Curl Request

```
curl -X POST \

https://login-sdkexamples.stg-sessionm.com/92db916f622814e3ded22faa47370f030f476dc3/accounts.js
on \
  -H 'Content-Type: application/json' \
  -d '{
    "user": {
        "email": "z17@example.com",
        "password": "Password1",
        "confirm_password": "Password1",
        "first_name": "John",
        "last_name": "Davis",
        "dob": "1999-02-02",
        "zip": "02210",
        "gender": "m"
    },
    "client_id": "cc9bf2e9b2fe047305957ff815db18295f40077f3362ce5d9da1fdd15fdf70d0"
}'
```

In the payload, identify a value for the *client_id* associated with your environment.

Other key attributes in this request are derived from the signup form, including *email* and *password*. Before making requests, make sure this data is configured by a SessionM integration engineer.

The platform returns a response object that contains the newly created profile:

Response

```
{
    "status": "ok",
    "user": {
        "email": "z17@example.com",
        "password": "Password1",
        "dob": "1999-02-02",
        "zip": "02210",
        "last_name": "Davis",
        "first_name": "John",
        "gender": "m",
        "referral": {
            "referral_code": null,
            "source": null
        }
    },
    "oauth": {
        "access_token":
"eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpYXQiOiIyMDE5LTA1LTI0IDE1OjIxOjM4ICswMDAwIiwiZXhwIjoiM
jAxOS0wNi0wNyAxNToyMTozOCArMDAwMCIsImRhdGEiOnsiaWQiOiI5ZmY5YzJjNi03ZTM3LTExZTktOWY0NC0wYzEwZDIz
NWU1NzgifX0.mAeU6WEi4RzBqUfzEx66668-iSsFDtg217Sw5IBYKEjOdyaQ9T0nQnr5i8tCn_1XAUIcCNamjnqtwZQdQkN
v9H5S1cwyRpKTIKPAdaFq-On4DVZwpFUccHLHK1CVNm643Xf6ZjvbwFRNDSgIdca7FQHA-jzvr-7pCD9e0eLUnIVy6cIxHi
VEgPQd2sFwmuhndwljmAuqSdt2huPu1pfiufNz9vfCYuHDfEuGfRR8fFD6wNQ7QXjLnRGpBAN5QSuNDaDH-F1hUWh-QLnqP
49TVOnK6E9tRvBJRKU3Tne85_Xo7ABVxWcFLXFZl3OMTeGRrcS43UhrMeHMJdlCsgtEOw",
        "token_type": "bearer",
        "expires_in": 1209600,
        "refresh_token": "bc30fd0c6c4b1c3875ef4d46d91e703087f94e6a26d8c0e3fc4a3b951c1ea39e",
        "scope": "openid",
        "created_at": 1558711298
    }
}
```

In addition to several personal attributes about the customer, this response includes an access token that can be utilized when you retrieve a customer profile in Step 4.

Step 3: Log in to the Platform

Often a customer needs to login to the platform. Maybe too much time has elapsed since their initial signup; maybe they want to login from a different device or web site. Either way, the access token needs to be re-issued so the customer accessing the platform from a client app or web site needs to be invited to login again.

Once the new access token has been acquired, you can facilitate whatever transactions are required on the customer's behalf. If you are following the flow that supports a customer signing up for a customer profile, you can build the signup logic to support two different scenarios.

Logging in with Social Media Credentials

You can log in using the credentials authenticated via a social media site as well as the associated access token. For more information, contact your integration engineer.

Logging in with Email and Password

You can log in using the email and password credentials the customer provided after being authenticated on the client's site.

For a pure client-side request, specify this endpoint to login:

```
POST :login_host/oauth/token/
```

In the endpoint, identify the *login_host* for your environment.

A sample *curl* command follows:

Curl Request

```
curl -X POST \
  https://login-sdkexamples.stg-sessionm.com/oauth/token/ \
  -H 'Content-Type: application/json' \
  -d '{
    "grant_type": "password",
    "email": "z17@example.com",
    "password": "Password1",
    "client_id": "cc9bf2e9b2fe047305957ff815db18295f40077f3362ce5d9da1fdd15fdf70d0",
    "scope": "openid profile email"
}'
```

In the payload, identify a value for the *client_id* associated with your environment.

Note that the *grant_type* attribute is set to "password" and values for *email* and *password* are provided.

After the endpoint makes the request to login, the platform returns a response object, which is shown below:

Response

```
{
    "access_token":
"eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpYXQiOiIyMDE5LTA1LTI0IDE1OjI0OjE0ICswMDAwIiwiZXhwIjoiM
jAxOS0wNi0wNyAxNToyNDoxNCArMDAwMCIsImRhdGEiOnsiaWQiOiJmY2ViOTdiNi03ZTM3LTExZTktOWU2NC1mNzlhZDIz
NWU1NzgifX0.KA1g_yEYyQDWbGEg4Y1yj-JBj1ydmgCCua8wYs-nDWCnTZPXFG1N-Aku2OksPbuM5iMOJn3ll1-chltXT4J
LLWQs9KQusW-dBmG7p_6DXY7ZCTggUu86HYKzL1NnyOvJL_WfiyHBsZC4VWNCFZOKb86KXA6W7EQsoRaPqUDGkBDkhfX79N
6AXLkNk9eqQJmpE1RjDtRBz6PKDj2t_YS5DmI5zhLvzNmM1aBVdVa8ow7KoJI5mXrsYocGtNYyDAxc0osJ0fCuNI7jLT993
W5Cx7VKa-LxOZHWh7zkwvBp4goaXNL5-DJwg0CCSqTCr4cZwWSp2jMEk0ePt7m0v6RBzg",
    "token_type": "bearer",
    "expires_in": 1209600,
    "refresh_token": "32c1161ef930bbbfa579468c0337188210be4e5aba05a09fcbbd2b11755e91b7",
    "scope": "openid profile email",
    "created_at": 1558711454,
    "id_token":
"eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImtpZCI6IkV6dHRMUy1zZEphd0lneWJIMl9OX3JwZDRXVDBZOURCOVl0R0
hVcVYyUEEifQ.eyJpc3MiOiJodHRwczovL2xvZ2luLXBvY3Muc3RnLXNlc3Npb25tLmNvbSIsInN1YiI6IjlmNTU4MzQ2LT
dlMzctMTFlOS05ODBkLTE5MzY0NjRiNmE5YSIsImF1ZCI6ImNjOWJmMmU5YjJmZTA0NzMwNTk1N2ZmODE1ZGIxODI5NWY0M
DA3N2YzMzYyY2U1ZDlkYTFmZGQxNWZkZjcwZDAiLCJleHAiOjE1NTg3MTE1NzQsImlhdCI6MTU1ODcxMTQ1NCwiYXV0aF90
aW1lIjoxNTU4NzExNDU0fQ.HTcCL5_gpKJ0zddZ14howbkwPETSB9K4H0Q0DtiPmJj4DrP8svoj4ZA78J6--LFmyhIRVITb
sVpLJpbbsDv6qppo_KfKO-2XSDC5KdxVOPy7FivIT8R9zKyq71Y4RrRvl_KZJrQxrHueVKgzs__Hv8QUNhdwezw2dfhFMqB
IlxMncnodAe1hdhP0y26ThbHqmkwLfU8hVjs7YDuA1hKiFwU_EXR_K5TwO0r5RA_SH3Xpd75z2k7aljSd5_X3Cf5Zu-Fc3H
LNsAEJyIXfV4UAEOFeQAN3wsQxsrpF6qyblWiXbBCD9ssOpa8ECenJHGCUH2kb9eQnJuHoA7V6xRC5Ng"
}
```

Note that the response contains the access token, which you can use to retrieve a customer profile in the next step.

Step 4: Retrieve a Customer Profile

Client-to-Server transactions reflect implementations that feature a client application or web site initiating communication with the SessionM Platform. To perform any of these transactions, such as updating a profile, you need to get the access token generated by the system.

Upon login, the access token is provided in the response; so for logged in customers, simply pull the access token from the login response. For more information, see the response featured in Logging in with Email and Password. Once you have the access token, be sure to include it in the header of any transaction request.

The next step in the workflow is retrieving the customer profile so you can update it. You can do this by specifying the following endpoint:

```
GET :api_host/api/v1/apps/:app_key/me
```

In the endpoint, identify the *api_host* and *app_key* for your environment. The endpoint uses the application key to retrieve the profile. Be sure to put the access token in the request header. To that end, here's an example of a full *curl* request:

Curl Request

```
curl -X GET \
https://api-client.stg-sessionm.com/api/v1/apps/fff66d34ee8b234bf6aab01293163472d957163a/me \
-H 'Authorization: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpYXQiOiIyMDE5LTA0LTI0IDE3OjE2OjE2ICswMDAwIiwiZXhwIjoiMj
AxOS0wNS0wOCAxNzoxNjoxNiArMDAwMCIsImRhdGEiOnsiaWQiOiJhYjA5NTM2MC02NmI0LTExZTktOTZlNC0zMjczMmFhO
TU4NGMifX0.KbLMYzUuAq2Qi1PGfASXeciHa-eC7ya25DSVRqbBeT937hAwnRMyUhSwwmhZBYQYyU9x4jSI0FayHZrXyrvM
NDlJ9rYVIZB9fgmjUqRwcOsa7CIa3o2W4k-6rZ1nUNRCJq8xz0JrRBojtX-VIwcS4dZJ7pWrQjiNeY3fqWTr01TFJ-2Hmll
vfG4JLBb7Ate5WGUruDYZzg8JrtjWm0S3hvzLdpX3YZTPMSwoBYLULwe-nYaCQSLY_G8Ap17uUfB1HXNr1qa2smewabf1TI
d7Uxictmkv9pBPwkyR1qeC2ro3eLRdBS8hv5R2f1KS0JI9x1n0WbTnr9z9MTceCrBSjg' \
-H 'cache-control: no-cache'
```

After the endpoint makes the request for the customer profile, the platform returns a response object, which is shown below:

Response

```
{
      "status": "ok",
      "user": {
            "id": "e76fc440-d7f0-11e8-91e4-469b02598280",
            "external_id": "88888888",
            "opted_in": true,
            "activated": false,
            "proxy_ids": [],
            "email": "zzz@example.com",
            "gender": "m",
            "dob": "1980-01-01",
            "created_at": "2018-10-25 00:57:11",
            "updated_at": "2018-10-25 00:57:11",
            "address": "7 Tremont Street",
            "city": "Boston",
            "zip": "02021",
            "dma": "506",
            "state": "MA",
            "country": "USA",
            "suspended": false,
            "last_name": "Davis",
            "first_name": "John",
            "registered_at": "2018-10-25 00:57:10",
            "profile_photo_url": "/images/account-neutral.png",
            "test_account": false,
            "account_status": "good",
            "tier_levels": [],
            "referrer_code": "JOHNS-708C6E",
            "phone_numbers": [{
                  "phone_number": "1234123123",
                  "phone_type": "home",
                  "preference_flags": ["primary"],
                  "verified_ownership": false
            }]
      }
}
```

With the profile retrieved, it can be updated. Notice that in this response the first name of the customer is “John.” In the next step, you can modify this attribute.

Step 5: Modify and Update a Customer Profile

The next step in the workflow is modifying and updating the customer profile, as shown with the following endpoint:

```
PUT :api_host/api/v1/apps/:app_key/me
```

In the endpoint, identify the *api_host* and *app_key* for your environment.

A sample *curl* command follows featuring the *user* request object, including the new values for *first_name* and *last_name*:

Curl Request

```
curl -X PUT \
  https://api-client.stg-sessionm.com/api/v1/apps/fff66d34ee8b234bf6aab01293163472d957163a/me \
  -H 'Authorization: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpYXQiOiIyMDE5LTA0LTI0IDE3OjE2OjE2ICswMDAwIiwiZXhwIjoiMj
AxOS0wNS0wOCAxNzoxNjoxNiArMDAwMCIsImRhdGEiOnsiaWQiOiJhYjA5NTM2MC02NmI0LTExZTktOTZlNC0zMjczMmFhO
TU4NGMifX0.KbLMYzUuAq2Qi1PGfASXeciHa-eC7ya25DSVRqbBeT937hAwnRMyUhSwwmhZBYQYyU9x4jSI0FayHZrXyrvM
NDlJ9rYVIZB9fgmjUqRwcOsa7CIa3o2W4k-6rZ1nUNRCJq8xz0JrRBojtX-VIwcS4dZJ7pWrQjiNeY3fqWTr01TFJ-2Hmll
vfG4JLBb7Ate5WGUruDYZzg8JrtjWm0S3hvzLdpX3YZTPMSwoBYLULwe-nYaCQSLY_G8Ap17uUfB1HXNr1qa2smewabf1TI
d7Uxictmkv9pBPwkyR1qeC2ro3eLRdBS8hv5R2f1KS0JI9x1n0WbTnr9z9MTceCrBSjg' \
  -H 'Content-Type: application/json' \
  -H 'cache-control: no-cache' \
  -d '{
    "user": {
        "first_name" : "jeb",
        "last_name": "smith"
    }
}'
```

After the endpoint makes the request for the customer profile update, the platform returns a response object, which is shown below:

Response

```
{
      "status": "ok",
      "user": {
            "id": "e76fc440-d7f0-11e8-91e4-469b02598280",
            "external_id": "88888888",
            "opted_in": true,
            "activated": false,
            "proxy_ids": [],
            "email": "zzz@example.com",
            "gender": "m",
            "dob": "1980-01-01",
            "created_at": "2018-10-25 00:57:11",
            "updated_at": "2018-10-25 00:57:11",
            "address": "7 Tremont Street",
            "city": "Boston",
            "zip": "02021",
            "dma": "506",
            "state": "MA",
            "country": "USA",
            "suspended": false,
            "last_name": "Smith",
            "first_name": "Jeb",
            "registered_at": "2018-10-25 00:57:10",
            "profile_photo_url": "/images/account-neutral.png",
            "test_account": false,
            "account_status": "good",
            "tier_levels": [],
            "referrer_code": "ZZS-B27C94",
            "phone_numbers": [{
                  "phone_number": "1234123123",
                  "phone_type": "home",
                  "preference_flags": ["primary"],
                  "verified_ownership": false
            }]
      }
}
```

The profile has been updated with the customer’s new first and last names, “Jeb” and “Smith.”