**MINISTRY OF TERTIARY EDUCATION, RESEARCH, SCIENCE AND TECHNOLOGY**

**DEPARTMENT OF TERTIARY EDUCATION FINANCING**

**dtef**
DEPARTMENT OF TERTIARY EDUCATION FINANCING

# Student Loan Management System

## Migration, Upgrading and Enhancement of Student Loan Management System (SLMS) and Provision of Twenty-Four (24) Months Support and Maintenance

## Interface Specification for Education Training Providers (ETPs)

*10th February 2023*

*VERSION 2.1*

AGILEUM
Flexible Services Digital.

VISION 2036
**Achieving Prosperity for All**

**OUR VISION**
**"A Knowledge based Society enabling Prosperity for all"**
**Collaboration Co-production Consuming**
**TOLL FREE: 0800-600-185**

BOTSWANA

## DOCUMENT CONTROL

### CHANGE RECORD

| DATE | AUTHOR | VERSION | CHANGE REFERENCE |
|------|--------|---------|------------------|
| 20-DEC-2021 | AGILEUM (PTY) LTD | 1.0 | NO PREVIOUS DOCUMENT |
| 21-JAN-2022 | AGILEUM (PTY) LTD | 1.1 | REVISED – WITH THE REST API |
| 08-DEC-2022 | AGILEUM (PTY) LTD | 2.0 | Incorporated results and registers |
| 10-FEB-2023 | AGILEUM (PTY) LTD | 2.1 | Added some fields as per the revised DTEF template |

### REVIEWERS

| NAME | POSITION |
|------|----------|
| SLMS PROJECT IMPLEMENTATION COMMITTEE | PROJECT DELIVERY RESPONSIBILITY |

### APPROVALS

| NAME | POSITION |
|------|----------|
| SLMS PROJECT IMPLEMENTATION COMMITTEE | PROJECT DELIVERY RESPONSIBILITY |
| SLMS PROJECT STEERING COMMITTEE | CHAIRPERSON SLMS PROJECT STEERING COMMITTEE |
| SLMS TECHNICAL REFERENCE COMMITTEE | PROJECT TECHNICAL ADVISORY AND APPROVAL COMMITTEE |

## TABLE OF CONTENTS

## ABBREVIATIONS AND/OR ACRONYMS

Below is a list of abbreviations used in this document:

| Acronym | Meaning |
|---------|---------|
| API | Application Programming Interface |
| DIT | Department of Information Technology |
| DTEF | Department of Tertiary Education Financing |
| ETP | Education Training Provider |
| GDN | Government Data Network |
| ID | Identity |
| IT | Information Technology |
| JSON | JavaScript Object Notation |
| MOTE | Ministry of Tertiary Education, Research, Science and Technology |
| NIS | National Identification System |
| PIC | Project Implementation Committee |
| PSC | Project Steering Committee |
| SLMS | Student Loan Management System |

## 1.  INTRODUCTION

The Ministry of Tertiary Education, Research, Science and Technology (MoTE) through the Department of Tertiary Education Financing (DTEF) has entrusted to AGILEUM the project to upgrade, migrate and enhance Student Loan Management System (SLMS), as well as provide support and maintenance services for 24 months. The enhancement of SLMS includes the implementation of an e-Services Platform to provide Tertiary Education Financing services to students and to enable data exchange with critical stakeholders through the internet. In addition, the enhancements of the SLMS will include integration/interface implementation with various stakeholders' systems such as those of Education Training Providers'.

SLMS is currently using an Oracle 12c database, and an Oracle 12c Web-Logic, including Oracle 12c forms and reports. The Oracle SLMS is hosted at the Department of Information Technology (DIT). The e-Services platform is implemented using Drupal 8.9, apache 2 for web-server and database is being migrated from MySQL to Maria DB.

The Oracle SLMS and the eServices platform is interfaced through web-services. Oracle SLMS is interfaced with Omang and BDRS systems and eServices interfaces with them through SLMS.

### 1.1.  PURPOSE OF DOCUMENT

This document contains the specifications for interfacing the DTEF's SLMS application with the Education Training Providers (ETPs) systems. It specifies what SLMS requires from an institution's system.

## 2.  SLMS OPERATIONS OVERVIEW

DTEF provides funding to qualifying students in the form of various loan/grant schemes. DTEF advertises sponsorships and students apply for such. DTEF sponsors only Batswana who meet other illegibility criteria specified in the sponsorship adverts.

To ensure that DTEF sponsors only Batswana, and this is done by verifying against the National Identification System (NIS), the Omang system and the Birth and Death Registration System.

The other information needed by the DTEF is the admission status of the student at a training institution the applicant wishes to apply the sponsorship for. Hence the need for the admissions to be readily available in the e-Services platform so that the system can automatically refer to it or verify against it as the student applies for sponsorship.

For the 2021 placement, the ETP admissions information was provided on e-Services through uploading of csv files for respective institutions. Going forward it is required that the system be directly interfaced and web-services interfacing mechanism be used for this.

It is imperative that ETPs also provide DTEF with registration information of the learners at the beginning of each semester or term. The same will be provided through direct interface using web-services.

The ETPs are expected to provide results of learners matching the registered students and program/courses at the end of each term/semester. These are to be provided through direct interface using web-services.

The document provides the web-services' Application Programming Interface (API) to be utilised by ETPs to send information to DTEF system.

# 3. INTERFACE SPECIFICATION FOR ADMISSION SUBMISSION

## 3.1. INFORMATION REQUIRED FROM ETPS - ADMISSIONS

The APIs for admissions requires the information provided in the table below, which is to be provided by a training institution as they process their admissions for Batswana students.

| Item/Field | Comments | Field Type and Size (Data Type) |
|---|---|---|
| Title | Made of Identity (ID) Number, Institution Code, Program Code | |
| ID Number | Botswana National or Birth Certificate Number | VARCHAR2(9) |
| Surname | | VARCHAR2(60) |
| First-name | Including middle names | VARCHAR2(60) |
| Institution Program Code | Institution's Program Code as defined in SLMS | VARCHAR2(35) |
| Level of Study | Professional, Diploma, High Diploma, Degree | VARCHAR2(60) |
| Duration of Study | Number of years it will take for the student to finish studying | NUMBER |

| Commencement Date | Date the student is to start studying: MM/DD/YYYY | DATE |
|---|---|---|
| Completion Date | Date the student is expected to graduate: MM/DD/YYYY | DATE |
| Level of Entry | At what year is the student joining | NUMBER |
| Program Cost | The total amount the student is expected to pay for studying the indicated program | NUMBER |

ETPs are expected to send/post this information into the eServices platform using the REST API shared in the next section.

## 3.2.    ETP ADD NEW ADMISSION REST API DETAILS

The details of the API are as follows: Note that the URL will differ as we go along the development to live environment stage.

- API
  The API to send the POST request:
  [https://tef1.gov.bw/api/post/studentadmissions?_format=hal_json](https://tef1.gov.bw/api/post/studentadmissions?_format=hal_json)

  [https://tef2.gov.bw/api/post/studentadmissions?_format=hal_json](https://tef2.gov.bw/api/post/studentadmissions?_format=hal_json)

  [https://tef.gov.bw/api/post/studentadmissions?_format=hal_json](https://tef.gov.bw/api/post/studentadmissions?_format=hal_json)

- Authorization
  Type of Authorization: Basic Auth
  Each institution should have an account on DTEF eservices application and these credentials should be used as authentication.
  Username: xxxxxxxxx
  Password: xxxxxxxxxx

- Headers
  1. **X-CSRF-Token**
     Issue a GET request against [https://tef2.gov.bw/rest/session/token](https://tef2.gov.bw/rest/session/token)
     In response, you will get a jumble of letters and numbers (e.g. Eh1INrGyEUNBog5ZL2o-dHFPnLoseIKCcL35aVSGg94); this is your X-CSRF-Token
  2. **Content-Type**
     Use the following as the content-type header:
     Content-type: application/hal+json

| X-CSRF-Token | cZkAIFbf8dODMGFNWH9ID9rU4nRAyGYfqRgiwd4I-Ik |
|---|---|
| Content-Type | application/hal+json |

An example of the request header with its values

- Body
  To create a node for student admission on eservices, send a POST request with the following data structure.

```json
{
  "type": [
    {
      "target_id": "program_of_study"
    }
  ],
  "title": [
    {
      "value": " "
    }
  ],
  "id": [
    {
      "value": "829026413"
    }
  ],
  "surname": [
    {
      "value": "MORAGI"
    }
  ],
  "firstname": [
    {
      "value": "MATLHOGONOLO"
    }
  ],
  "institution": [
    {
      "value": "UNIVERSITY OF BOTSWANA"
    }
  ],
  "institution_program_code": [
    {
      "value": "BSB01"
    }
  ],
  "program_name": [
    {
      "value": "BACHELOR OF ACCOUNTANCY"
    }
  ],
  "program_duration": [
    {
      "value": "3"
    }
  ],
  "start_date": [
    {
      "value": "08/31/2022"
    }
  ],
  "completion_date": [
    {
      "value": "08/15/2025"
    }
  ],
  "entry_level": [
    {
      "value": "1"
    }
  ],
  "cost": [
    {
      "value": "10000"
    }
  ]
}
```

Above is an example of a JSON body.

For the first part of the JSON (type and title), these values should be left as given in the example. The **type** must always be **program_of_study** and **title** should always **be kept blank**.

```
{
  "type": [
    {
      "target_id": "program_of_study"
    }
  ],
  "title": [
    {
      "value": " "
    }
  ],
```

Start date and completion date should be in the following format: mm/dd/yyyy

```
"start_date": [
  {
    "value": "08/31/2022"
  }
],
"completion_date": [
  {
    "value": "08/15/2025"
  }
],
```

The value for institution should be exactly the same as from SLMS database.

```
"institution": [
  {
    "value": "UNIVERSITY OF BOTSWANA"
  }
],
```

The value for institution program code should be exactly the same as from SLMS database.

```
"institution_program_code": [
  {
    "value": "BSB01"
  }
],
```

Refer to the attached **post-student-admission.json** file for an example of the body, on how each record will look like.

OUR VISION
"A Knowledge based Society enabling Prosperity for all"
Collaboration Co-production Consuming
TOLL FREE: 0800-600-185

Achieving Prosperity for All

## 3.3.    ETP UPDATE ADMISSION REST API DETAILS

The details of the API are as follows: Note that the URL will differ as we go along the development to live environment stage.

- API
  The API to send the POST request:
  https://tef1.gov.bw/api/update/studentadmissions?_format=hal_json

  https://tef2.gov.bw/api/update/studentadmissions?_format=hal_json

  https://tef.gov.bw/api/update/studentadmissions?_format=hal_json

- Authorization
  Type of Authorization: Basic Auth
  Each institution should have an account on DTEF eservices application, and these credentials should be used as authentication.
  Username: xxxxxxxxx
  Password: xxxxxxxxxx

- Headers
  3. **X-CSRF-Token**
     Issue a GET request against https://tef2.gov.bw/rest/session/token
     In response, you will get a jumble of letters and numbers (e.g. Eh1INrGyEUNBog5ZL2o-dHFPnLoseIKCcL35aVSGg94); this is your X-CSRF-Token
  4. **Content-Type**
     Use the following as the content-type header:
     Content-type: application/hal+json

| X-CSRF-Token | cZkAIFbf8dODMGFNWH9ID9rU4nRAyGYfqRgiwd4I-Ik |
|---|---|
| Content-Type | application/hal+json |

An example of the request header with its values

- Body
  To update a node for student admission on eservices, send a PACTH request with the following data structure.

```
{
  "id": [
    {
      "value": "317518323"
    }
  ],
  "institution": [
    {
      "value": "UNIVERSITY OF BOTSWANA"
    }
  ],
  "institution_program_code": [
    {
      "value": "BS001"
    }
  ],
  "surname": [
    {
      "value": "THOMAS"
    }
  ],
  "firstname": [
    {
      "value": "KEABETSWE"
    }
  ],
  "program_name": [
    {
      "value": "BACHELOR OF ACCOUNTANCY"
    }
  ],
  "new_institution_program_code": [
    {
      "value": "BSC"
    }
  ],
  "new_program_name": [
    {
      "value": "BACHELOR OF SCIENCE"
    }
  ],
  "program_duration": [
    {
      "value": "3"
    }
  ],
  "start_date": [
    {
      "value": "31/08/2022"
    }
  ],
  "completion_date": [
    {
      "value": "08/15/2025"
    }
  ],
  "entry_level": [
    {
      "value": "1"
    }
  ],
  "cost": [
    {
      "value": "10000"
    }
  ]
}
```

Above is an example of a JSON body.

For the first part of the JSON (id, institution, institution_program_code), these values are required to update an admission, they should be the same as those of the target admission to being updated and should always be included. All other values are optional and should only be included when they are being updated.

```
{
  "id": [
    {
      "value": "317510323"
    }
  ],
  "institution": [
    {
      "value": "UNIVERSITY OF BOTSWANA"
    }
  ],
  "institution_program_code": [
    {
      "value": "BSB01"
    }
  ],
```

The value for institution should be the same as from SLMS database.

```
"institution": [
  {
    "value": "UNIVERSITY OF BOTSWANA"
  }
],
```

The value for institution program code should be the same as from SLMS database.

```
"institution_program_code": [
  {
    "value": "BSB01"
  }
],
```

To update programme on admission, add the three required fields as they are on the admission and add the following fields with the new program details, the value for new institution program code should be the same as from SLMS database.

```
"new_institution_program_code": [
  {
    "value": "BSC"
  }
],
"new_program_name": [
  {
    "value": "BACHELOR OF SCIENCE"
  }
],
```

Refer to the attached **patch-student-admission.json** file for an example of the body, on how each record could look like.

**NB:** Institution and ID cannot be updated

## 3.4. ADMISSIONS RESPONSES

The following are the responses to be expected from the API requests as per the given scenarios

**Add new admission**

5. Admission successfully added/captured

JSON response containing admission payload details with the first 13 lines looking as example below

```
{
    "_links": {
        "self": {
            "href": "http://tef2.gov.bw/node/90240?_format=hal_json"
        },
        "type": {
            "href": "http://tef2.gov.bw/rest/type/node/program_of_study"
        },
        "http://tef2.gov.bw/rest/relation/node/program_of_study/revision_uid": [
            {
                "href": "http://tef2.gov.bw/user/2222?_format=hal_json"
            }
        ],
```

6. Invalid ID number provided

JSON response with error message as below

```
{
  "error": " ID does not match any in Omang or Birth Registration Systems"
}
```

7. Incorrect Institution name provided

JSON response with error message as below

```
{
  "error": " Could not find the institution on the DTEF system"
}
```

8.  Incorrect Programme name provided

JSON response with error message as below

```
{
  "error": " Could not find the corresponding program on DTEF system"
}
```

9.  Duplicate admission attempt

Error message as below

```
{
  "error": "Duplicate Student Admission!"

}
```

**Update admission**

10. Admission successfully updated

JSON response containing admission payload details with the first 13 lines looking like example below

```
{
    "_links": {
        "self": {
            "href": "http://tef2.gov.bw/node/90240?_format=hal_json"
        },
        "type": {
            "href": "http://tef2.gov.bw/rest/type/node/program_of_study"
        },
        "http://tef2.gov.bw/rest/relation/node/program_of_study/revision_uid": [
            {
                "href": "http://tef2.gov.bw/user/2222?_format=hal_json"
            }
        ],
```

11. ID number, Institution or Institution program code not provided

JSON response with error message as below

```
{
  "error": "national ID, institution and program code must be provided to update admission"
}
```

12. Invalid ID number provided

JSON response with error message as below

```
{
  "error": "Admission does not exist on DTEF system"
}
```

13. Incorrect Institution name provided

JSON response with error message as below

```
{
  "error": " Could not find the institution on the DTEF system"
}
```

14. Incorrect new institution programme code provided

JSON response with error message as below

```
{
  "error": " Could not find the corresponding program on DTEF system"
}
```

15. Incorrect institution programme code provided

JSON response with error message as below

```
{
  "error": "Admission does not exist on DTEF system"
}
```

## 4. INTERFACE SPECIFICATION FOR REGISTER SUBMISSION

### 4.1. INFORMATION REQUIRED FROM ETPS – REGISTER

The API shared for the register requires the information stated in the table below and it is to be provided by a training institution as they process their register submissions for Batswana students sponsored by DTEF.

| Item/Field | Comments | Value, Data Type and/or Size |
|---|---|---|
| id | Botswana National or Birth Certificate Number | Number |
| tr | Beneficiary's DTEF tr number | Number |
| names | Beneficiary's names as they appear in ID card | Varchar |
| surname | Beneficiary's surname as it appears in ID card | Varchar |
| prog_name | Program name of program beneficiary is currently studying as specified in DTEF's SLMS | Varchar |
| prog_code | Program code of program beneficiary is currently studying as specified in DTEF's SLMS | Varchar |
| inst | Institution name beneficiary is currently studying at as specified in DTEF's SLMS | Varchar |

| campus | Institution 's campus name beneficiary is currently studying at | Varchar |
|--------|------------------------------------------------------------------|---------|
| accomo | Beneficiary's accommodation status at intitution | on / off |
| study_year | The year level in which the beneficiary is being registered to study in | Number (3) |
| study_semester | The semester in which the beneficiary is being registered to study in | Number (1) |
| sem_start_date | The date in which the semester the beneficiary is being registered for will commence | Date (d-m-y) |
| sem_end_date | The date in which the semester the beneficiary is being registered for will complete | Date (d-m-y) |
| modules | Comma separated values of module codes in which the beneficiary will be undertaking | Varchar |

ETPs are expected to send/post this information into the eServices platform using the REST API shared in the next section.

## 4.2.    ETP ADD NEW REGISTER RECORD REST API DETAILS

The details of the API are as follows: Note that the URL will differ as we go along the development to live environment stage.

- API
  The API to send the POST request:
  https://tef1.gov.bw/api/post/studentregistration?_format=hal_json
  https://tef2.gov.bw/api/post/studentregistration?_format=hal_json
  https://tef.gov.bw/api/post/studentregistration?_format=hal_json

- Authorization
  Type of Authorization: Basic Auth
  Each institution should have an account on DTEF eservices application and these credentials should be used as authentication.
  Username: xxxxxxxxx
  Password: xxxxxxxxxx

- Headers
  5. **X-CSRF-Token**
     Issue a GET request against ENVIRONMENT_URL/rest/session/token
     In response, you will get a jumble of letters and numbers (e.g. Eh1INrGyEUNBog5ZL2o-dHFPnLoseIKCcL35aVSGg94); this is your X-CSRF-Token
  6. **Content-Type**
     Use the following as the content-type header:
     Content-type: application/hal+json

| X-CSRF-Token | cZkAlFbf8dODMGFNWH9ID9rU4nRAyGYfqRgiwd4I-Ik |
|---|---|
| Content-Type | application/hal+json |

An example of the request header with its values

- Body
  To create a node for student register on eservices, send a POST request with the following data structure.

```json
{
    "id": [
        {
            "value": "317510322"
        }
    ],
    "tr": [
        {
            "value": "212648"
        }
    ],
    "names": [
        {
            "value": "Keabetswe Thomas"
        }
    ],
    "surname": [
        {
            "value": "Mambule"
        }
    ],
    "prog_name": [
        {
            "value": "Web and Mobile Technologies"
        }
    ],
    "prog_code": [
        {
            "value": "mtec"
        }
    ],
    "inst": [
        {
            "value": "Botswana Accountancy College"
        }
    ],
    "campus": [
        {
            "value": "Main Campus"
        }
    ],
    "accomo": [
        {
            "value": "on"
        }
    ],
```

OUR VISION
"A Knowledge based Society enabling Prosperity for all"
Collaboration Co-production Consuming
TOLL FREE: 0800-600-185

Achieving Prosperity for All

```json
    "status": [
        {
            "value": "continue"
        }
    ],
    "study_year": [
        {
            "value": "3"
        }
    ],
    "study_semester": [
        {
            "value": " 1"
        }
    ],
    "sem_start_date": [
        {
            "value": "26-2-2023"
        }
    ],
    "sem_end_date": [
        {
            "value": "26-2-2023"
        }
    ],
    "modules": [
        {
            "value": "cpdma, ooad, awd"
        }
    ]
}
```

Above is an example of a JSON body.

Refer to the attached **post-student-registration.json** file for an example of the body, on how each record should look like.

## 4.3.    REGISTERS RESPONSES

The following are the responses to be expected from the API requests as per the given scenarios

1. Registration record successfully added.

```json
{
    "Success": "Registered 317510322"
}
```

2. Invalid ID number format provided.

```
{
    "error": "Invalid ID number value format provided, please provide a numerical value"
}
```

3. Invalid ID number provided.

```
{
    "error": "ID does not match any in Omang or Birth Registration Systems"
}
```

4. Invalid Tr Number format provided.

```
{
    "error": "Invalid TR number value format provided, please provide a numerical value"
}
```

5. Tr number provided not matching presented beneficiary's.

```
{
    "error": "Invalid TR number provided for beneficiary"
}
```

6. Invalid TR number provided.

```
{
    "error": "Invalid TR number provided"
}
```

7. Accommodation status provided is not 'on' or 'off'.

```
{
    "error": "Invalid accommodation status provided. Please provide either "on" or "off" only"
}
```

8. Invalid study year or study semester values provided.

```
{
    "error": "Invalid Study year value format provided, please provide a numerical value"
}
```

```
{
    "error": "Invalid Study semester value format provided, please provide a numerical value"
}
```

9. Duplicate registration.

```
{
    "error": "Duplicate registration prevented for 317510322"
}
```

10. When beneficiary's status does not permit registration.

```
{
    "error": "Beneficiary cannot be registered for this semester as they are either completed
or terminated"
}
```

11. When beneficiary has failed or terminated by institution.

```
{
    "error": " Beneficiary cannot be registered for this semester as they have either not met
DTEF's proceed requirements or been terminated by institution"
}
```

# 5. INTERFACE SPECIFICATION FOR ACADEMIC RESULTS SUBMISSION

## 5.1. INFORMATION REQUIRED FROM ETPS – ACADEMIC RESULTS

The APIs for results requires the following information to be provided by a training institution as they submit the students' academic results for Batswana students sponsored by DTEF.

| Item/Field | Comments | Value, Data Type and/or Size |
|---|---|---|
| id | Botswana National or Birth Certificate Number | Number |
| tr | Beneficiary's DTEF tr number | Number |
| names | Beneficiary's names as they appear in ID card | Varchar |
| surname | Beneficiary's surname as it appears in ID card | Varchar |
| study_year | The year level in which the beneficiary results are being submitted for | Number (4) |
| study_semester | The semester in which the beneficiary results are being submitted for | Number (2) |
| Status | The status of the beneficiary at the institution. | Terminate or Continue |
| Modules | Comma separated values of module codes which the beneficiary results are being submitted for | Varchar |

AGILEUM
Flexible Services. Digital.

| results | Comma separated values of module results in the same order as modules. E.g. if module code cpdma is first in modules its results should be first in results. | p for pass and f for fail |
|---------|------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------|

ETPs are expected to send/post this information into the eServices platform using the REST API shared in the next section.

## 5.2.    ETP ADD NEW RESULTS REST API DETAILS

The details of the API are as follows: Note that the URL will differ as we go along the development to live environment stage.

- API
  The API to send the POST request:
  https://tef1.gov.bw/api/post/studentresults?_format=hal_json
  https://tef2.gov.bw/api/post/studentresults?_format=hal_json
  https://tef.gov.bw/api/post/studentresults?_format=hal_json

- Authorization
  Type of Authorization: Basic Auth
  Each institution should have an account on DTEF eservices application and these credentials should be used as authentication.
  Username: xxxxxxxxx
  Password: xxxxxxxxxx

- Headers
  7. **X-CSRF-Token**
     Issue a GET request against ENVIRONMENT_URL/rest/session/token In response, you will get a jumble of letters and numbers (e.g. Eh1INrGyEUNBog5ZL2o-dHFPnLoseIKCcL35aVSGg94); this is your X-CSRF-Token
  8. **Content-Type**
     Use the following as the content-type header:
     Content-type: application/hal+json

| X-CSRF-Token | cZkAIFbf8dODMGFNWH9ID9rU4nRAyGYfqRgiwd4I-Ik |
|---|---|
| Content-Type | application/hal+json |

An example of the request header with its values

- Body
  To create a node for student results on eservices, send a POST request with the following data structure.

```
{
    "id": [
        {
            "value": "317510322"
        }
    ],
    "tr": [
        {
            "value": "212648"
        }
    ],
    "names": [
        {
            "value": "Keabetswe Thomas"
        }
    ],
    "surname": [
        {
            "value": "Mambule"
        }
    ],
    "study_year": [
        {
            "value": "4"
        }
    ],
    "study_semester": [
        {
            "value": "1"
        }
    ],
    "modules": [
        {
            "value": "aad, java, cipa"
        }
    ],
    "results": [
        {
            "value": "p, p, f"
        }
    ],
    "status": [
        {
            "value": "continue"
        }
    ]
}
```

Above is an example of a JSON body.

Refer to the attached **post-student-results.json** file for an example of the body, on how each record will look like.

## 5.3.        ACADEMIC RESULTS RESPONSES

The following are the responses to be expected from the API requests as per the given scenarios.

### Add new results.

1. Results successfully submitted.

```
{
    "Success": "Results submitted for 317510322"
}
```

2. Invalid ID number format provided.

```
{
    "error": "Invalid ID number value format provided, please provide a numerical value"
}
```

3. Invalid ID number provided.

```
{
    "error": "ID does not match any in Omang or Birth Registration Systems"
}
```

4. Invalid Tr Number format provided.

```
{
    "error": "Invalid TR number value format provided, please provide a numerical value"
}
```

16. Tr number provided not matching presented beneficiary's.

```
{
    "error": "Invalid TR number provided for beneficiary"
}
```

17. Invalid TR number provided.

```
{
    "error": "Invalid TR number provided"
}
```

7. Invalid study year or study semester values provided.

```
{
```

```
    "error": "Invalid Study year value format provided, please provide a numerical value"
}


{
    "error": "Invalid Study semester value format provided, please provide a numerical value"
}
```

8. Duplicate result submission.

```
{
    "error": " Duplicate results submission prevented for for 317510322"
}
```

9. When beneficiary's status does not permit result submission.

```
{
    "error": "Beneficiary results cannot be submitted for this semester as they are either completed or terminated"
}
```

10. When the number of results and number of modules don't match.

```
{
    "error": " Provided number of results does not match number of modules provided"
}
```

11. When invalid result format are provided instead of p or f

```
{
    "error": " Invalid results provided, please provide comma separated values. "p" for pass and "f" for fail"
}
```

12. When results study year does not match study year of the latest register

```
{
    "error": " Study year value does not match that on the latest register"
}
```

13. When results study semester does not match study semester of the latest register

```
{
    "error": " Study semester value does not match that on the latest register"
}
```

14. When modules in the results don't match those provided during previous registration.

```
{
    "error": " Provided modules dont match the ones beneficiary was previously registered for.
Please provide them in the same order as provided during registration."
}
```

15. When a beneficiary has never been registered before.

```
{
    "error": " Beneficiary has not been registered before"
}
```

16. When an invalid status has been provided.

```
{
    "error": " Invalid value provided for Status field, please provide either, "continue",
"defer" or "terminate"
}
```

**FINAL INTERFACE SPECIFICATIONS DOCUMENT FOR SLMS and ETPs – Acceptance Certificate**

Version – 2.1

## APPROVAL TABLE

| Approved & Accepted By: | Manager, AGILEUM (PTY) | Technical Reference Group Chairperson |
|---|---|---|
| Name: | | |
| Signature: | _____ <br><br> _____ | _____ <br><br> _____ |
| | Date: ____/____/_____ | Date: ____/____/_____ |
| **Witnessed By:** | Project Manager, AGILEUM (PTY) LTD | Project Manager, SLMS |
| Name: | _____ | _____ |
| Signature: | _____ | _____ |
| | Date: ____/____/_____ | Date: ____/____/_____ |

KTM

**OUR VISION**
**"A Knowledge based Society enabling Prosperity for all"**
**Collaboration Co-production Consuming**
**TOLL FREE: 0800-600-185**

Achieving Prosperity for All