



[PC-2: APIs and RPA]

ASSIGNMENT 1

APIs

by Arvind Subramanian



Qn 1: Consuming APIs (Open Banking)

API 1	
Name	OCBC ATM Locator
About	Provides convenient, easy access to info on OCBC ATM Locations. https://api.ocbc.com/store/site/pages/api_documentation.jag?name=ATM_Locator&title=ATM
Methods	.get
URL	https://api.ocbc.com:8243/atm_locator/1.1
URI	Nil
Params	Category: '1' for ATMs, '2' for Fastlanes, '3' for Passbook Update Machines (optional) Country: 'SG' for Singapore, 'MY' for Malaysia (optional) Landmark: e.g. Esplanade Mall, 7-Eleven etc. (optional) Latitude: Coordinates of Location (optional) Longitude: Coordinates of Location (optional) Radius: Search radius from given coordinates (required if coordinates are entered)
Headers	Authorization: Bearer <Access Token> Content-Type: application/json Accept: application/json
Body	Nil
Responses	200: Successful 4xx/5xx: Failed. Response body provides details of error

Mocked and Tested
on Postman

Mock Server Call Log C		
TIME	REQUEST	RESPONSE
3:11 AM, 17 Sep 2020	GET /*?category=&country=SG&landmark=&latitude=1.421448&longitude=1...	success
3:07 AM, 17 Sep 2020	GET /*?category=&country=SG&landmark=&latitude=1.421448&longitude=1...	success
3:05 AM, 17 Sep 2020	GET /*?category=&country=SG&landmark=&latitude=1.421448&longitude=1...	success
3:05 AM, 17 Sep 2020	GET /*?category=&country=SG&landmark=&latitude=1.421448&longitude=1...	success
3:04 AM, 17 Sep 2020	GET /*?category=&country=SG&landmark=&latitude=1.421448&longitude=1...	success
3:04 AM, 17 Sep 2020	GET /*?category=&country=SG&landmark=&latitude=1.421448&longitude=1...	success

Mock Server
Request Logs

Postman interface showing a GET request to the OCBC ATM Locator API. The request parameters are: category=1, country=SG, landmark=, latitude=1.421448, longitude=103.842629, radius=1. The response is a JSON array of ATM locations.

```
1  {
2    "atmList": [
3      {
4        "category": "Atm",
5        "address": "413 Yishun Ring Road #01-1873",
6        "landmark": "Yishun Ring Road - 7-Eleven",
7        "latitude": 1.4249477,
8        "longitude": 103.8464827,
9        "postalCode": "760413",
10       "remark": ""
11      },
12      {
13        "category": "Atm",
14        "address": "849 Yishun Ring Road",
15        "landmark": "Yishun Ring Road",
16        "latitude": 1.417124006,
17        "longitude": 103.8352096,
18        "postalCode": "760849",
19        "remark": "2 Locations: #01-K2, #01-3701/703 FairPrice"
20      },
21      {
22        "category": "Atm",
23        "address": "414 Yishun Ring Road #01-1853",
24        "landmark": "Yishun Ring Road - FairPrice",
25        "latitude": 1.424943,
26        "longitude": 103.846419,
27        "postalCode": "760414",
28        "remark": ""
29      },
30      {
31        "category": "Atm",
32        "address": "301 Yishun Avenue Concourse Area",
33        "landmark": "Yishun MRT Station",
34        "latitude": 1.4294324,
35        "longitude": 103.8349697,
36        "postalCode": "769893",
37        "remark": ""
38      }
39    ]
40  }
```

Qn 1: Consuming APIs (Open Banking)

API 1	
Name	OCBC ATM Locator
About	Provides convenient, easy access to info on OCBC ATM Locations. https://api.ocbc.com/store/site/pages/api_documentation.jag?name=ATM_Locator&title=ATM
Methods	.get
URL	https://api.ocbc.com:8243/atm_locator/1.1
URI	Nil
Params	Category: '1' for ATMs, '2' for Fastlanes, '3' for Passbook Update Machines (optional) Country: 'SG' for Singapore, 'MY' for Malaysia (optional) Landmark: e.g. Esplanade Mall, 7-Eleven etc. (optional) Latitude: Coordinates of Location (optional) Longitude: Coordinates of Location (optional) Radius: Search radius from given coordinates (required if coordinates are entered)
Headers	Authorization: Bearer <Access Token> Content-Type: application/json Accept: application/json
Body	Nil
Responses	<u>200</u> : Successful <u>4xx/5xx</u> : Failed. Response body provides details of error

Calling Mock API
from JavaScript

Code is attached in folder:

- qn1/qn1a_xx

The screenshot shows a web browser window at the top with the title "ATM Locations" and a URL bar showing "127.0.0.1:5500/APIs-and-RPA/Assignments/Assignment%201/qn%201/qn1a_index.html". The browser displays a JSON response for "atmList" with details for an ATM at "413 Yishun Ring Road #01-1873". Below the browser, a Visual Studio Code window is open with a file named "qn1a_script.js". The code in the editor is a JavaScript script that defines a URL for a mock API, sets up headers for a GET request, and uses the fetch API to retrieve data, which is then displayed in the browser's HTML.

```
//(from sandbox)
//const url = "https://api.ocbc.com:8243/atm_locator/1.1?country=SG&longitude=103.8464827"

//(from mock)
const url = "https://0641ff11-5ca1-4eb3-ba0b-d6b8c02fe024.mock.pstmn.io/*?category=

const options = {
  method: 'GET',
  headers: {
    'Accept': 'application/json',
    'Content-Type': 'application/json',
    'Authorization': 'Bearer 3aab265ece717fc7f0fca79db6c5fb6a'
  }
}

fetch(url, options)
  .then(res => res.json())
  .then(function(data) {
    document.getElementById("atms").innerHTML = JSON.stringify(data, null, '<br>')
  })
```

Qn 1: Consuming APIs (Open Banking)

API 2		
Name	OCBC Home Loan Eligibility Calculator	
About	Provides convenient, easy valuation on Home Loan eligibility. https://api.ocbc.com/store/site/pages/api_documentation.jag?name=Home_Loan&title=Home%20Loan%20Calculator	
Methods	.get	
URL	https://api.ocbc.com:8243/Home_Loan/1.0	
URI	Nil	
Params	<u>currentAge1</u> <u>totalMonthlyIncome1</u> : Details of borrower 1 <u>totalMonthlyDebt1</u> <u>outstandingLoans1</u> <u>repaymentPeriod</u> : No. of years in integers <u>hdbFlat</u> : True or False	<u>currentAge2</u> : (optional) <u>totalMonthlyIncome2</u> : (optional) <u>totalMonthlyDebt2</u> : (optional) <u>outstandingLoans2</u> : (optional)
Headers	Authorization: Bearer <Access Token> Content-Type: application/json Accept: application/json	
Body	Nil	
Responses	200: Successful 4xx/5xx: Failed. Response body provides details of error	

Mock Server Call Log		
TIME	REQUEST	RESPONSE
5:35 PM, 17 Sep 2020	GET /*?currentAge1=28&totalMonthlyIncome1=7000&totalMonthlyDebt1=20...	success
3:37 AM, 17 Sep 2020	GET /*?currentAge1=28&totalMonthlyIncome1=7000&totalMonthlyDebt1=20...	success
3:15 AM, 17 Sep 2020	GET /*?currentAge1=28&totalMonthlyIncome1=7000&totalMonthlyDebt1=20...	success
3:15 AM, 17 Sep 2020	GET /*?currentAge1=28&totalMonthlyIncome1=7000&totalMonthlyDebt1=20...	success
3:14 AM, 17 Sep 2020	GET /*?currentAge1=28&totalMonthlyIncome1=7000&totalMonthlyDebt1=20...	success
3:14 AM, 17 Sep 2020	GET /*?currentAge1=28&totalMonthlyIncome1=7000&totalMonthlyDebt1=20...	success
3:14 AM, 17 Sep 2020	GET /*?currentAge1=28&totalMonthlyIncome1=7000&totalMonthlyDebt1=20...	success

Mocked and Tested
on Postman

Postman interface showing a GET request to a mock API endpoint. The request parameters are: currentAge1=28, totalMonthlyIncome1=7000, totalMonthlyDebt1=2000, outstandingLoans1=25000, currentAge2, totalMonthlyIncome2, and totalMonthlyDebt2. The response body is a JSON object containing loan details.

```
1 {
2   "results": {
3     "eligibleLoanAmount": 222000,
4     "year1MonthlyPayment": 2002,
5     "year2MonthlyPayment": 2005,
6     "year3MonthlyPayment": 2005,
7     "thereafterMonthlyPayment": 2005,
8     "year1InterestRate": 1.58,
9     "year2InterestRate": 1.62,
10    "year3InterestRate": 1.62,
11    "thereafterInterestRate": 1.62
12  },
13  "disclaimer": "The data returned are indicative and are subjected to changes without prior notice. Please use the data at your own discretion and risk."
14 }
```

Mock Server
Request Logs

Qn 1: Consuming APIs (Open Banking)

API 2		
Name	OCBC Home Loan Eligibility Calculator	
About	Provides convenient, easy valuation on Home Loan eligibility. https://api.ocbc.com/store/site/pages/api_documentation.jag?name=Home_Loan&title=Home%20Loan%20Calculator	
Methods	.get	
URL	https://api.ocbc.com:8243/Home_Loan/1.0	
URI	Nil	
Params	<u>currentAge1</u> <u>totalMonthlyIncome1</u> : Details of borrower 1 <u>totalMonthlyDebt1</u> <u>outstandingLoans1</u> <u>repaymentPeriod</u> : No. of years in integers <u>hdbFlat</u> : True or False	<u>currentAge2</u> : (optional) <u>totalMonthlyIncome2</u> : (optional) <u>totalMonthlyDebt2</u> : (optional) <u>outstandingLoans2</u> : (optional)
Headers	Authorization: Bearer <Access Token> Content-Type: application/json Accept: application/json	
Body	Nil	
Responses	<u>200</u> : Successful <u>4xx/5xx</u> : Failed. Response body provides details of error	

Calling Mock API
from JavaScript

Code is attached in folder:

- qn1/qn1b_xx

Loan Eligibility

```
{
  "results": {
    "eligibleLoanAmount": 222000,
    "year1MonthlyPayment": 2002,
    "year2MonthlyPayment": 2005,
    "year3MonthlyPayment": 2005,
    "thereafterMonthlyPayment": 2005,
    "year1InterestRate": 1.58,
    "year2InterestRate": 1.62,
    "year3InterestRate": 1.62,
  }
}
```

qn1b_scriptjs - PC 2 - Visual Studio Code

```
qn1b_index.html  JS qn1b_scriptjs X
APIs-and-RPA > Assignments > Assignment 1 > qn 1 > JS qn1b_scriptjs > ...
1  //(from sandbox)
2  //const url = "https://api.ocbc.com:8243/Home_Loan/1.0?currentAge1=28&totalMonthlyI
3
4  //(from mock)
5  const url = "https://33e04854-be96-4027-a0b5-fc074c05f2da.mock.pstmn.io/*?currentAg
6
7  const options = {
8    method: 'GET',
9    headers: {
10      'Accept': 'application/json',
11      'Content-Type': 'application/json',
12      'Authorization': 'Bearer 3aab265ece717fc7f0fca79db6c5fb6a'
13    }
14  }
15
16  fetch(url, options)
17    .then(res => res.json())
18    .then(function(data) {
19      document.getElementById("loan").innerHTML = JSON.stringify(data, null, '<br>
20    })
}
```

Qn 2: Executing Payments Related API

API 3					
Name	Square Payments API				
About	Provides developers with a range of functions to obtain and manage payments: <u>List, Create, Get, Cancel, Complete</u> https://developer.squareup.com/reference/square/payments-api				
URL	https://connect.squareup.com/v2/payments				
Methods	List	Create	Get	Cancel	Complete
URI	Nil	Nil	/{{payment_id}}	/{{payment_id}}/cancel	/{{payment_id}}/complete
Params	Nil				
Headers	Authorization: Bearer <Access Token> Content-Type: application/json Square-Version: 2020-08-26				
Body	Nil	Yes	Nil	Nil	Nil
Responses	200: Successful 4xx/5xx: Failed. Response body provides details of error				

Mocked and Tested on Postman

Mock Server Request Logs

Postman

File Edit View Help

New Import Runner

My Workspace

Invite

No Environment

Upgrade

Filter

History Collections APIs

New Collection Trash

POST createpayment

createpayment

Examples 1 BUILD

POST https://76ec209c-ab21-4032-b90e-3a5600736800.mock-postman.io/v2/payments

Send Save

Params Authorization Headers (11) Body Pre-request Script Tests Settings

Cookies Code

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Beautify

1 2 3 4 5 6 7 8 9

10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38

Body Cookies Headers (21) Test Results

Status: 200 OK Time: 1341 ms Size: 2.03 KB Save Response

Pretty Raw Preview Visualize JSON

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38

Find and Replace Console

Bootcamp Build Browse

Qn 2: Executing Payments Related API

Create Payment <pre>{ "payment": { "id": "RC5uOvybD94Xe8Vr1RwADzqPWIEZY", "created_at": "2020-09-17T08:26:04.709Z", "updated_at": "2020-09-17T08:26:04.824Z", "amount_money": { "amount": 320, "currency": "USD" }, "status": "APPROVED", } }</pre>	Get Payment <pre>{ "payment": { "id": "RC5uOvybD94Xe8Vr1RwADzqPWIEZY", "created_at": "2020-09-17T08:26:04.709Z", "updated_at": "2020-09-17T08:26:04.824Z", "amount_money": { "amount": 320, "currency": "USD" }, "status": "APPROVED", } }</pre>
Cancel Payment <pre>{ "payment": { "id": "n5lFecJuWQ8c9soUETV6Vdjoaa8YY", "created_at": "2020-09-17T08:18:30.944Z", "updated_at": "2020-09-17T08:24:36.629Z", "amount_money": { "amount": 320, "currency": "USD" }, "status": "CANCELED", } }</pre>	Complete Payment <pre>{ "payment": { "id": "RC5uOvybD94Xe8Vr1RwADzqPWIEZY", "created_at": "2020-09-17T08:26:04.709Z", "updated_at": "2020-09-17T08:28:03.836Z", "amount_money": { "amount": 320, "currency": "USD" }, "status": "COMPLETED", } }</pre>

Calling Mock API
from JavaScript

Code is attached in folder
- qn2/qn2a_xx to qn2e_xx

Sample code for 'create payment'

```
Terminal Help qn2b_scriptjs - PC 2 - Visual Studio Code
JS qn2b_scriptjs x
APIs-and-RPA > Assignments > Assignment 1 > qn 2 > JS qn2b_scriptjs > url
1 // (from sandbox)
2 // const url = "https://connect.squareupsandbox.com/v2/payments";
3
4 // (from mock)
5 const url = "https://76ec209c-ab21-4032-b90e-3a5600736800.mock.pstmn.io/v2/payments";
6
7 const options = {
8   method: 'POST',
9   headers: {
10     'Square-Version': '2020-08-26',
11     'Content-Type': 'application/json',
12     'Authorization': 'EAAAEDT0g5R8jH5fHpiNr3l8F2hHj_nsyj2Iw9fbZlTeEsJS1Cv6qpSw7',
13   },
14   body: JSON.stringify({
15     "amount_money": {
16       "amount": 320,
17       "currency": "USD"
18     },
19     "idempotency_key": "c7fecba2-9185-4900-b3c4-12d36b83919d",
20     "source_id": "cnon:card-nonce-ok",
21     "autocomplete": "False"
22   })
23 }
24 fetch(url, options)
25   .then(res => res.json())
26   .then(function(data) {
27     document.getElementById("crtpmt").innerHTML = JSON.stringify(data, null, '<
28   })
29 }
```

Qn 2: Executing Payments Related API

API 4			
Name	OCBC Transfer Limit Modifier		
About	Provides easy modifications to a variety of Transfer Limits such as Overseas Transfer, Bill Payment and Card Payment (only 3 are depicted here). https://api.ocbc.com/store/site/pages/api_documentation.jag?title=Update%20Card%20Payment%20Authorization%20Limit		
Methods	.post		
URL	https://api.ocbc.com:8243/transactional/authlimits/1.0/		
URI	/updateBillPaymentAuthLimit	/updateOverseasTransferAuthLimit	/updateCardPaymentAuthLimit
Params	Nil		
Headers	Authorization: Bearer <Access Token> Content-Type: application/json Accept: application/json		
Body	-AccountNo -NewBillPaymentAuthLimit	-AccountNo -NewOverseasTransferAuthLimit	-AccountNo -NewCardPaymentAuthLimit
Responses	<u>200</u> : Successful <u>4xx/5xx</u> : Failed. Response body provides details of error		

Postman

File Edit View Help

New Import Runner

My Workspace Invite

No Environment

History Collections APIs

New Collection Trash

POST /updateCardPaymentAuthLimit

Send Save

Params Authorization Headers (11) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "AccountNo": "123-xxxxx-001",
3   "NewCardPaymentAuthLimit": "50000"
4 }
```

Body Cookies Headers (14) Test Results

Status: 200 OK Time: 1076 ms Size: 722 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "Success": true,
3   "Results": {
4     "AccountNo": "123-xxxxx-001",
5     "CardPaymentAuthLimit": 50000
6   },
7   "Disclaimer": "The data returned are indicative and are subjected to changes without prior notice. Please use the data at your own discretion and risk."
8 }
```

Mocked and Tested
on Postman

https://web.postman.co/mocks/9320c515-8531-4335-af3e-1c261a4f3daa/call-logs?workspace=5bad183c...

Build Beta Browse Explore Analyze My Workspace Invite Upgrade

ocbc.transferlimits_mock

OCBC - Transfer Limits No environment Arvind Subramanian

View Documentation Copy Mock Server URL

Mock Server Call Log

Search your calls...

TIME	REQUEST	RESPONSE
5:55 PM, 17 Sep 2020	POST /updateCardPaymentAuthLimit	success
3:52 AM, 17 Sep 2020	POST /updateCardPaymentAuthLimit	success
3:52 AM, 17 Sep 2020	POST /updateCardPaymentAuthLimit	success
3:44 AM, 17 Sep 2020	POST /updateCardPaymentAuthLimit	success
3:44 AM, 17 Sep 2020	POST /updateCardPaymentAuthLimit	success
3:43 AM, 17 Sep 2020	POST /updateCardPaymentAuthLimit	success
3:43 AM, 17 Sep 2020	POST /updateCardPaymentAuthLimit	success

Mock Server
Request Logs

Qn 2: Executing Payments Related API

Calling Mock API
from JavaScript

```
JS qn2g_scriptjs X
APIs-and-RPA > Assignments > Assignment 1 > qn 2 > JS qn2g_scriptjs > ...
1  //(from sandbox)
2  //const url = "https://api.ocbc.com:8243/transactional/authlimits/1.0/updateCardPay
3
4  //(from mock)
5  const url = "https://9320c515-8531-4335-af3e-1c261a4f3daa.mock.pstmn.io/updateCardF
6
7  const options = {
8    method: 'POST',
9    headers: {
10      'Accept': 'application/json',
11      'Content-Type': 'application/json',
12      'Authorization': 'Bearer 3aab265ece717fc7f0fca79db6c5fb6a'
13    },
14    body: JSON.stringify({
15      'AccountNo': '123-XXXX-001',
16      'NewCardPaymentAuthLimit': '50000'
17    })
18  }
19
20  fetch(url, options)
21    .then(res => res.json())
22    .then(function(data) {
23      document.getElementById("card").innerHTML = JSON.stringify(data, null, '<br>
24    })
25  }
```

<- Sample code for 'change card
payment limit'

Code is attached in folder:

- qn2/qn2f_xx to qn2g_xx

127.0.0.1:5500/APIs-and-RPA/Assignments/Assignment%201/qn%202/qn2f_index.html

Get Card Payment Limit

1

```
{
  "Success": true,
  "Results": {

    "AccountNo": "123-xxxx-001",

    "CardPaymentAuthLimit": 20000
  },
  "Disclaimer": "The data returned are indicative and are subjected to changes without prior notice.Please use the data at your own discretion and risk." }
```

127.0.0.1:5500/APIs-and-RPA/Assignments/Assignment%201/qn%202/qn2g_index.html

Change Card Payment Limit

2

```
{
  "Success": true,
  "Results": {

    "AccountNo": "123-xxxx-001",

    "CardPaymentAuthLimit": 50000
  },
  "Disclaimer": "The data returned are indicative and are subjected to changes without prior notice.Please use the data at your own discretion and risk." }
```

Qn 3: Designing an API

API IDEA			
SMRT Train Location and Crowd Tracker			
About	Provide Accurate and Real-Time Data on: <ul style="list-style-type: none">- Location of each train in each MRT line- Crowd concentration in each train and in each compartment- Crowd concentration in every station and passenger walkway		Leading to... <ul style="list-style-type: none">- Less waiting time and overall commute duration- Less crowdedness and enhanced comfort level
Current Situation	Lack of awareness on the arrival times of trains until at or very near the MRT stations Estimates displayed by SMRT may be inaccurate Lack of awareness on the crowdedness of trains and platforms		
Possible Consumers	Map Service Providers (Google, Apple etc.) Taxi Companies (Grab, Comfort etc.)		
Value-Add to Consumer	Map Service Providers Increased Usage by MRT Commuters <ul style="list-style-type: none">- Provide more accurate train commute timings on their own services- Maps generally more utilized by drivers- Providing quality-of-life upgrade encourages day-to-day usage Effective Promotion of Other Services <ul style="list-style-type: none">- Better awareness of MRT service conditions allows more targeted advertising- e.g. '6-mins till the next train, why not grab a quick curry-puff at the OCK near you?'	Taxi Companies Improve Assessment of Passenger Demand <ul style="list-style-type: none">- Monitor MRT service conditions alongside weather, traffic etc.- Adjust taxi pricing in accordance with real-time 'attractiveness' of MRT commute	
End-Users of Client App	Everyday MRT Commuters		
Value-Add to End-Users	Enhanced Quality of Life in MRT Commute <ul style="list-style-type: none">- Lower travel durations- Know where to stand/wait to enjoy the most comfortable ride- Flexibility to squeeze in other personal activities during/before commute (e.g. grab a quick toilet break)	Better Overall Commuting Experience <ul style="list-style-type: none">- Know for sure whether to walk to the MRT or just call a Grab instead- Improved accessibility to taxi services when needed	
Value-Add to Company	Greater ridership Improved crowd distribution on trains and platforms (no need for station marshals) Automated data collection on train run-times, mileage and passenger load <ul style="list-style-type: none">- Improved maintenance and train scheduling Accelerated response to train delays/faults Revenue generation from API usage →	Pricing Model Pay-per-Use or Subscription Based <ul style="list-style-type: none">- No direct transactions provided to consumers to charge commission off- Focus is on enhancing business efficiency and improving commuter quality of life	
Transmitted Data	[Request] .get StationCrowdedness [Response] crowd density across all areas of a station	[Request] .get NearestTrainLocations [Response] location and arrival time of next train for a given station	[Request] .get TrainCrowdedness [Response] crowd density across the length of a train for a given train



END