# Pre-requisites

Pre-requisites set-up tutorial https://developers.sap.com/tutorials/appgyver-subscribe-service.html

|  |  |
| --- | --- |
| Descriptin | Screenshot |
| enterprise account (optional, use community edition instead) | https://developers.sap.com/tutorials/appgyver-subscribe-service.html  The community edition is available here: [AppGyver.com](https://www.appgyver.com/) |
| booster (optional, use community edition instead) | https://developers.sap.com/tutorials/appgyver-subscribe-service.html  The community edition is available here: [AppGyver.com](https://www.appgyver.com/) |
| preview app (optional, use the preview link and open it in mobile browser instead) | https://developers.sap.com/tutorials/appgyver-subscribe-service.html |
|  |  |

# Food Barcode Scanner - Scan Barcode to Text

|  |  |
| --- | --- |
| Descriptin | Screenshot |
| Start | |
| Start a new project |  |
| Click the link to use AppGyver Community edition,  Give it name   |  |  | | --- | --- | | Name | Barcode Scanner App |   Click create |  |
| Get to know the interface,  check device iphone (default is what we want so no need to change it) |  |
| Keep the project open on the phone:  Open Launch tab,  You can choose to open preview as webpage,  Or see it in the Preview app on your mobile device, then click here to generate qr code, and use the app on your phone to scan it  The app can be downloaded from app store or google play  You need to save it first, and then see the latest change on your phone |  |
| UI | |
| Delete the default components |  |
| Drag a button, name it “Scan Barcode” |  |
| Drage two “Text” before the button:  first line to show product name, second line to show product calories |  |
| Business rules in logic flow: | |
| Select the button,  Unfold logic canvas,  drag and drop “Device | Scan QR/barcode”,  connect it to “Event | Component tap”, |  |
| drag and drop “Dialog | Alert”,  connect it to “Device | Scan QR/barcode”, (remember to choose the first output port) |  |
| Variable and component binding: | |
| Select “Dialog|Alert”,  Bind variable to Dialog title |  |
| In the wizard, select “Output value of another node” > “Scan QR/barcode” > “QR barcode content” |  |
| Save it and test it on the phone | Pop out: 5020411121182 |
| Test results | Graphical user interface, text, application, chat or text message  Description automatically generated  A picture containing graphical user interface  Description automatically generated |

# Food Barcode Scanner – Call API

|  |  |
| --- | --- |
| Descriptin | Screenshot |
| Check API | |
| Open <https://world.openfoodfacts.org/>  Scroll to the footer and open "Data, API and SDKs", |  |
| notice the URL to read data for a product here, we will make use of it in our app: https://world.openfoodfacts.org/api/v0/product/[barcode].json  be noted that the paramter in url should be in {} curl brackets |  |
| add it as data source | |
| "data" tab > "Create data entity" > "REST API integration" |  |
| fil in info   |  |  | | --- | --- | | resource ID | openfoodfacts | | short description | data from openfoodfacts api | | resource URL | https://world.openfoodfacts.org/api/v0/ | |  |
| select method "Get Record (GET)",  fill in the path: /product/{barcode}.json  set parameter as url placeholder: {barcode} |  |
| test it, using "5020411121182"  check response, find the field we want (ctrl+f to find by key word):   * product\_name * energy-kcal\_100g |  |
| set schema,  check:   * product.product\_name * product.nutriments["energy-kcal\_100g"]   save |  |
| fetching and saving data | |
| go back to "UI canvas" tab, choose scan barcode button  in logic canvas, delete the alert node  add "Data | Get Record" node, link it to its last node |  |
| bind for get record > inputs > barcode |  |
| click output value of another node > scan qr/barcode |  |
| create a data variable:  in the panel of right-top, switch from view to variables,  choose add data variable  select add data, from data resource: openfoodfacts  name it FoodFact  change variable type to single data record |  |
| select FoodFact, in logic canvas, all nodes except the two page event nodes (because the default logic here is to fetch data every 5 s, and we only need to fetch data after button is tapped) |  |
| select scan barcode button, in its logic, add "variables | set data variable",  bind data for its input: output value of another node > get record > record  save |  |
| display the data in two texts | |
| select first text,  edit binding for content > data and variable > data variable > FoodFact > FoodFact.product.product\_name (you can ctrl+f search for the field "product\_name" instead of scroll down the list) > set preview value "food product name" > save |  |
| do the same for second text,  bind to FoodFact.product. nutriments["energy-kcal\_100g"] |  |
| edit the string in formula:  "calories per 100g: " + data. FoodFact.product. nutriments["energy-kcal\_100g"]  check the string to be displayed in preview  save |  |
| you can save the project and check in Preview app on your mobile device:  then you will find the default text is not user friendly (if the data has not been retrieved, the display should be hidden) |  |
| select first text > properties > advanced properties > edit the binding,  binding type > formula >   |  |  | | --- | --- | | formula | data.FoodFact.status == 1 |   save  test on your device |  |

# SAP Sales Cloud Extension - Appointment List App

|  |  |
| --- | --- |
| Descriptin | Screenshot |
| UI | |
| create a project : Appointment List  Page name: Appointment List App  change the Headline component to: Appointments  delete text component |  |
| add Card component, |  |
| open its component template editor  add text component to the card  close the component template editor by double-click outside of the card component (or exist on top-right)  save |  |
| connect to SAP Sales Cloud OData API | |
| verify the API URL by open it in browser:  <https://btp150-proxy.c-2828549.kyma.ondemand.com/sap/c4c/odata/v1/c4codataapi/$metadata> |  |
| in Data tab > add data resource > OData integration > |  |
| basic authentication  copy and paste the url > select AppointmentCollection > switch on all > save data resource  <https://btp150-proxy.c-2828549.kyma.ondemand.com/sap/c4c/odata/v1/c4codataapi/$metadata> |  |
| list > test > run test  the returned data is in JSON format {[,]},  find in text:  - subject  - categorytext  - organizerName  - appointmenttextcollection[0.text] |  |
| see schema  save data resource |  |
| get back to UI canvas |  |
| consume the data | |
| create data variable  data variable type: collection of data records |  |
| add basic authentication:  object with properties > auth type / username / password   |  |  | | --- | --- | | authentication type | basic | | username | Demouser | | password | Welcome1 |   no authentication, no username and password also works |  |
| filter to show only the category of marketing:  filter condition > object with property > add condition (property and condition type are selected from drop-down list)   |  |  |  | | --- | --- | --- | | property | condition type | comparaed value | | CategoryText | equal | Marketing |   (note: the language of your phone should better be set to English! because the CategoryText will be displayed in your language, in my case, iphone,chinese=>"市场营销" )  save |  |
| repeat the card component for the appointment list, and text component for the note list | |
| choose card component > repeat with > data and variables > data variable > select AppointmentCollection1 > save |  |
| bind data to the card  bind title > data item in repeat >   |  |  | | --- | --- | | select repeat | current | | repeat data property | current.Subject | | preview value | Meeting With Customer A |     save |  |
| bind content > data item in repeat >   |  |  | | --- | --- | | select repeat | current | | repeat data property | current.OrganizerName | | preview value | Amy Lee |   save |  |
| bind meeting notes (also repeated items) to card:  click the card template editor icon > select the text component > repeat with > data item in repeat >   |  |  | | --- | --- | | select repeat | current | | repeat data property | current.AppointmentTextCollection | |  |
| bind content > data item in repeat >   |  |  | | --- | --- | | select repeat | current | | repeat data property | current.Text | | preview value | prepare the slides beforehand |   save |  |
| test on the phone | Graphical user interface, text, application, Teams  Description automatically generated |