

# SYSTEMS DEVELOPMENT OFFICER PROJECT

**DUE DATE : Friday, 26 NOV 2021 midnight**

## Choice of Language

Use **any** of the following as the core backend programming language. Use whatever language for the front end.

1. Java
2. PHP
3. C#

## Challenge

MRA intends to widen its tax net by registering new tax payers. One way to do this is to send officers to recruit new taxpayers who are operating businesses yet they have not registered for taxes.

You have been identified as a developer to develop an application (that will consume the provided webservice) to be used by the field officers to register tax payers.

## Requirements

- The application should allow field officers to login using a username and a password
- The application should allow field officers to create a new taxpayer record with the attributes specified in the registration web service.

***NOTE:*** All validations are up to you. Analyse the fields exposed by the webservice and make sure your application includes all necessary validations to ensure quality data is submitted.

- The application shall allow view of all taxpayers registered by the logged in user.
- Field officers should be able to create, edit and delete taxpayers.

**NOTE:** Web methods for all the specified functions have been provided by the webservice. The key objectives of the exercise are:

1. For you to build a front end with validations to ensure quality data submission.
2. For you to consume the webservice and handle all response types

## Webservice

MRA has exposed a registration web service at this end point

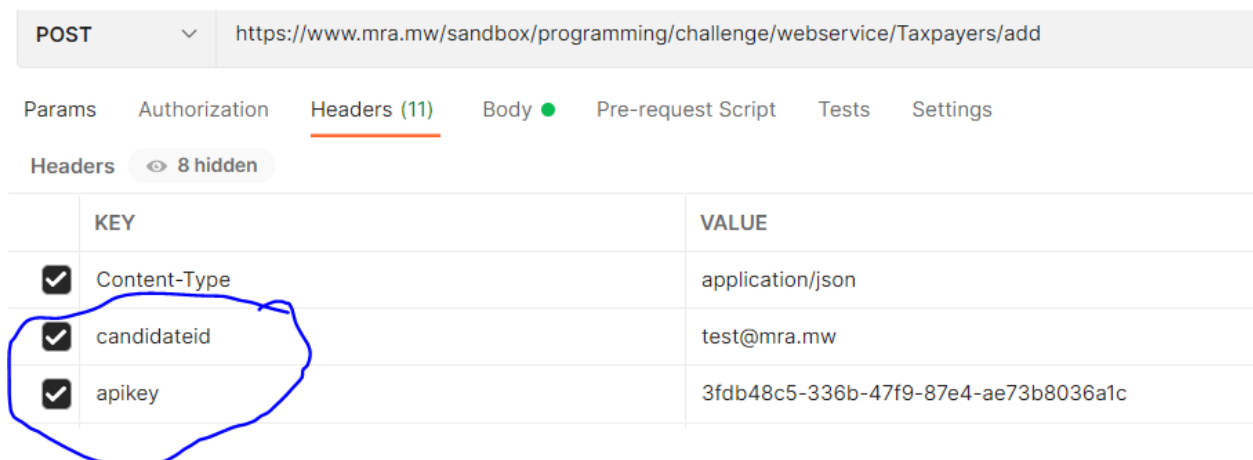
<https://www.mra.mw/sandbox>

The documentation is found in Appendix A.

As part of security, the web service requires a username and a password to be passed in the request header as follows;

Custom Http Header Name	Header Value
candidateid	Same as email address used for speedExam.net login in small cases
apikey	3fdb48c5-336b-47f9-87e4-ae73b8036a1c

See sample screenshot of the headers in PostMan



## Submission requirements

- A prototype of screen designs and menus in **power point**.
- A **word document** with steps taken to develop the solution. The document should be written to “target a junior programmer” who wants step by step process to create a similar project.
- A readMe.txt document with details on how to deploy your application
- A **public link** to a git repository on **github** where all your sources will be hosted.

When your project is complete, send an email to [mdnyirenda@mra.mw](mailto:mdnyirenda@mra.mw) copy [mchibaka@mra.mw](mailto:mchibaka@mra.mw), [mwmlomba@mra.mw](mailto:mwmlomba@mra.mw)

Email body should include your full name, link to your public repository

Note that the official submission is the most recent git commit before the dead line.

**All documents should be part of the git repository, do not attach any source codes or any documents in your email.**

## APPENDIX A: WEB SERVICE SPECIFICATION

Web Methods where **baseUri** = <https://www.mra.mw/sandbox>

### 1. Authentication

<b>Link</b>	<b>{{baseUri}}/programming/challenge/webservice/auth/login</b>
<b>Description</b>	This web method is for authenticating users of the web service.
<b>Sample Request</b>	<pre>{   "Email": "candidate@gmail.com",   "Password": "password000122" }</pre>

## 2. Log out

<b>Link</b>	<b>{{baseUrl}}/programming/challenge/webservice/auth/logout</b>
<b>Description</b>	This web method is for logging users out of the web service.
<b>Sample Request</b>	<pre>{   "Email": "bmangisoni@mra.mw" }</pre>
<b>Sample Response</b>	<pre>{   "ResultCode": 1,   "Remark": "Log out Successful",   "data": null }</pre>
<b>Result Codes</b>	1: Successful 2: Error

## 3. Create/Add Taxpayer

<b>Link</b>	<b>{{baseUrl}}/programming/challenge/webservice/Taxpayers/add</b>
<b>Description</b>	This web method is for creating taxpayers in the db
<b>Sample Request</b>	<pre>{   "TPIN": "20203030",   "BusinessCertificateNumber": "MBRS123456",   "TradingName": "Sample Trader",   "BusinessRegistrationDate": "2021/11/02",   "MobileNumber": "0995900000",   "Email": "bmangisoni@gmail.com",   "PhysicalLocation": "Namiwawa",   "Username": "bmangisoni" }</pre>
<b>HTTP Responses</b>	400: Bad request (Invalid Request, Username does not exist) 200: Successful 300: Ambiguous (Taxpayer already exists) 500: Internal Server Error
<b>Additional notes</b>	<ul style="list-style-type: none"><li>- Username is checked to ensure it is valid. It is required for association purposes and can be retrieved from the authentication response and should be passed along with all requests of this type.</li></ul>

#### 4. Get Taxpayer list

<b>Link</b>	<code>{{baseUrl}}/programming/challenge/webservice/Taxpayers/getAll</code>
<b>Description</b>	This web method is for retrieving a list of all registered taxpayers.
<b>Sample Request</b>	None. No request data required.
<b>Sample Response</b>	<pre>[   {     "TPIN": "20202021",     "BusinessCertificateNumber": "MBRS123456",     "TradingName": "Test Trading Name",     "BusinessRegistrationDate": "2021/11/02",     "MobileNumber": "0995926084",     "Email": "bmangisoni@gmail.com",     "PhysicalLocation": "Namiwawa",     "Username": "bmangisoni",     "Deleted": false,     "id": 5   },   {     "TPIN": "20202024",     "BusinessCertificateNumber": "MBRS123456",     "TradingName": "Test Trading Name",     "BusinessRegistrationDate": "2021/11/02",     "MobileNumber": "0995926084",     "Email": "bmangisoni@gmail.com",     "PhysicalLocation": "Namiwawa",     "Username": "bmangisoni",     "Deleted": false,     "id": 8   },   {     "TPIN": "20202025",     "BusinessCertificateNumber": "MBRS123456",     "TradingName": "Simple Trader",     "BusinessRegistrationDate": "2021/11/02",     "MobileNumber": "0995926084",     "Email": "bmangisoni@gmail.com",     "PhysicalLocation": "Namiwawa",     "Username": "bmangisoni",     "Deleted": false,     "id": 9   } ]</pre>
<b>HTTP Response Codes</b>	200: Successful 500: Error

## Sample of Getting all taxpayers using Postman

GET

https://www.mra.mw/sandbox/programming/challenge/webservice/Taxpayers/GetAll

Send

Params

Authorization

Headers (11)

Body

Pre-request Script

Tests

Settings

Cookies

Headers

8 hidden

	KEY	VALUE	DESCRIPTION		Bulk Edit	Presets
<input checked="" type="checkbox"/>	Content-Type	application/json				
<input checked="" type="checkbox"/>	candidateid	test@mra.mw				
<input checked="" type="checkbox"/>	apikey	3fdb48c5-336b-47f9-87e4-ae73b8036a1c				
	Key	Value	Description			

Body

Cookies

Headers (9)

Test Results

Status: 200 OKTime: 57 msSize: 793 BSave Response

Pretty

Raw

Preview

Visualize

```
[{"TPIN": "20203030", "BusinessCertificateNumber": "MBRS123456", "TradingName": "Sample Trader", "BusinessRegistrationDate": "2021/11/02", "MobileNumber": "0995926084", "Email": "taxpayer@gmail.com", "PhysicalLocation": "Namiwawa", "Username": "test@mra.mw", "Deleted": false, "id": 1}, {"TPIN": "20203034", "BusinessCertificateNumber": "MBRS123456", "TradingName": "Sample Trader", "BusinessRegistrationDate": "2021/11/02", "MobileNumber": "0995926084", "Email": "taxpayer@gmail.com", "PhysicalLocation": "Namiwawa", "Username": "test@mra.mw", "Deleted": false, "id": 2}]
```

### 5. Edit taxpayer

Link	{{baseUrl}}/programming/challenge/webservice/Taxpayers/edit
Description	This web method is for retrieving a list of all registered taxpayers.
Sample Request	<pre>{   "TPIN": "20203030",   "BusinessCertificateNumber": "MBRS123456",   "TradingName": "Simple Trader",   "BusinessRegistrationDate": "2021/11/02",   "MobileNumber": "0995926084",   "Email": "bmangisoni@gmail.com",   "PhysicalLocation": "Namiwawa",   "Username": "bmangisoni",   "Deleted": false }</pre>
Sample Response	<pre>{   "ResultCode": 1,   "Remark": "Taxpayer edited successfully!",   "data": "20203030" }</pre>
HTTP Response Codes	400: Bad request (Invalid Request, Username does not exist) 200: Successful

	300: Ambiguous (Taxpayer already exists) 500: Internal Server Error
--	--

## 6. Delete Taxpayer

<b>Link</b>	<b>{{baseUrl}}/programming/challenge/webservice/Taxpayers/delete</b>
<b>Description</b>	This web method is for retrieving a list of all registered taxpayers.
<b>Sample Request</b>	<pre>{   "TPIN": "20203030" }</pre>
<b>Sample Response</b>	<pre>{   "resultCode": 1,   "Remark": "Taxpayer deleted successfully!",   "data": "20203030" }</pre>
<b>HTTP Response Codes</b>	200: Successful 500: Error