



RMS_API v1 OAS3

/swagger/v1/swagger.json

BookChapter



GET

/api/BookChapter Get all book chapter records in the database.

Parameters

[Try it out](#)

No parameters

Responses

Code

Description

Links

200

Success

No links

Media type

text/plain



Controls Accept header.

[Example Value](#) | [Schema](#)

```
[  
  {  
    "pId": 0,  
    "pages": 0,  
    "year": 0,  
    "bkTitle": "string",  
    "chTitle": "string",  
    "publisher": "string"  
  }  
]
```

GET

/api/BookChapter/{pId} Get all book chapters belonging to a specific project.

Can be empty if pId supplied has no associated book chapters

Parameters

Try it out

Name	Description
------	-------------

pld * required

`integer($int32)` Project ID
(path)

pld - Project ID

Responses

Code	Description	Links
------	-------------	-------

200

Success

No links

Media type

text/plain



Controls Accept header.

[Example Value](#) | [Schema](#)

```
[  
  {  
    "pId": 0,  
    "pages": 0,  
    "year": 0,  
    "bkTitle": "string",  
    "chTitle": "string",  
    "publisher": "string"  
  }  
]
```

POST

/api/BookChapter/add Add a book chapter to a specific project.

Parameters

Try it out

No parameters

Request body

multipart/form-data

PId
`integer($int32)`

Pages
`integer($int32)`

Year
`integer($int32)`

BkTitle
`string`

ChTitle
`string`

Publisher
`string`

Responses

Code	Description	Links
200	Success	<i>No links</i>

POST `/api/BookChapter/edit` Edit a book chapter record.

Record found using old_bkTitle and old_chTitle which make up the primary key.

Parameters

[Try it out](#)

No parameters

Request body

`multipart/form-data` ▾

PId
`integer($int32)`

Pages

```
integer($int32)

Year
integer($int32)

BkTitle
string

ChTitle
string

Publisher
string

old_chTitle
string      Current chapter title (might be changed)

old_bkTitle
string      Current book title (might be changed)
```

Responses

Code	Description	Links
200	Success	<i>No links</i>

DELETE /api/BookChapter/{bkTitle}/{chTitle} Delete a book chapter record from the database.

bkTitle and chTitle used to uniquely identify the record

Parameters

[Try it out](#)

Name	Description
------	-------------

bkTitle * required
string
(path)
Book's title

bkTitle - Book's title

Name**Description****chTitle** * required

string

(path)

Chapter's title

chTitle - Chapter's title

Responses**Code****Description****Links**

200

Success

No links

CArticle

**GET**

/api/CArticle Get all conference articles in the database.

Parameters**Try it out**

No parameters

Responses**Code****Description****Links**

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header.

[Example Value](#) | [Schema](#)

```
[
  {
    "pId": 0,
    "pages": 0,
    "year": 0,
    "cname": "string",
    "title": "string",
    "publisher": "string",
    "location": "string"
  }
]
```

GET	/api/CArticle/{pId} Get conference articles belonging to a specific project ID.										
Can be empty if project has no conference articles											
Parameters	Try it out										
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Name</th><th>Description</th></tr> </thead> <tbody> <tr> <td>pId * required</td><td></td></tr> <tr> <td>integer(\$int32)</td><td>Project ID</td></tr> <tr> <td>(path)</td><td></td></tr> <tr> <td colspan="2" style="text-align: center;">pId - Project ID</td></tr> </tbody> </table>		Name	Description	pId * required		integer(\$int32)	Project ID	(path)		pId - Project ID	
Name	Description										
pId * required											
integer(\$int32)	Project ID										
(path)											
pId - Project ID											
Responses											
Code	Description	Links									

Code	Description	Links
200	<p>Success</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">text/plain</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "pId": 0, "pages": 0, "year": 0, "cname": "string", "title": "string", "publisher": "string", "location": "string" }]</pre>	No links

POST /api/CArticle/add Add conference article to a specific project.

Parameters
[Try it out](#)

No parameters

Request body
multipart/form-data

PId
`integer($int32)`

Pages
`integer($int32)`

Year
`integer($int32)`

Cname
`string`

Title
`string`

Publisher
`string`

Location

string

Responses

Code	Description	Links
200	Success	<i>No links</i>

POST

/api/CArticle/edit Edit a conference article's details.

old_cname and old_title are used to uniquely identify the record.

Parameters

[Try it out](#)

No parameters

Request body

multipart/form-data ▾

PId

integer(\$int32)

Pages

integer(\$int32)

Year

integer(\$int32)

Cname

string

Title

string

Publisher

string

Location

string

old_title

string

Current conference article title (could be changed)

old_cname

string

Current conference name (could be changed)

Responses

Code	Description	Links
200	Success	No links

DELETE /api/CArticle/{cname}/{title} Delete a conference article record from the database.

cname and title are used to uniquely identify the record.

Parameters

[Try it out](#)

Editorial

Name

Description



GET

/api/Editorial required Get all editorials in the database.

Conference name

string

Parameters

cname - Conference name

[Try it out](#)

No parameters
title * required

string

Article's title

(path)

title - Article's title

Responses

Code	Description	Links
Responses		

Code	Description	Links
200	Success	No links

Media type

text/plain



Controls Accept header.

[Example Value](#) | [Schema](#)

```
[  
  {  
    "activity": "string",  
    "publication": "string",  
    "url": "string",  
    "date": "string"  
  }  
]
```

GET

/api/Editorial/{empId} Get all editorials done by a certain employee.

Parameters

[Try it out](#)

Name	Description
------	-------------

empId * required

`integer($int32)` Employee ID
(path)

empId - Employee ID

Responses

Code	Description	Links
------	-------------	-------

Code	Description	Links
200	<p>Success</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">text/plain</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "activity": "string", "publication": "string", "url": "string", "date": "string" }]</pre>	No links

POST /api/Editorial/add Add editorial done by a specific employee.

Parameters	Try it out
Name	Description
empld <small>integer(\$int32)</small>	Employee ID <small>(query)</small>
	empld - Employee ID
Request body	multipart/form-data
Activity	
Publication	
Url	
Date	

Responses

Code	Description	Links
200	Success	<i>No links</i>

POST

/api/Editorial/edit Edit an editorial's details.

old_activity and old_publication are used to uniquely identify the existing record in the database.

Parameters

[Try it out](#)

No parameters

Request body

[multipart/form-data](#) ▾

Activity

`string`

Publication

`string`

Url

`string`

Date

`string`

old_activity

`string`

Current activity details (could be changed)

old_publication

`string`

Current publication details (could be changed)

Responses

Code	Description	Links
200	Success	No links

DELETE`/api/Editorial/{activity}/{publication}` Delete an editorial record from the database.

activity and publication are used to uniquely identify the record.

Parameters

Try it out

Name	Description
------	-------------

activity * required**string***(path)*

activity

publication * required**string***(path)*

publication

Responses

Code	Description	Links
200	Success	No links

Employee

**GET**`/api/Employee` Get all employees in the database.

Parameters

Try it out

No parameters

Responses

Code	Description	Links
200	Success	No links

Media type

▼

Controls Accept header.

[Example Value](#) | [Schema](#)

```
[  
  {  
    "fname": "string",  
    "sname": "string",  
    "department": "string",  
    "position": "string",  
    "qualification": "string",  
    "interests": "string",  
    "phone": "string",  
    "email": "string",  
    "photo": "string",  
    "empId": 0  
  }  
]
```

GET

/api/Employee/{email}/{empId} Get a specific employee record.

Employee identified using their unique email address or employee ID.

Parameters

[Try it out](#)

Name	Description
email * required <small>string (path)</small>	Email Address <i>Default value :</i> <input type="text" value="email - Email Address"/>

Name**Description****empId** * required

integer(\$int32)

(path)

Employee ID

Default value : 0

empId - Employee ID

Responses**Code****Description****Links**

200

Success

No links

Media type

text/plain



Controls Accept header.

[Example Value](#) | [Schema](#)

```
[  
  {  
    "fname": "string",  
    "sname": "string",  
    "department": "string",  
    "position": "string",  
    "qualification": "string",  
    "interests": "string",  
    "phone": "string",  
    "email": "string",  
    "photo": "string",  
    "empId": 0  
  }  
]
```

POST

/api/Employee/add Add an employee to the database.

Parameters[Try it out](#)

No parameters

Request body

multipart/form-data

Fname

`string`

Sname

`string`

Department

`string`

Position

`string`

Qualification

`string`

Interests

`string`

Phone

`string`

Email

`string`

Photo

`string`

Empld

`integer($int32)`

Responses

Code	Description	Links
200	Success	<i>No links</i>

POST

`/api/Employee/edit` Edit an employee record.

Employee identified using their empld

Parameters

Try it out

No parameters

Request body

multipart/form-data ▾

Fname

`string`

Sname

`string`

Department

`string`

Position

`string`

Qualification

`string`

Interests

`string`

Phone

`string`

Email

`string`

Photo

`string`

Empld

`integer($int32)`

Responses

Code	Description	Links
------	-------------	-------

200

Success

No links

POST`/api/Employee/author` Get authors of a publication. Used by all publication types.

An author is an employee. type identifies the type of publication id1 and id2 uniquely identify the record.

Parameters

[Try it out](#)

No parameters

Request body

[multipart/form-data](#) ▾

id1
string journal name, conference name, or book title

id2
string article title or chapter title

type
string Type of publication: jArticle, cArticle, or bookChapter

Responses

Code**Description****Links**

Code	Description	Links
200	<p>Success</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">text/plain</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "fname": "string", "surname": "string", "department": "string", "position": "string", "qualification": "string", "interests": "string", "phone": "string", "email": "string", "photo": "string", "empId": 0 }]</pre>	No links

POST

/api/Employee/author/add Associate an author with a publication they wrote.

An author is an employee. type identifies the type of publication id1 and id2 uniquely identify the record.

Parameters

[Try it out](#)

No parameters

Request body

[multipart/form-data](#)

emplId
`integer($int32)` Employee ID

id1
`string` journal name, conference name, or book title

id2
`string` article title or chapter title

type
Type of publication: jArticle, cArticle, or bookChapter

`string`

Responses

Code	Description	Links
200	Success	<i>No links</i>

POST

`/api/Employee/editor` Get editors/reviewers of an editorial

An editor is an employee. activity and publication are used to uniquely identify the record.

Parameters

[Try it out](#)

No parameters

Request body

`multipart/form-data` ▾

`activity`

`string`

`publication`

`string`

Responses

Code	Description	Links

Code	Description	Links
200	<p>Success</p> <p>Media type</p> <div><p>text/plain</p><p>▼</p></div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "fname": "string", "sname": "string", "department": "string", "position": "string", "qualification": "string", "interests": "string", "phone": "string", "email": "string", "photo": "string", "empId": 0 }]</pre>	No links

Funder



GET

/api/Funder Get all project funders in the database.

Parameters

[Try it out](#)

No parameters

Responses

Code	Description	Links

Code	Description	Links
200	<p>Success</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">text/plain</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "fId": 0, "name": "string", "organisation": "string", "email": "string" }]</pre>	No links

GET	/api/Funder/{pId} Get funders of a specific project								
Cannot be empty because each project has at least one funder.									
Parameters	Try it out								
<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>pId * required</td> <td></td></tr> <tr> <td>integer(\$int32)</td> <td>pld</td></tr> <tr> <td>(path)</td> <td></td></tr> </tbody> </table>		Name	Description	pId * required		integer(\$int32)	pld	(path)	
Name	Description								
pId * required									
integer(\$int32)	pld								
(path)									

Code	Description	Links

Code	Description	Links
200	<p>Success</p> <p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">text/plain</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "fId": 0, "name": "string", "organisation": "string", "email": "string" }]</pre>	No links

POST /api/Funder/add Add a funder to a certain project.

pld is used to associate a funder to a project.

Parameters	Try it out
No parameters	
Request body	multipart/form-data ▾
Fld <code>integer(\$int32)</code>	
Name <code>string</code>	
Organisation <code>string</code>	
Email <code>string</code>	
pld <code>integer(\$int32)</code>	Project ID

Responses

Code	Description	Links
200	Success	<i>No links</i>

POST

/api/Funder/edit Edit a funder's details.

Funder identified using the fId supplied

Parameters

[Try it out](#)

No parameters

Request body

[multipart/form-data](#) ▾

fId

`integer($int32)`

Name

`string`

Organisation

`string`

Email

`string`

Responses

Code	Description	Links
------	-------------	-------

Code	Description	Links
200	Success	No links

DELETE`/api/Funder/{pId}/{fId}` Delete a funder from a specified project.

pId identifies the project from which to remove the funder.

Parameters

Try it out

Name	Description
------	-------------

pId * required

`integer($int32)` Project ID
(path)

pId - Project ID

fId * required

`integer($int32)` Funder's ID
(path)

fId - Funder's ID

Responses

Code	Description	Links
------	-------------	-------

200

Success

No links

JArticle

**GET**`/api/JArticle` Get all journal articles in the database.

Parameters

Try it out

No parameters

Responses

Code	Description	Links
200	Success	<i>No links</i>

Media type

text/plain



Controls Accept header.

[Example Value](#) | [Schema](#)

```
[  
  {  
    "pId": 0,  
    "pages": 0,  
    "year": 0,  
    "jname": "string",  
    "title": "string",  
    "volume": 0,  
    "issue": 0  
  }  
]
```

GET

/api/JArticle/{pId} Get journal articles belonging to a specific project ID.

Can be empty if project has no journal articles.

Parameters

Try it out

Name Description

pId * required

`integer($int32)` Project ID
(path)

pId - Project ID

Responses

Code	Description	Links
200	Success	No links

Media type

`text/plain` ▾

Controls Accept header.

[Example Value](#) | [Schema](#)

```
[  
  {  
    "pId": 0,  
    "pages": 0,  
    "year": 0,  
    "jname": "string",  
    "title": "string",  
    "volume": 0,  
    "issue": 0  
  }  
]
```

POST

/api/JArticle/add Add journal article to a specific project.

Parameters

[Try it out](#)

No parameters

Request body

`multipart/form-data` ▾

PId

`integer($int32)`

Pages

`integer($int32)`

Year

`integer($int32)`

Jname

`string`

Title

`string`

Volume
`integer($int32)`

Issue
`integer($int32)`

Responses

Code	Description	Links
200	Success	<i>No links</i>

POST `/api/JArticle/edit` Edit a journal article's details.

old_jname and old_title are used to uniquely identify the record.

Parameters

[Try it out](#)

No parameters

Request body

`multipart/form-data` ▾

PId
`integer($int32)`

Pages
`integer($int32)`

Year
`integer($int32)`

Jname
`string`

Title
`string`

Volume
`integer($int32)`

Issue
`integer($int32)`

old_title
`string` Current journal article title (could be changed)

old_jname
`string` Current journal name (could be changed)

Responses

Code	Description	Links
200	Success	<i>No links</i>

DELETE `/api/JArticle/{jname}/{title}` Delete a journal article record from the database.

jname and title are used to uniquely identify the record.

Parameters

[Try it out](#)

Name Description

jname * required
`string` Journal name
(path)

jname - Journal name

title * required
`string` Article's title
(path)

title - Article's title

Responses

Code	Description	Links
200	Success	No links

Project



GET

/api/Project Get All Projects in BIUST.

Parameters

Try it out

No parameters

Responses

Code	Description	Links
------	-------------	-------

200	Success	No links
-----	---------	----------

Media type

text/plain



Controls Accept header.

Example Value | Schema

```
[  
  {  
    "pId": 0,  
    "title": "string",  
    "leader": 0,  
    "fsource": "string",  
    "famout": 0  
  }  
]
```

GET

/api/Project/{empId} Get All Projects done by a certain employee.

Could be an empty list if empld is not found.

Parameters

[Try it out](#)

Name	Description
------	-------------

empld * required <code>integer(\$int32)</code> (path)	Employee ID Number
--------------------------------------------------------------------	--------------------

empld - Employee ID Number

Responses

Code	Description	Links
------	-------------	-------

200	Success	<i>No links</i>
-----	---------	-----------------

Media type

`text/plain`



Controls Accept header.

[Example Value](#) | [Schema](#)

```
[  
  {  
    "pId": 0,  
    "title": "string",  
    "leader": 0,  
    "fsource": "string",  
    "famout": 0  
  }  
]
```

POST

`/api/Project/add` Add a project done by a certain employee.

Parameters

[Try it out](#)

No parameters

Request body

`multipart/form-data`

title
`string` Title of the project

leader
`integer($int32)` Project leader's employee ID

famount
`integer($int32)` Funding amount

fsource
`string` Source of funding

empld
`integer($int32)` Employee ID

fld
`integer($int32)` Funder's ID

Responses

Code	Description	Links
200	Success	<i>No links</i>

POST /api/Project/edit Edit an existing project's details.

Parameters

[Try it out](#)

No parameters

Request body

`multipart/form-data` ▾

title
`string` Title of the project

leader
`integer($int32)` Project leader's employee ID

```
famount      Funding amount  
integer($int32)  
  
fsource      Source of funding  
string  
  
pld          Project ID  
integer($int32)
```

Responses

Code	Description	Links
200	Success	<i>No links</i>

DELETE /api/Project/{projectId} Delete a project from the database.

Finds project using the pld supplied.

Parameters

[Try it out](#)

Name	Description
------	-------------

pld * required	<input type="text" value="pld"/>
integer(\$int32)	
(path)	

Responses

Code	Description	Links
200	Success	<i>No links</i>

Schemas

```
BookChapter ▼ {  
    pid          integer($int32)  
    pages        integer($int32)  
    year         integer($int32)  
    bkTitle      string  
                nullable: true  
    chTitle      string  
                nullable: true  
    publisher    string  
                nullable: true  
}  
}
```

```
CArticle ▼ {  
    pid          integer($int32)  
    pages        integer($int32)  
    year         integer($int32)  
    cname        string  
                nullable: true  
    title        string  
                nullable: true  
    publisher    string  
                nullable: true  
    location     string  
                nullable: true  
}  
}
```

```
Editorial ▼ {  
    activity     string  
                nullable: true  
    publication string  
                nullable: true  
    url          string  
                nullable: true  
    date         string  
                nullable: true  
}  
}
```

```
Employee ▼ {  
    fname          string  
    nullable: true  
    surname        string  
    nullable: true  
    department     string  
    nullable: true  
    position       string  
    nullable: true  
    qualification   string  
    nullable: true  
    interests      string  
    nullable: true  
    phone          string  
    nullable: true  
    email          string  
    nullable: true  
    photo          string  
    nullable: true  
    empId         integer($int32)  
}  
}
```

```
Funder ▼ {  
    fid            integer($int32)  
    name           string  
    nullable: true  
    organisation   string  
    nullable: true  
    email          string  
    nullable: true  
}  
}
```

```
JArticle ▼ {  
    pid            integer($int32)  
    pages          integer($int32)  
    year           integer($int32)  
    jname          string  
    nullable: true  
    title          string  
    nullable: true  
    volume         integer($int32)  
    issue          integer($int32)  
}  
}
```

```
Project ▼ {  
    pid            integer($int32)  
    title          string  
    nullable: true  
    leader         integer($int32)  
    fsource        string  
    nullable: true  
    famout        number($double)  
}  
}
```