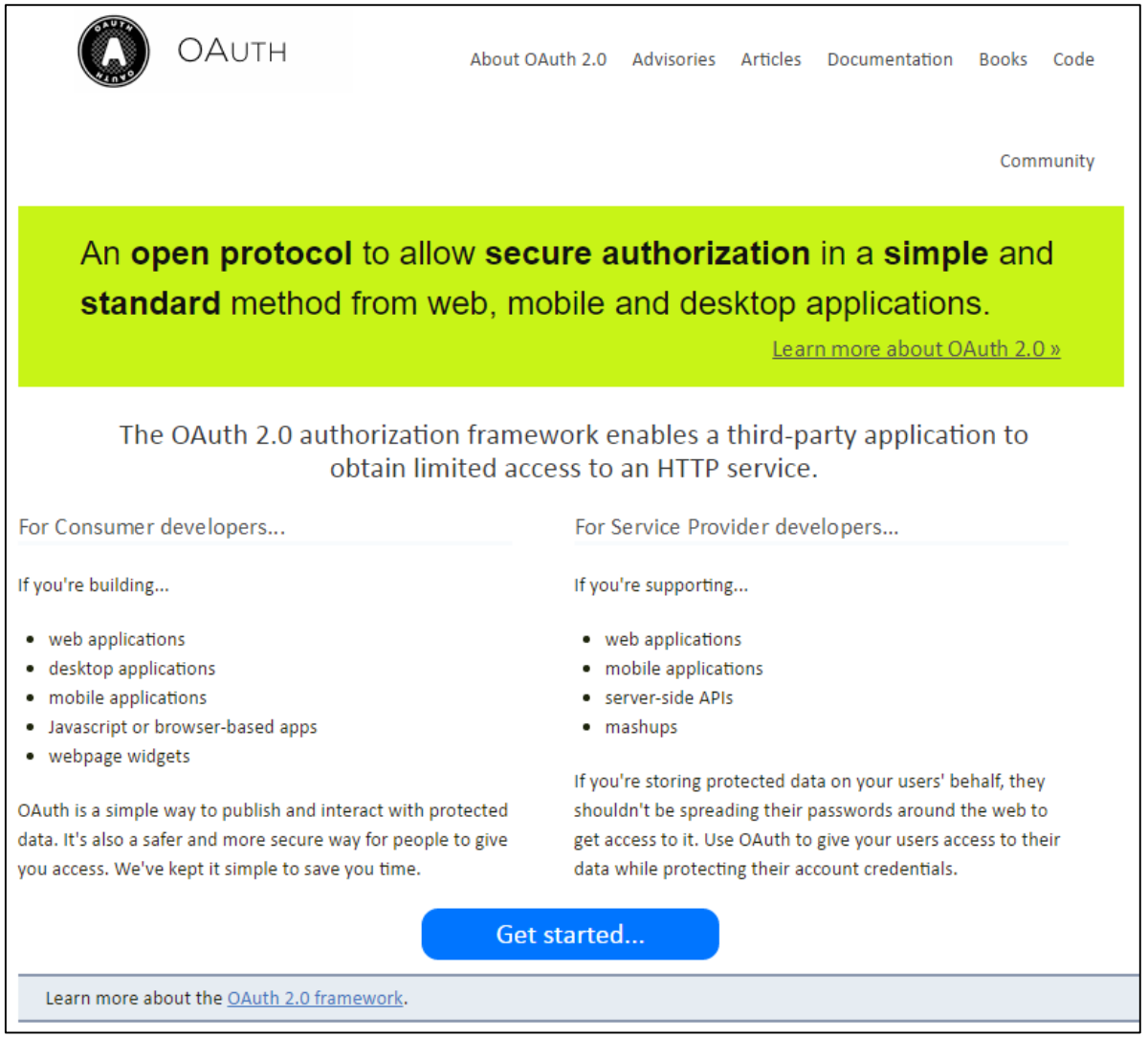
Week 4 Unit 5: **More Engagement with Gamification**

More Engagement with Gamification

OAuth fundamentals

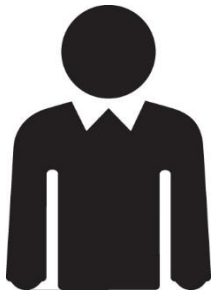
OAuth fundamentals

- Open protocol
- Enables applications to obtain limited access to an HTTP service in a simple and standard method
- Delegates user authentication to the service that hosts the user account
- e.g. Facebook, Twitter,...



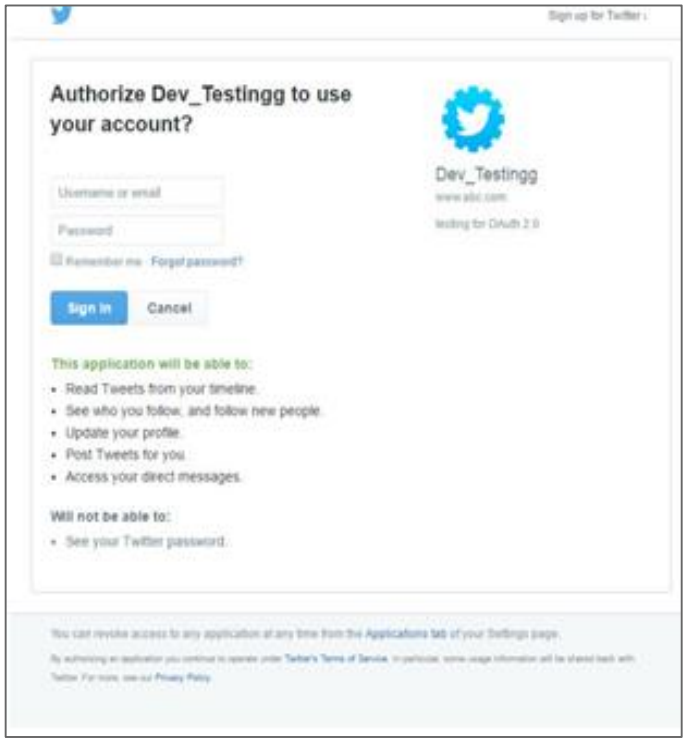
More Engagement with Gamification

Scenario: Twitter integration for the ESPM application

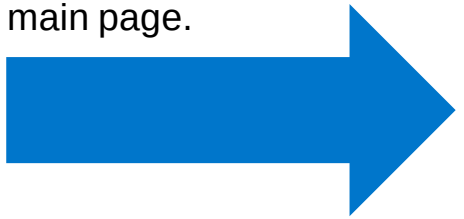


Customer

User enters ESPM application and is redirected to Twitter login page



On authentication, user is redirected back from Twitter to the application with the Twitter CallBack service. The service in turn calls the ESPM main page.

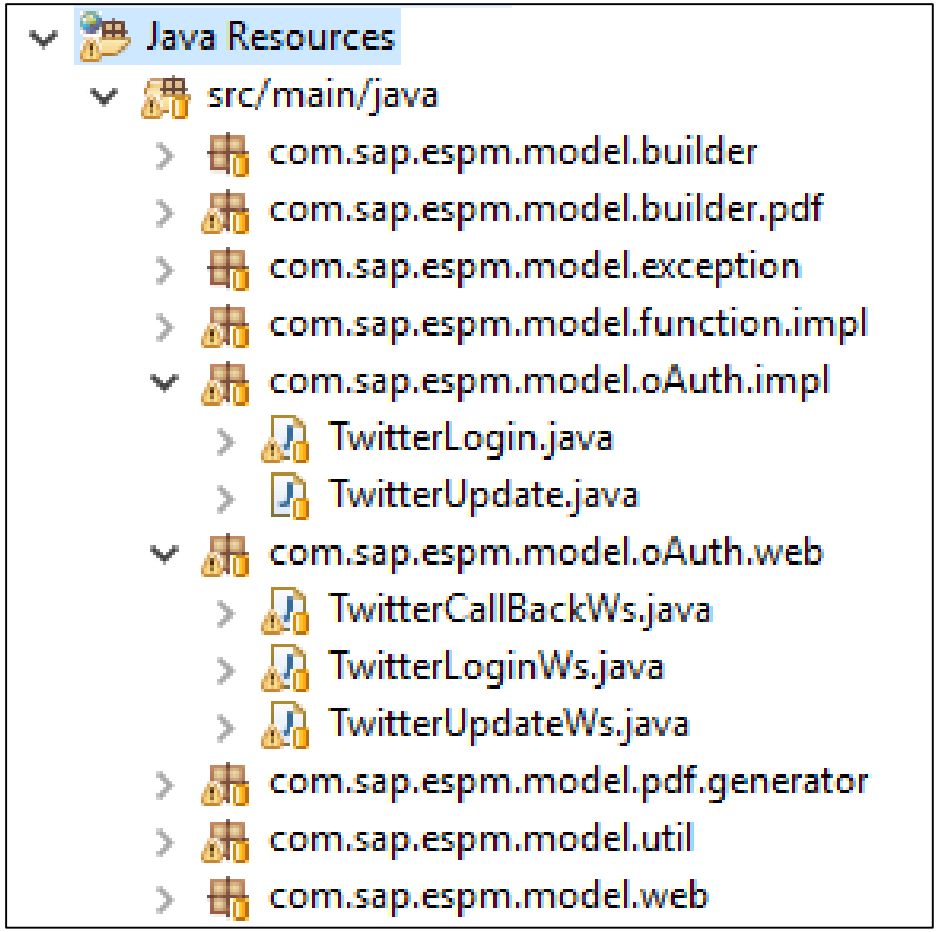


Customer

More Engagement with Gamification
Implementation

Twitter integration for the ESPM application

- Use of the Twitter4j library
- Register ESPM Webshop application and receive “client credentials”
- Create Twitter Login service
- Create Twitter CallBack service
- Create Twitter Update service
- Package under src/main/java: com.sap.espm.model.oAuth



More Engagement with Gamification

Hands-on

Let's tweet a bit

- Checkout Commit of the Twitter Integration branch
- Register your ESPM Webshop application with your Twitter account and receive "client credentials"
- Modify TwitterUpdateWs class statusMessage variable and put in a fixed string, e.g. ". @openSAP @saphcp @svenkohlhaas #developingjavabasedappsonsaphcprocks"
- Build and deploy to the cloud
- Create destination in the cockpit called twitterOauth with your client credentials as properties
  - consumerApplicationKey
  - consumerApplicationSecret
- Log in with your Twitter account, go through the buying process, tweet about it and check your Twitter account

*TwitterUpdateWs.java

```
39    protected void doPost(HttpServletRequest request, HttpServletResponse response
40            throws ServletException, IOException {
41        String body = null;
42        StringBuilder stringBuilder = new StringBuilder();
43        BufferedReader bufferedReader = null;
44        try {
45            InputStream inputStream = request.getInputStream();
46            if (inputStream != null) {
47                bufferedReader = new BufferedReader(new InputStreamReader(inputSt
48                char[] charBuffer = new char[128];
49                int bytesRead = -1;
50                while ((bytesRead = bufferedReader.read(charBuffer)) > 0) {
51                    stringBuilder.append(charBuffer, 0, bytesRead);
52                }
53            } else {
54                stringBuilder.append("");
55            }
56        } catch (IOException ex) {
57            throw ex;
58        } finally {
59            if (bufferedReader != null) {
60                try {
61                    bufferedReader.close();
62                } catch (IOException ex) {
63                    throw ex;
64                }
65            }
66        }
67
68        body = stringBuilder.toString();
69        System.out.println("body");
70        try {
71            JSONObject statusObject = new JSONObject(body);
72            String statusMessage = statusObject.getString("status");
73            System.out.println(statusMessage);
74            TwitterUpdate.tweet(TwitterCallBackWs.accessToken, statusMessage);
75            System.out.println("Successfully updated the status in Twitter.");
76        } catch (twitter4j.JSONException e) {
77            System.out.println("JSON Request Object Error");
78        }
79    }
80 }
```

More Engagement with Gamification

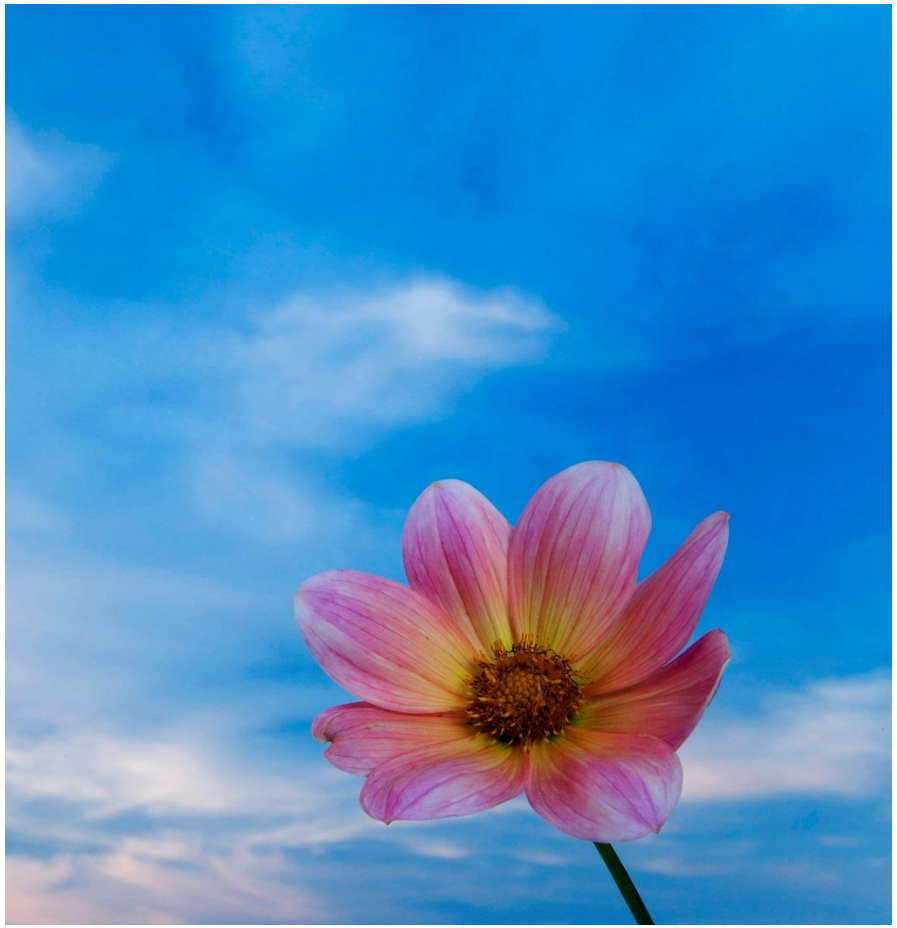
What you've learned in this unit

- What the fundamentals of OAuth are
- How to integrate Twitter in the ESPM application
  - Scenario
  - Implementation
- How to work with the ESPM application when Twitter is integrated

More Engagement with Gamification
Further reading

Additional Material

- OAuth Community Site



Thank you

Contact information:

open@sap.com

openSAP

© 2016 SAP SE or an SAP affiliate company. All rights reserved.