

iRewards Control XML – Process Documentation

From your most recent comments, it seems as though you understand the benefits of using Control XML to perform this multi-step Use Case. There are ways to achieve the same outcome, but as you noted, this would require multiple API calls using our REST service or SFTP file upload.

We have documentation about how to get started with Control XML.

From within this starting point, use the left-hand navigation to find more details:

<https://docs.mapp.com/apidocs/soap-cxml>

The key steps to follow will be:

- 1. Produce an XML file which contains the correct structure for your Use Case**
- 2. Test the file**
- 3. Choose between API or SFTP**
- 4. Sending the content using SOAP API**
- 5. Upload the file(s) to an SFTP location & setup an Automation to process the file**

Producing The XML File

Several factors need to be considered when producing the XML. The body content should be structured inside <control> tags and written to the standards for each action.

Information about importing recipients can be found here:

import recipients - <https://docs.mapp.com/apidocs/import-recipients>

Information about sending embedded messages can be found here:

send embedded message - <https://docs.mapp.com/apidocs/send-embedded-message>

Testing The File(s)

Once you are ready, you can test the file inside Mapp Engage by following these steps: **Test Control XML** - <https://docs.mapp.com/apidocs/test-control-xml>

Please refer to Sample Responses for more information about expected responses.

<https://docs.mapp.com/apidocs/sample-responses>

API or SFTP

Once you have the XML ready, you need to consider whether you will be sending it through an API call or uploading it to an SFTP server and configuring a recurring import.

Sending Through SOAP API

The requests you send to the API will need follow a standard format, outlined here: [SOAP API - https://docs.mapp.com/apidocs/soap-api](https://docs.mapp.com/apidocs/soap-api)

The endpoint will be the system URL followed by the following for the eCMessageService, so in umobile's case, it will be:

`https://mail-u.com.my/eCMessageService/rpc`

To Authenticate, you will need to use "umobile" as a system user's Realm and API Details.

```
<SOAP-ENV:Header>
  <authenticate xmlns="http://webservices.ecircle-ag.com/ws">
    <realm>umobile</realm>
    <email>your_email@example.com</email>
    <password>your_password</password>
  </authenticate>
</SOAP-ENV:Header>
An full example could look like this:
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/">
  <SOAP-ENV:Header>
    <authenticate xmlns="http://webservices.ecircle-ag.com/ws">
      <realm>umobile</realm>
      <email>your_email@example.com</email>
      <password>your_password</password>
    </authenticate>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <postMessageRequest xmlns="http://webservices.ecircle-ag.com/ws">
      <control xmlns="http://webservices.ecircle-ag.com/ecm"
        request-id="UniqueID-12345" group-id="your_group_id">
        <message message-id="new">
          <send-date>
            <immediately/>
          </send-date>
          <content target-content-encoding="UTF-8">
            <subject>Your Email Subject</subject>
            <html target-content-encoding="UTF-8">
              <![CDATA[
                <!DOCTYPE html>
                <html lang="en">
                <head>
                  <meta charset="UTF-8">
                  <title>Your Email Title</title>
                  <style>
                    body { font-family: Arial, sans-serif;
padding: 20px; }
                    .header { background-color: #f0f0f0;
padding: 10px; text-align: center; }
                    .content { padding: 20px; }
                    .footer { background-color: #f0f0f0;
padding: 10px; text-align: center; }
                  </style>
                </head>
                <body>
                  <div class="header">
```

```

        <h1>Welcome to Our Newsletter</h1>
    </div>
    <div class="content">
        <p>Hello <${user['FirstName']}>,</p>
        <p>This is a sample embedded email
content.</p>
        
        <p>Check out our <a
href="https://www.example.com">website</a> for more information.</p>
    </div>
    <div class="footer">
        <p>© 2024 Your Company. All rights
reserved.</p>
        <p><a
href="<%unsubscribe_link_text%">Unsubscribe</a></p>
    </div>
</body>
</html>
]]>
</html>
</content>
</message>
<failure-report-address xmlns="http://webservices.ecircle-
ag.com/ecm">
    <email>failure_reports@mail-u.com.my</email>
</failure-report-address>
</control>
</postMessageRequest>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Uploading The File(s) & Setting Up an Automation

To turn this into a recurring action, you must place the relevant file(s) on an SFTP location, which can be accessed through Mapp Engage.

If you are using a .zip file with multiple files for the XML and CSV, then you can find details here:

Automation Example with zipped File - <https://docs.mapp.com/apidocs/automation-example-with-zipped-file>

Alternatively, you can find details on a single file setup here: **Example Automation** - <https://docs.mapp.com/apidocs/example-automation>

To set up an Automation, follow the same steps to test the file (above), but continue setting to add the details required to pull the file from the SFTP Location.

Details can be found here:

Process Control XML - <https://docs.mapp.com/v1/docs/en/system-process-control-xml>