

Practice Exercises - GAM Advanced Course

Requirements:

Generate a KB with GAM applied. The web application must have the built-in security level set to *Authentication*.

Challenge 1: Authorization

This challenge consists in creating a Transaction (Users) in your application with Id and Name attributes, a Web Panel (UserInfo), two Users (TestUser1 and TestUser2), and a Role (Manager).

You have to apply the *Work With for Web* pattern to the transaction. Next, select the *Authorization* value as built-in security level for both the transaction and the automatically created Web Panel.

Once this is done, the application permissions must have the *Restricted* value as default access to the previously generated permission (users_FullControl).

Associate the user TestUser1 with the created role, and associate this role with the permission corresponding to the transaction with full control level.

Finally, check that the Web Panel can be accessed by any of the two users, while the WorkWith of the Users transaction can only be accessed by TestUser1.

Challenge 2: Menu

In your application, define a menu that provides access options to the Transaction and Web Panel, and display it.

Check that user TestUser2 can only view the menu option associated with the Web Panel and that user TestUser1 can view both.

Challenge 3: API

In the Web Panel defined in challenge 1 (UserInfo), the following information should be displayed to the user programmatically by using the API:

- UserGUID
- Username
- Name
- Birthday
- Email
- Is it an active user?
- Activation Date
- Authentication Type

Challenge 4: Events

Subscribe to the events to add, modify, and delete users. When they occur, a log record (with Info level) must be generated with the following information:

- Event name
- Username
- IP address

Optional:

- Check input JSON content.
- Check for possible GAM errors and log them if they exist.

Challenge 5: Mobile

A mobile application will be provided for this challenge. This application will contain items and the option to mark them as favorites. It will also allow you to register and log in with a user account.

The challenge consists in managing an anonymous user.

You will have to simulate the flow of marking one or more items as favorites. Since the anonymous user property is activated, when registering in the application those favorites are associated with the user and are not lost.

Once logged out, the favorites should be cleared from the application because the user's GUID has changed.

To confirm that everything worked correctly, there are two options:

1. From the application website, access the `wwuserproductsfavorites` object and check that the `<GUID, Fav>` pair exists.
2. From the mobile application, once logged out, check that the previously selected favorites are no longer there, and log in again with the registered user. There, the favorites should appear again.

Additional requirements to do the challenge:

- Have the Android SDK installed.
- Create a KB with enabled GAM and Android support.
- Import the xpz file provided.
- Do a Build.
- Load products to the application from the "MainLoadData" Web Panel.

Notes:

If the application design is corrupted, in Customization/Platforms/Any Android Device check if the ECommerce style is set in its properties. If not, set it and do a Rebuild All.

The correct application design is as follows:

