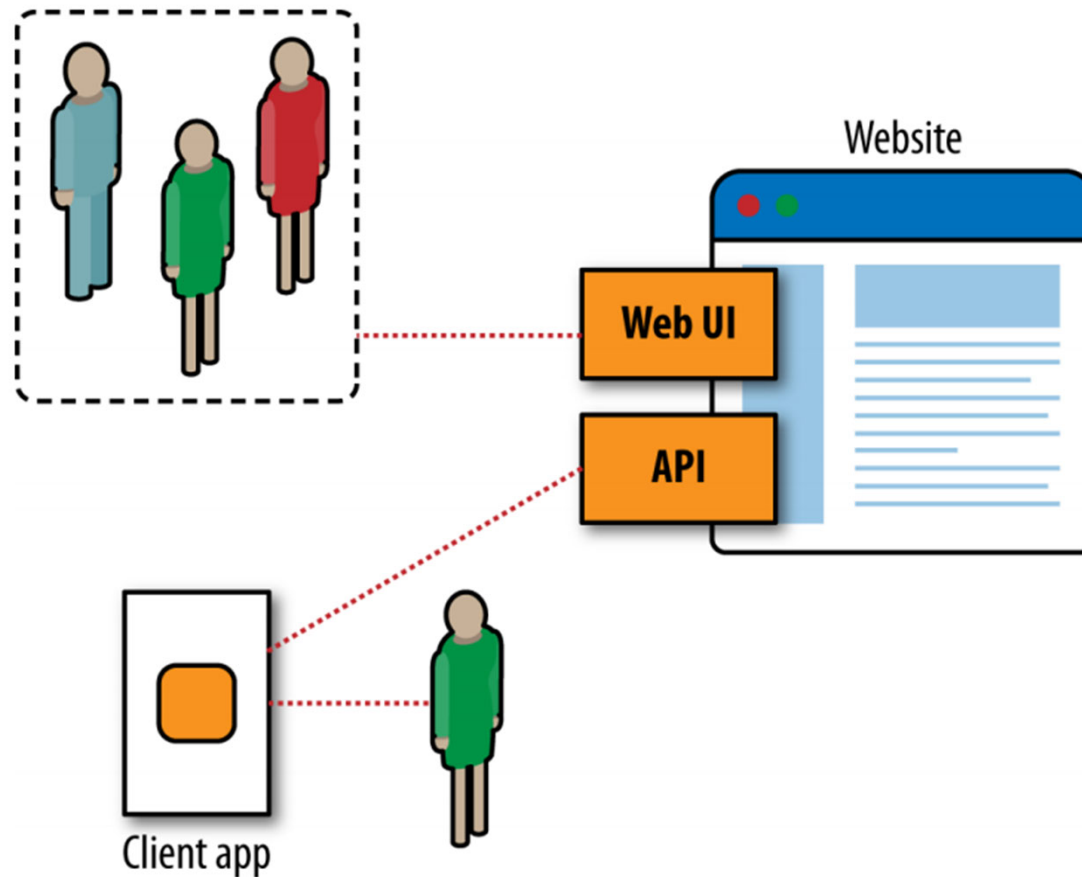




07 Web

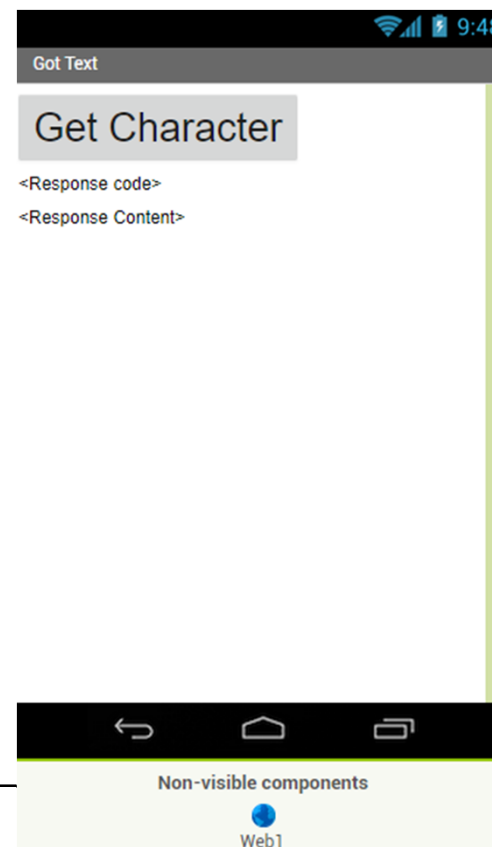
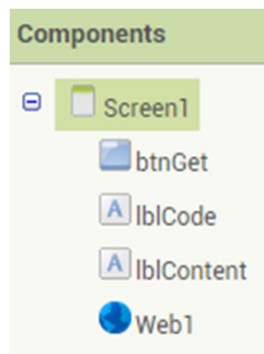
Got Text
JSON Text Decode

Communicate with the web api



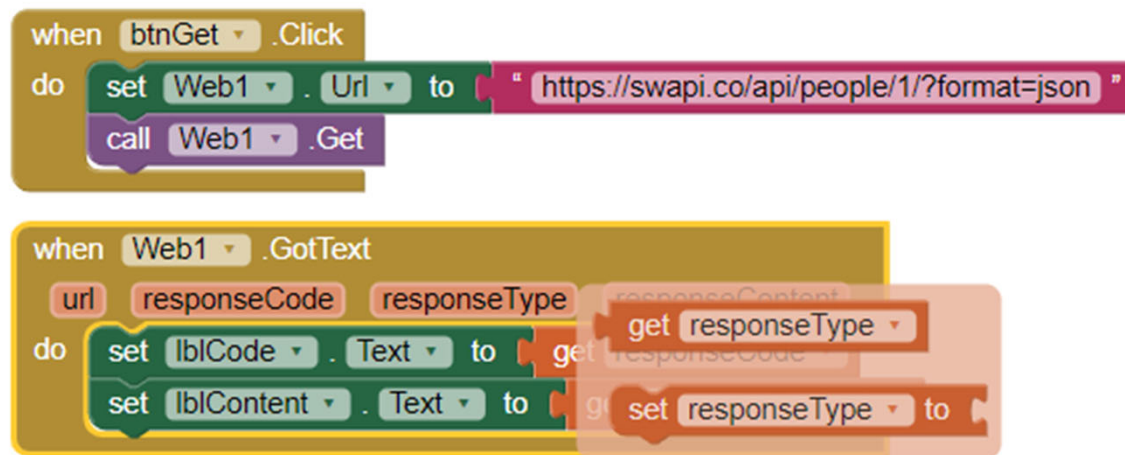
Got Text

- **App: Make a HTTP GET request to obtain information on a Star Wars character**
- **Designer view:**



Got Text

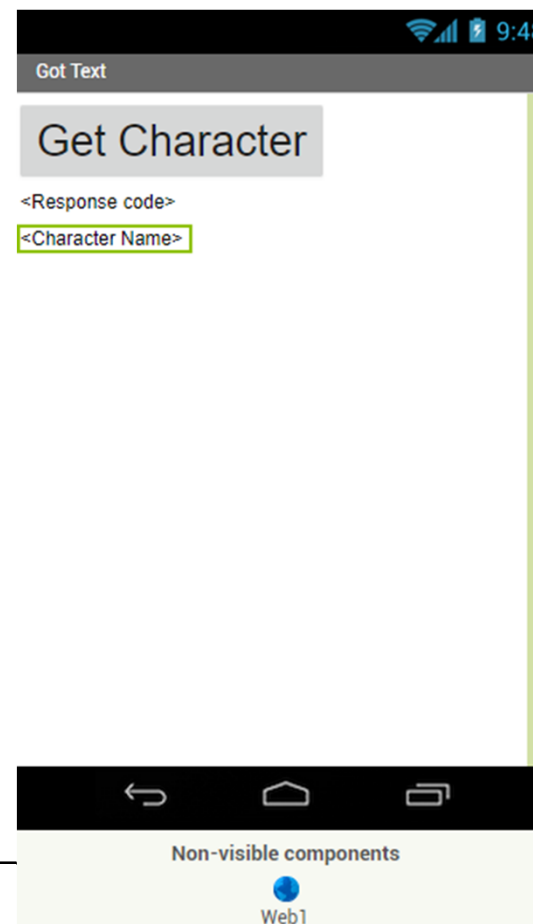
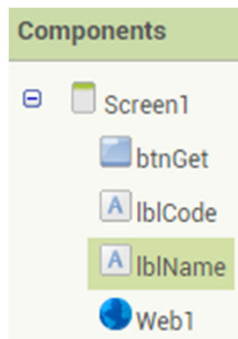
- **Blocks view:**



- **URL:**
 - <https://swapi.co/api/people/1/?format=json>

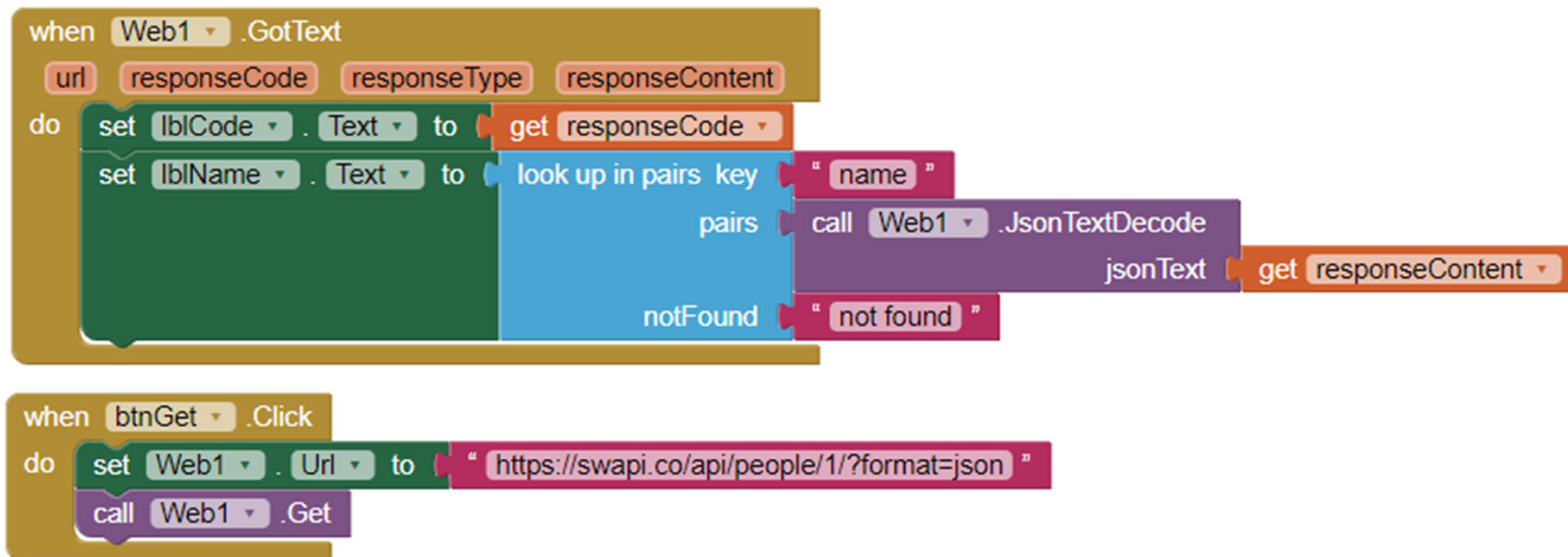
JSON Text Decode

- **App: Retrieve meaningful info from the raw web response**
- **Designer view:**



JSON Text Decode

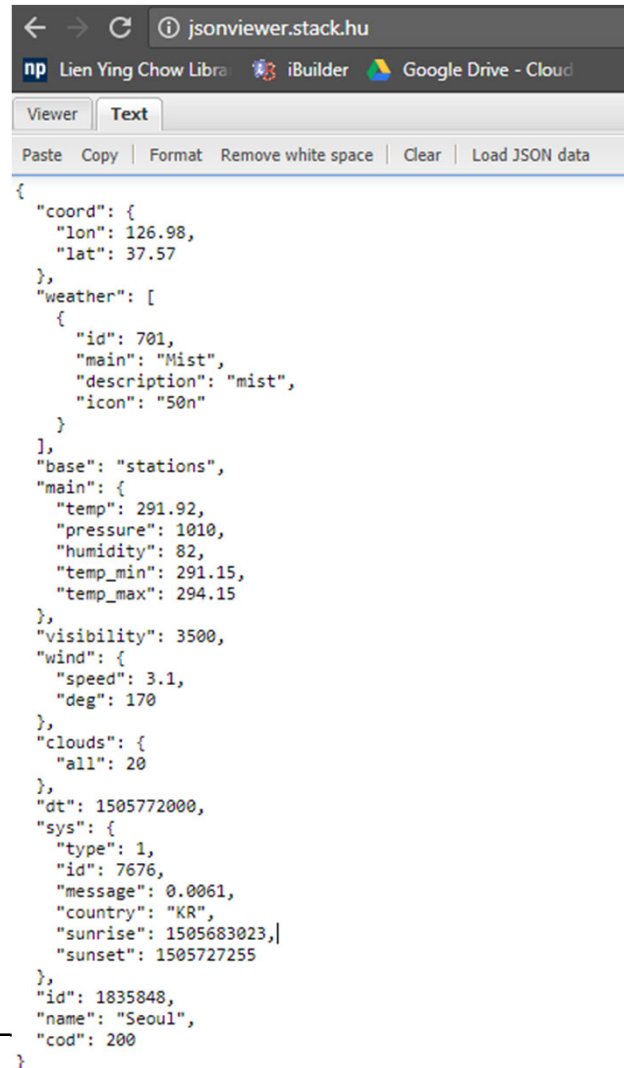
- **Blocks view:**



Assignment 2b

- **Web URL consists of three parts:**
 - <http://api.openweathermap.org/data/2.5/weather?id=>
 - <city_id>
 - &APPID=eb7f0bcecfed520b7d95898f925cf8e3
- **Example:**
 - <http://api.openweathermap.org/data/2.5/weather?id=1835848&APPID=eb7f0bcecfed520b7d95898f925cf8e3>

Assignment 2b



The screenshot shows a web browser window with the address bar displaying 'jsonviewer.stack.hu'. The browser's tab bar shows 'np Lien Ying Chow Libra', 'iBuilder', and 'Google Drive - Cloud'. The page has a 'Viewer' tab selected, and a toolbar with 'Paste', 'Copy', 'Format', 'Remove white space', 'Clear', and 'Load JSON data'. The main content area displays a JSON object representing weather data for Seoul.

```
{
  "coord": {
    "lon": 126.98,
    "lat": 37.57
  },
  "weather": [
    {
      "id": 701,
      "main": "Mist",
      "description": "mist",
      "icon": "50n"
    }
  ],
  "base": "stations",
  "main": {
    "temp": 291.92,
    "pressure": 1010,
    "humidity": 82,
    "temp_min": 291.15,
    "temp_max": 294.15
  },
  "visibility": 3500,
  "wind": {
    "speed": 3.1,
    "deg": 170
  },
  "clouds": {
    "all": 20
  },
  "dt": 1505772000,
  "sys": {
    "type": 1,
    "id": 7676,
    "message": 0.0061,
    "country": "KR",
    "sunrise": 1505683023,
    "sunset": 1505727255
  },
  "id": 1835848,
  "name": "Seoul",
  "cod": 200
}
```


Assignment 2b

- **<City_id>:**
 - Refer to 'city.list.json.gz'
 - unzip and open using wordpad

Singapore	1880252
Johor Bahru	1732752
Seoul	1835848

Assignment 2b

- **Blocks view:**
 - Create a global list “country” with three elements, i.e. the city_ids of Singapore, Johor Bahru & Seoul
 - Create two global variables to store the static portions of the web URL
 - When ListPicker.AfterPicking
 - Set ListPicker.Text to ListPicker.Selection
 - Set the Web.Url to a complete link with the city_id added
 - Call Web.Get

Assignment 2b

- **Blocks view:**
 - When Web.GotText:
 - if the responseCode = 200:
 - make lblLog visible
 - set lblLog.Text to Web.Url

Assignment 2b

initialize global APPID to "&APPID=eb7f0bcecfed520b7d95898f925cf8e3"

initialize global URL to "http://api.openweathermap.org/data/2.5/weather?id="

initialize global country to

- make a list
 - "1880252"
 - "1732752"
 - "1835848"

when Web1 . GotText

url responseCode responseType responseContent

do

- if
 - get responseCode = 200
- then
 - set lblLog . Visible to true
 - set lblLog . Text to Web1 . Url

when ListPicker1 . AfterPicking

do

- set ListPicker1 . Text to ListPicker1 . Selection
- set Web1 . Url to
 - join
 - get global URL
 - select list item list
 - get global country
 - index ListPicker1 . SelectionIndex
 - get global APPID
- call Web1 . Get

Assignment 2c

- **Blocks view:**
 - Create a subroutine “procedure” with input parameter “json” containing the response content from web GET request
 - Hint: Use local variable and JsonTextDecode in the subroutine to extract the information needed, i.e. humidity, pressure, temperature etc