**API Documentation**

**User**

**1. Create User**

**REQUEST**

* **Uri: POST /users**
* **Body Parameters**
  + username - name of the user pretending to sign in (required)
  + password - password given by user (required)
* **Body**:

{ "username" : "aluno", "password" : "isel" }

**RESPONSE**

SUCCESS

* **Status**: 201 CREATED
* **Headers**: Content-type: application/json
* **Body**:

{ "id" : 1}

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.UserInputModel] value failed for JSON property username or password due to missing (therefore NULL) value for creator parameter username which is a non-nullable type",

"status": 400

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "User or password cannot be blank!",

"status": 500

}

* **Status**: 409 CONFLICT
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "UniqueAttributeException",

"title": "Unique attribute exception.",

"detail": "User aluno already exists.",

"status": 409

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: PreparedStatementCallback; bad SQL grammar [INSERT INTO apiuser (username, password) VALUES (?, ?) RETURNING id]; nested exception is org.postgresql.util.PSQLException: ERROR: relation \"apiuser\" does not exist\n Position: 13",

"status": 500

}

**2. Login User**

**REQUEST**

* **Uri: POST /users/login**
* **Body Parameters**
  + username - name of the user pretending to login (required)
  + password - password given by user (required)
* **Body**:

{ "username" : "aluno", "password" : "isel" }

**RESPONSE**

SUCCESS

* **Status**: 200 OK
* **Headers**: Content-type: application/json
* **Body**: { "id" : 1 }

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.UserInputModel] value failed for JSON property username or password due to missing (therefore NULL) value for creator parameter username which is a non-nullable type",

"status": 400

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "User or password cannot be blank!",

"status": 500

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Password is incorrect!",

"status": 500

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**3. Obtain All Users**

**REQUEST**

* **Uri: GET /users**

**RESPONSE**

SUCCESS

* **Status**: 200 OK
* **Headers**: Content-type: application/json
* **Body**:

{ "usernames": ["aluno1", "aluno2"] }

ERROR

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**4. Remove User**

**REQUEST**

* **Uri: DELETE /users/{username}**
* **Path Parameters**
  + username - name of the user that will be removed (required)

**RESPONSE**

SUCCESS

* **Status**: 204 NO-CONTENT

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.UserInputModel] value failed for JSON property username due to missing (therefore NULL) value for creator parameter username which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**PROJECT**

**1. Create Project**

**REQUEST**

* **Uri**: POST /projects
* **Headers**: Authorization: Basic Auth
* **Body Parameters**:
  + name - name of the project being created (required)
  + description - short description of the project
  + issueStates - array of possible states to this project
  + issueLabels - array of possible labels to this project
* **Body**:

{  
 "name" : "Grupo 3",  
 "description" : "Projecto do grupo 3",  
 "issueStates" : ["open", "closed", "archived"],  
 "issueLabels" : ["new\_functionality", "defect"]  
}

**RESPONSE**

SUCCESS

* **Status:** 201 CREATED
* **Headers:** Content-type: text/plain
* **Body:** Project Grupo 3 created successfully.

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.ProjectInputModel] value failed for JSON property name due to missing (therefore NULL) value for creator parameter name which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 409 CONFLICT
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "UniqueAttributeException",

"title": "Unique attribute exception.",

"detail": "Project Grupo 3 already exists."

"status": 409

}

* **Status**: 409 CONFLICT
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "UniqueAttributeException",

"title": "Unique attribute exception.",

"detail": "IssueState '${state.name}' already exists on Project '${pname}'."

"status": 409

}

* **Status**: 409 CONFLICT
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "UniqueAttributeException",

"title": "Unique attribute exception.",

"detail": "IssueLabel '${issueLabel.name}' already exists on Project '${pname}'."

"status": 409

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**2. Obtain All Projects**

**REQUEST**

* **Uri:** GET /projects
* **Headers:** 
  + Accept: application/vnd.siren+json (optional)

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:** 
  + Content-type: application/json, application/vnd.siren+json
* **Body (application/json):**

{  
 "name": "Grupo 3",  
 "description": "Projecto do grupo 3",  
 "issueStates": [  
 "open",  
 "closed",  
 "archived"  
 ],  
 "issueLabels": [  
 "new\_functionality",  
 "defect"  
 ]  
}

* **Body (application/vnd.siren+json):**

{  
 "properties": {  
 "count": 1  
},  
 "links": [  
 {  
 "rel": [  
 "self",  
 "collection",  
 "/rels/projects"  
 ],  
 "href": "/projects",  
 "title": "All Projects"  
 }  
 ],  
 "actions": [  
 {  
 "name": "/actions/post/project",  
 "href": "/projects",  
 "title": "Create project",  
 "class": [  
 "/classes/projects"  
 ],  
 "method": "POST",  
 "type": "application/json",  
 "fields": [  
 {  
 "name": "name",  
 "type": "text"  
 },  
 {  
 "name": "description",  
 "type": "text"  
 },  
 {  
 "name": "issueState",  
 "type": "IssueState[]"  
 },  
 {  
 "name": "issueLabel",  
 "type": "IssueLabel[]"  
 }  
 ]  
 }  
 ],  
 "entities": [  
 {  
 "class": [  
 "/classes/project"  
 ],  
 "properties": {  
 "name": "Grupo 3",  
 "description": "Projecto do grupo 3",  
 "issueStates": [  
 "open",  
 "closed",  
 "archived"  
 ],  
 "issueLabels": [  
 "new\_functionality",  
 "defect"  
 ]  
 },  
 "links": [  
 {  
 "rel": [  
 "self"  
 ],  
 "href": "/projects/Grupo%203",  
 "title": "Project Grupo 3"  
 }  
 ]  
 }  
 ],  
 "clazz": [  
 "/classes/projects"  
 ]  
}

ERROR

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**3. Obtain Project Details**

**REQUEST**

* **Uri:** GET /projects/{name}
* **Headers:**
  + Accept: application/vnd.siren+json (optional)
* **Path Parameters:** 
  + name - obtain the details of the given project name

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:**
  + Content-type: application/json, application/vnd.siren+json
* **Body (application/json):**

{  
 "name": "Grupo 3",  
 "description": "Projecto do grupo 3",  
 "issueStates": [  
 "open",  
 "closed",  
 "archived"  
 ],  
 "issueLabels": [  
 "new\_functionality",  
 "defect"  
 ]  
}

* **Body (application/vnd.siren+json):**

{  
 "properties": {  
 "name": "Grupo 3",  
 "description": "Projecto do grupo 3",  
 "issueStates": [  
 "open",  
 "closed",  
 "archived"  
 ],  
 "issueLabels": [  
 "new\_functionality",  
 "defect"  
 ]  
},  
 "links": [  
 {  
 "rel": [  
 "self",  
 "/rels/project"  
 ],  
 "href": "/projects/Grupo%203",  
 "title": "Project Grupo 3"  
 },  
 {  
 "rel": [  
 "/rels/projects",  
 "parent",  
 "collection"  
 ],  
 "href": "/projects",  
 "title": "All Projects"  
 },  
 {  
 "rel": [  
 "/rels/project/issues",  
 "collection"  
 ],  
 "href": "/projects/Grupo%203/issues",  
 "title": "Project Grupo 3's Issues"  
 },  
 {  
 "rel": [  
 "/rels/project/contributors",  
 "collection"  
 ],  
 "href": "/projects/Grupo%203/contributors",  
 "title": "Project Grupo 3's Contributors"  
 }  
 ],  
 "actions": [  
 {  
 "name": "/actions/put/project",  
 "href": "/projects/Grupo%203",  
 "title": "Update Project Grupo 3",  
 "class": [  
 "/classes/project"  
 ],  
 "method": "PUT",  
 "type": "application/json",  
 "fields": [  
 {  
 "name": "name",  
 "type": "text"  
 },  
 {  
 "name": "description",  
 "type": "text"  
 },  
 {  
 "name": "issueState",  
 "type": "IssueState"  
 },  
 {  
 "name": "issueLabel",  
 "type": "IssueLabel"  
 }  
 ]  
 },  
 {  
 "name": "/actions/delete/project",  
 "href": "/projects/Grupo%203",  
 "title": "Delete Project Grupo 3",  
 "class": [  
 "/classes/project"  
 ],  
 "method": "DELETE",  
 "type": "application/json"  
 },  
 {  
 "name": "/actions/patch/project/name",  
 "href": "/projects/Grupo%203/name",  
 "title": "Patch Project Grupo 3's Name",  
 "class": [  
 "/classes/project"  
 ],  
 "method": "PATCH",  
 "type": "application/json",  
 "fields": [  
 {  
 "name": "newName",  
 "type": "text"  
 }  
 ]  
 },  
 {  
 "name": "/actions/patch/project/description",  
 "href": "/projects/Grupo%203/description",  
 "title": "Patch project Grupo 3's Description",  
 "class": [  
 "/classes/project"  
 ],  
 "method": "PATCH",  
 "type": "application/json",  
 "fields": [  
 {  
 "name": "description",  
 "type": "text"  
 }  
 ]  
 }  
 ],  
 "entities": [],  
 "clazz": [  
 "/classes/project"  
 ]  
}

ERROR

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project COVID-19 not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**4. Update Project Details**

**REQUEST**

* **Uri**: PUT /projects/{name}
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project being updated
* **Body Parameters**:
  + name – new name of the project (required)
  + description - new description of the project
  + issueStates – new array of possible states to this project
  + issueLabels – new array of possible labels to this project
* **Body**:

{  
 "name" : "Grupo 3 Edited",  
 "description" : "Nova descrição do Projecto do grupo 3",  
 "issueStates" : ["open", "closed", "archived"],  
 "issueLabels" : ["exploration"]  
}

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:** Content-type: text/plain
* **Body:** Project Grupo 3 Edited updated successfully.

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.ProjectInputModel] value failed for JSON property name due to missing (therefore NULL) value for creator parameter name which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "User is not a contributor and doesn't have permission to change project Grupo 3.",

"status": 403

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project Grupo 3 not found.",

"status": 404

}

* **Status**: 409 CONFLICT
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "UniqueAttributeException",

"title": "Unique attribute exception.",

"detail": "Project COVID 19 already exists.",

"status": 409

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**5. Patch Project Name**

**REQUEST**

* **Uri**: PATCH /projects/{name}/name
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project being patched
* **Body Parameters**:
  + name – new name of the current project (required)
* **Body**:

{  
 "name" : "new Grupo 3"  
}

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:** Content-type: text/plain
* **Body:** Project Grupo 3's name updated successfully to ‘new Grupo 3’.

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.ProjectInputModel$ProjectNameInputModel] value failed for JSON property name due to missing (therefore NULL) value for creator parameter name which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project Grupo 3 not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "User is not a contributor and doesn't have permission to change project Grupo 3.",

"status": 403

}

* **Status**: 409 CONFLICT
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "UniqueAttributeException",

"title": "Unique attribute exception.",

"detail": "Project new Grupo 3 already exists.",

"status": 409

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**6. Patch Project Description**

**REQUEST**

* **Uri**: PATCH /projects/{name}/description
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project being patched
* **Body Parameters**:
  + description – new description of the current project (required)
* **Body**:

{  
 "description" : "nova descrição do grupo 3"  
}

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:** Content-type: text/plain
* **Body:** Project Grupo 3's description updated successfully.

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.ProjectInputModel$ProjectNameInputModel] value failed for JSON property name due to missing (therefore NULL) value for creator parameter name which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project Grupo 3 not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "User is not a contributor and doesn't have permission to change project Grupo 3.",

"status": 403

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**7. Delete Project**

**REQUEST**

* **Uri**: DELETE /projects/{name}
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project being deleted

**RESPONSE**

SUCCESS

* **Status:** 204 NO-CONTENT

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.ProjectInputModel$ProjectNameInputModel] value failed for JSON property name due to missing (therefore NULL) value for creator parameter name which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project Grupo 3 not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "User is not a contributor and doesn't have permission to change project Grupo 3.",

"status": 403

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**Issue**

**1. Create Issue**

**REQUEST**

* **Uri**: POST /projects/{name}/issues
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project where the issue is being created
* **Body Parameters**:
  + name - name of the issue being created (required)
  + description - short description of the issue
  + labelType - array of possible labels to this issue. Must be contained in project labels
* **Body**:

{  
 "name”: "new issue",  
 "description”: "new description",

"labelType" : [ "new\_functionality", "defect" ]  
}

**RESPONSE**

SUCCESS

* **Status:** 201 CREATED
* **Headers:** Content-type: text/plain
* **Body:** Issue 11 for project Grupo 3 created successfully.

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.IssueInputModel] value failed for JSON property name due to missing (therefore NULL) value for creator parameter name which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project InventedProject not found.",

"status": 404

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "User is not a contributor and doesn't have permission to change project Grupo 3.",

"status": 403

}

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "InvalidOperationException",

"title": "Invalid operation exception.",

"detail": "Label 'exploration' is not available in the group Grupo 3 label's list",

"status": 400

}

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "HttpMessageNotReadableException",

"title": "Bad parameter type",

"detail": "org.springframework.http.converter.HttpMessageNotReadableException: Invalid JSON input: Cannot deserialize value of type `pt.isel.daw.model.IssueLabel` from String \"nonExistingLabel\": not one of the values accepted for Enum class: [defect, exploration, new\_functionality]; nested exception is com.fasterxml.jackson.databind.exc.InvalidFormatException: Cannot deserialize value of type `pt.isel.daw.model.IssueLabel` from String \"nonExistingLabel\": not one of the values accepted for Enum class: [defect, exploration, new\_functionality]\n at [Source: (PushbackInputStream); line: 4, column: 16] (through reference chain: pt.isel.daw.controller.representation.input.IssueInputModel[\"labelType\"]->java.util.ArrayList[0])",

"status": 400

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "PSQLException",

"title": "SQL Exception",

"detail": "org.postgresql.util.PSQLException: ERROR: duplicate key value violates unique constraint \"issue\_issuelabel\_pkey\"\n Detail: Key (projectname, issueid, statetype, labeltype)=(Grupo 3, 5, open, new\_functionality) already exists.",

"status": 500

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**2. Obtain All Issues from a Project**

**REQUEST**

* **Uri:** GET /projects/{name}/issues
* **Headers:** 
  + Accept: application/vnd.siren+json (optional)
* **Path Parameters:**
  + name - name of the project in which we want to retrieve the list of issues

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:** 
  + Content-type: application/json, application/vnd.siren+json
* **Body (application/json):**

{  
 "count": 1,  
 "issues": [  
 **{** "id": 11,  
 "name": "new issue",  
 "description": "new description",  
 "createdBy": "davidwilsonluis",  
 "creationDate": "2020-07-15 21:18",  
 "closeDate": null,  
 "stateType": "open",  
 "labelType": [],  
 "pname": "Grupo 3"  
 **}** ]  
}

* **Body (application/vnd.siren+json):**

{

"properties": {  
 "count": 1  
},  
"links": [  
{  
 "rel": [  
 "self",  
 "/rels/project/issues",  
 "collection"  
 ],  
 "href": "/projects/Grupo%203/issues",  
 "title": "Project Grupo 3's Issues"  
},  
{  
 "rel": [  
 "/rels/project",  
 "parent"  
 ],  
 "href": "/projects/Grupo%203",  
 "title": "Project Grupo 3"  
}  
],  
"actions": [  
{  
 "name": "/actions/post/issue",  
 "href": "/projects/Grupo%203/issues",  
 "title": "Create Issue",  
 "class": [  
 "/classes/issues"  
 ],  
 "method": "POST",  
 "type": "application/json",  
 "fields": [  
 **{** "name": "name",  
 "type": "text"  
 **}**,  
 **{** "name": "description",  
 "type": "text"  
 **}**,  
 **{** "name": "labelType",  
 "type": "IssueLabel[]"  
 **}** ]  
}  
],  
"entities": [  
{  
 "class": [  
 "/classes/issue"  
 ],  
 "properties": **{** "id": 11,  
 "name": "new issue",  
 "description": "new description",  
 "createdBy": "davidwilsonluis",  
 "creationDate": "2020-07-15 21:18",  
 "closeDate": null,  
 "stateType": "open",  
 "labelType": [],  
 "pname": "Grupo 3"  
**}**,  
 "links": [  
 **{** "rel": [  
 "self"  
 ],  
 "href": "/projects/Grupo%203/issues/11",  
 "title": "Issue 11"  
 **}** ]  
}  
],  
"clazz": [  
"/classes/project"  
]

ERROR

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project InventedProject not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**3. Obtain Issue Details**

**REQUEST**

* **Uri:** GET /projects/{name}/issues/{id}
* **Headers:**
  + Accept: application/vnd.siren+json (optional)
* **Path Parameters:** 
  + name – name of the project where {id} issue is contained
  + id – obtain issue with id = {id}

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:**
  + Content-type: application/json, application/vnd.siren+json
* **Body (application/json):**

{

    "id": 11,

    "name": "new issue",

    "description": "new description",

    "createdBy": "davidwilsonluis",

    "creationDate": "2020-07-15 21:18",

    "closeDate": null,

    "stateType": "open",

    "labelType": [],

    "pname": "Grupo 3"

}

* **Body (application/vnd.siren+json):**

{  
 "properties": {  
 "id": 11,  
 "name": "new issue",  
 "description": "new description",  
 "createdBy": "davidwilsonluis",  
 "creationDate": "2020-07-15 21:18",  
 "closeDate": null,  
 "stateType": "open",  
 "labelType": []  
},  
 "links": [  
 {  
 "rel": [  
 "self",  
 "/rels/project/issue"  
 ],  
 "href": "/projects/Grupo%203/issues/11",  
 "title": "Issue 11"  
 },  
 {  
 "rel": [  
 "/rels/project/issues",  
 "parent",  
 "collection"  
 ],  
 "href": "/projects/Grupo%203/issues",  
 "title": "Project Grupo 3's Issues"  
 },  
 {  
 "rel": [  
 "/rels/project/issue/comments",  
 "collection"  
 ],  
 "href": "/projects/Grupo%203/issues/11/comments",  
 "title": "Issue 11's Comments"  
 }  
 ],  
 "actions": [  
 {  
 "name": "/actions/put/issue",  
 "href": "/projects/Grupo%203/issues/11",  
 "title": "Update Issue 11",  
 "class": [  
 "/classes/issue"  
 ],  
 "method": "PUT",  
 "type": "application/json",  
 "fields": [  
 {  
 "name": "name",  
 "type": "text"  
 },  
 {  
 "name": "description",  
 "type": "text"  
 },  
 {  
 "name": "labelType",  
 "type": "IssueLabel[]"  
 }  
 ]  
 },  
 {  
 "name": "/actions/patch/issue/name",  
 "href": "/projects/Grupo%203/issues/11/name",  
 "title": "Update Issue 11's Name",  
 "class": [  
 "/classes/issue"  
 ],  
 "method": "PATCH",  
 "type": "application/json",  
 "fields": [  
 {  
 "name": "name",  
 "type": "text"  
 }  
 ]  
 },  
 {  
 "name": "/actions/patch/issue/description",  
 "href": "/projects/Grupo%203/issues/11/description",  
 "title": "Update Issue 11's Description",  
 "class": [  
 "/classes/issue"  
 ],  
 "method": "PATCH",  
 "type": "application/json",  
 "fields": [  
 {  
 "name": "description",  
 "type": "text"  
 }  
 ]  
 },  
 {  
 "name": "/actions/patch/issue/state",  
 "href": "/projects/Grupo%203/issues/11/state",  
 "title": "Update Issue 11's State",  
 "class": [  
 "/classes/issue"  
 ],  
 "method": "PATCH",  
 "type": "application/json",  
 "fields": [  
 {  
 "name": "state",  
 "type": "text"  
 }  
 ]  
 },  
 {  
 "name": "/actions/delete/issue",  
 "href": "/projects/Grupo%203/issues/11",  
 "title": "Delete Issue 11",  
 "class": [  
 "/classes/issue"  
 ],  
 "method": "DELETE",  
 "type": "application/json"  
 }  
 ],  
 "entities": [],  
 "clazz": [  
 "/classes/issue"  
 ]  
}

ERROR

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project InventedProject not found.",

"status": 404

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Issue 200 of project Grupo 3 not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**4. Update Issue Details**

**REQUEST**

* **Uri**: PUT /projects/{name}/issues/{id}
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project where the issue is present
  + id – id of the issue being updated
* **Body Parameters**:
  + name – new name for the issue (required)
  + description - new description for the issue
  + labelType - array of possible labels to this issue. Must be contained in project labels
* **Body**:

{  
 "name": "new name 5",  
 "description": "new description"  
}

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:** Content-type: text/plain
* **Body:** Issue 11 updated successfully

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.ProjectInputModel] value failed for JSON property name due to missing (therefore NULL) value for creator parameter name which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "User is not a contributor and doesn't have permission to change project Grupo 3.",

"status": 403

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project Grupo 3 not found.",

"status": 404

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Issue 5 of project Grupo 3 not found.",

"status": 404

}

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "HttpMessageNotReadableException",

"title": "Bad parameter type",

"detail": "org.springframework.http.converter.HttpMessageNotReadableException: Invalid JSON input: Cannot deserialize value of type `pt.isel.daw.model.IssueLabel` from String \"nonExistingLabel\": not one of the values accepted for Enum class: [defect, exploration, new\_functionality]; nested exception is com.fasterxml.jackson.databind.exc.InvalidFormatException: Cannot deserialize value of type `pt.isel.daw.model.IssueLabel` from String \"nonExistingLabel\": not one of the values accepted for Enum class: [defect, exploration, new\_functionality]\n at [Source: (PushbackInputStream); line: 4, column: 16] (through reference chain: pt.isel.daw.controller.representation.input.IssueInputModel[\"labelType\"]->java.util.ArrayList[0])",

"status": 400

}

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "InvalidOperationException",

"title": "Invalid operation exception.",

"detail": "Label 'exploration' is not available in the group Grupo 3 label's list",

"status": 400

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**5. Patch Issue Name**

**REQUEST**

* **Uri**: PATCH /projects/{name}/issues/{id}/name
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project containing the issue that is being patched
  + id – id of the issue that is being patched
* **Body Parameters**:
  + newName – new name for the issue (required)
* **Body**:

{ "newName": "name" }

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:** Content-type: text/plain

**Body:** Issue 11's name updated successfully

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.IssueInputModel$ProjectIssueNameInputModel] value failed for JSON property newName due to missing (therefore NULL) value for creator parameter newName which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project Grupo 3 not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "User is not a contributor and doesn't have permission to change project Grupo 3.",

"status": 403

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Issue 5 of project Grupo 3 not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**6. Patch Issue Description**

**REQUEST**

* **Uri**: PATCH /projects/{name}/issues/{id}/description
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project containing the issue that is being patched
  + id – id of the issue that is being patched
* **Body Parameters**:
  + description – new description for the issue (required)
* **Body**:

{ "description”: "new description of issue in patch method" }

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:** Content-type: text/plain

**Body:** Issue 11's description updated successfully

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.IssueInputModel$ProjectIssueNameInputModel] value failed for JSON property newName due to missing (therefore NULL) value for creator parameter newName which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project Grupo 3 not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "User is not a contributor and doesn't have permission to change project Grupo 3.",

"status": 403

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Issue 5 of project Grupo 3 not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**6. Patch Issue State**

**REQUEST**

* **Uri**: PATCH /projects/{name}/issues/{id}/state
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project containing the issue that is being patched
  + id – id of the issue that is being patched
* **Body Parameters**:
  + nextState – new state for the issue (required)
* **Body**:

{ "nextState": "closed" }

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:** Content-type: text/plain

**Body:** Issue 11's state updated successfully

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.IssueInputModel$ProjectIssueNameInputModel] value failed for JSON property newName due to missing (therefore NULL) value for creator parameter newName which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project Grupo 3 not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "User is not a contributor and doesn't have permission to change project Grupo 3.",

"status": 403

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Issue 5 of project Grupo 3 not found.",

"status": 404

}

* **Status**: 406 NOT ACCEPTABLE
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "InvalidTransitionException",

"title": "Invalid transition exception.",

"detail": "The issue Grupo 3 - 5 is already archived and cannot be reopened.",

"status": 406

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**7. Delete Issue**

**REQUEST**

* **Uri**: DELETE /projects/{name}/issues/{id}
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project that contains the issue that is being deleted
  + id – id of the issue that is being deleted

**RESPONSE**

SUCCESS

* **Status:** 204 NO-CONTENT

ERROR

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project Grupo 3 not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "User is not a contributor and doesn't have permission to change project Grupo 3.",

"status": 403

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Issue 5 of project Grupo 3 not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**Comments**

**1. Create Comment**

**REQUEST**

* **Uri**: POST /projects/{name}/issues/{id}/comments
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project where the issue is being created
  + id – id of the issue where the comment is being created
* **Body Parameters**:
  + text - user’s comment concerning {id} issue (required)
* **Body**:

{ "text”: "issue comment text" }

**RESPONSE**

SUCCESS

* **Status:** 201 CREATED
* **Headers:** Content-type: text/plain
* **Body:** Comment 17 created successfully for issue 12.

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.CommentInputModel] value failed for JSON property text due to missing (therefore NULL) value for creator parameter text which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project Grupo 3 not found.",

"status": 404

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "User is not a contributor and doesn't have permission to change project Grupo 3.",

"status": 403

}

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "InvalidOperationException",

"title": "Invalid operation exception.",

"detail": "Label 'exploration' is not available in the group Grupo 3 label's list",

"status": 400

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**2. Obtain All Comments from an Issue of a Project**

**REQUEST**

* **Uri:** GET /projects/{name}/issues/{id}/comments
* **Headers:** 
  + Accept: application/vnd.siren+json (optional)
* **Path Parameters:**
  + name - name of the project in which we want to retrieve the list of issues
  + id – id of the issue where the comments are written

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:** 
  + Content-type: application/json, application/vnd.siren+json
* **Body (application/json):**
* {  
   "count": 1,  
   "comments": [  
   **{** "id": 17,  
   "issueID": 12,  
   "text": "issue comment text",  
   "creationDate": "2020-07-16 00:17",  
   "createdBy": "davidwilsonluis",  
   "pname": "Grupo 3"  
   **}** ]  
  }
* **Body (application/vnd.siren+json):**
* {  
   "properties": **{** "count": 1  
  **}**,  
   "links": [  
   **{** "rel": [  
   "self",  
   "/rels/project/issue/comments",  
   "collection"  
   ],  
   "href": "/projects/Grupo%203/issues/12/comments",  
   "title": "Issue 12's Comments"  
   **}**,  
   **{** "rel": [  
   "/rels/project/issue",  
   "parent"  
   ],  
   "href": "/projects/Grupo%203/issues/12",  
   "title": "Issue 12"  
   **}** ],  
   "actions": [  
   **{** "name": "/actions/post/comment",  
   "href": "/projects/Grupo%203/issues/12/comments",  
   "title": "Create comment",  
   "class": [  
   "/classes/comments"  
   ],  
   "method": "POST",  
   "type": "application/json",  
   "fields": [  
   **{** "name": "text",  
   "type": "text"  
   **}** ]  
   **}** ],  
   "entities": [  
   **{** "class": [  
   "/classes/comment"  
   ],  
   "properties": **{** "id": 17,  
   "issueID": 12,  
   "text": "issue comment text",  
   "creationDate": "2020-07-16 00:17",  
   "createdBy": "wilsondavidluis",  
   "pName": "Grupo 3"  
   **}**,  
   "links": [  
   **{** "rel": [  
   "self"  
   ],  
   "href": "/projects/Grupo%203/issues/12/comments/17",  
   "title": "Comment 17"  
   **}** ]  
   **}** ],  
   "class": [  
   "/classes/comments"  
   ]  
  }

ERROR

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project NonExistingGroup not found.",

"status": 404

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Issue 2 of project Grupo 3 not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**3. Obtain Comment Details**

**REQUEST**

* **Uri:** GET /projects/{name}/issues/{id}/comments/{cid}
* **Headers:**
  + Accept: application/vnd.siren+json (optional)
* **Path Parameters:** 
  + name – name of the project where {id} issue is contained
  + id – obtain issue with id = {id}
  + cid – obtain comment with id = {cid}

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:**
  + Content-type: application/json, application/vnd.siren+json
* **Body (application/json):**

{  
 "id": 17,  
 "issueID": 12,  
 "text": "issue comment text",  
 "creationDate": "2020-07-16 00:17",  
 "createdBy": "davidwilsonluis",  
 "pname": "Grupo 3"  
}

* **Body (application/vnd.siren+json):**

{  
 "properties": **{** "createdBy": "davidwilsonluis",  
 "text": "issue comment text",  
 "creationDate": "2020-07-16 00:17"  
 **}**,  
 "links": [  
 **{** "rel": [  
 "self",  
 "/rels/project/issue/comment"  
 ],  
 "href": "/projects/Grupo%203/issues/12/comments/17",  
 "title": "Comment 17"  
 **}**,  
 **{** "rel": [  
 "/rels/project/issue/comments",  
 "parent",  
 "collection"  
 ],  
 "href": "/projects/Grupo%203/issues/12/comments",  
 "title": "Issue 12's Comments"  
 **}** ],  
 "actions": [  
 **{** "name": "/actions/delete/comment",  
 "href": "/projects/Grupo%203/issues/12/comments/17",  
 "title": "Delete comment",  
 "class": [  
 "/classes/comment"  
 ],  
 "method": "DELETE"  
 **}**,  
 **{** "name": "/actions/put/comment",  
 "href": "/projects/Grupo%203/issues/12/comments/17",  
 "title": "Update comment",  
 "class": [  
 "/classes/comment"  
 ],  
 "method": "PUT",  
 "type": "application/json",  
 "fields": [  
 **{** "name": "text",  
 "type": "text"  
 **}** ]  
 **}** ],  
 "entities": [],  
 "class": [  
 "/classes/comment"  
 ]  
}

ERROR

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project NonExistingGroup not found.",

"status": 404

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Issue 2 of project Grupo 3 not found.",

"status": 404

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Comment 2 not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**4. Update Comment on an Issue**

**REQUEST**

* **Uri**: PUT /projects/{name}/issues/{id}/comments/{cid}
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project where the issue is present
  + id – id of the issue containing the comment that is being created
  + cid – id of the comment being created
* **Body Parameters**:
  + text – new comment’s text for the issue (required)
* **Body**:

{ "text": "updated comment text" }

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:** Content-type: text/plain
* **Body:** Comment 17 edited successfully

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.CommentInputModel] value failed for JSON property text due to missing (therefore NULL) value for creator parameter text which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "Comment doesn't belong to this user.",

"status": 403

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project Grupo 3 not found.",

"status": 404

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Issue 5 of project Grupo 3 not found.",

"status": 404

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Comment 2 not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**5. Delete Comment of an Issue**

**REQUEST**

* **Uri**: DELETE /projects/{name}/issues/{id}/comments/{cid}
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project that contains the issue that is being deleted
  + id – id of the issue that contains the comments that’s being deleted
  + cid – id of the comment that is being deleted

**RESPONSE**

SUCCESS

* **Status:** 204 NO-CONTENT

ERROR

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "User is not a contributor and doesn't have permission to change project Grupo 3."

"status": 403

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project Grupo 3 not found.",

"status": 404

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Issue 5 of project Grupo 3 not found.",

"status": 404

}

* **Status**: 404 NOT\_FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Comment 2 not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**Contributors**

**1. Create Contributor**

**REQUEST**

* **Uri**: POST /projects/{name}/contributors
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project where the contributor is being added
* **Body Parameters**:
  + username – name of the user being added has contributor to project (required)
* **Body**:

{ "username" : "aluno2" }

**RESPONSE**

SUCCESS

* **Status:** 201 CREATED
* **Headers:** Content-type: text/plain
* **Body:** Contributor 2 added to project Grupo 3.

ERROR

* **Status**: 400 BAD REQUEST
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "MissingKotlinParameterException",

"title": "Bad parameter",

"detail": "Instantiation of [simple type, class pt.isel.daw.controller.representation.input.ContributorInputModel] value failed for JSON property username due to missing (therefore NULL) value for creator parameter username which is a non-nullable type",

"status": 400

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project InventedProject not found.",

"status": 404

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 403 FORBIDDEN
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "AuthorizationException",

"title": "Authorization exception.",

"detail": "User is not a contributor and doesn't have permission to change project Grupo 3.",

"status": 403

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**2. Obtain All Contributors of Project**

**REQUEST**

* **Uri:** GET /projects/{name}/contributors
* **Headers:** 
  + Accept: application/vnd.siren+json (optional)
* **Path Parameters:**
  + name - name of the project in which we want to retrieve the list of contributors

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:** 
  + Content-type: application/json, application/vnd.siren+json
* **Body (application/json):**

{  
 "count": 2,  
 "contributors": [  
 **{** "userID": 1,  
 "username": "aluno1",  
 "pname": "Grupo 3"  
 **}**,  
 **{** "userID": 2,  
 "username": "aluno2",  
 "pname": "Grupo 3"  
 **}** ]  
}

* **Body (application/vnd.siren+json):**

{  
 "properties": {  
 "count": 2  
},  
 "links": [  
 {  
 "rel": [  
 "self",  
 "/rels/project/contributors",  
 "collection"  
 ],  
 "href": "/projects/Grupo%203/contributors",  
 "title": "All Contributors"  
 },  
 {  
 "rel": [  
 "/rels/project",  
 "parent"  
 ],  
 "href": "/projects/Grupo%203",  
 "title": "Project Grupo 3"  
 }  
 ],  
 "actions": [  
 {  
 "name": "/actions/post/contributor",  
 "href": "/projects/Grupo%203/contributors",  
 "title": "Add contributor to project Grupo 3",  
 "class": [  
 "/classes/contributors"  
 ],  
 "method": "POST",  
 "type": "application/json",  
 "fields": [  
 {  
 "name": "userID",  
 "type": "int"  
 },  
 {  
 "name": "ProjectName",  
 "type": "text"  
 }  
 ]  
 }  
 ],  
 "entities": [  
 {  
 "class": [  
 "/classes/contributor"  
 ],  
 "properties": {  
 "userID": 1,  
 "username": "aluno1",  
 "projectName": "Grupo 3"  
 },  
 "links": [  
 {  
 "rel": [  
 "self"  
 ],  
 "href": "/projects/Grupo%203/contributors/aluno1",  
 "title": "Contributor aluno1"  
 }  
 ]  
 },  
 {  
 "class": [  
 "/classes/contributor"  
 ],  
 "properties": {  
 "userID": 2,  
 "username": "aluno2",  
 "projectName": "Grupo 3"  
 },  
 "links": [  
 {  
 "rel": [  
 "self"  
 ],  
 "href": "/projects/Grupo%203/contributors/aluno2",  
 "title": "Contributor aluno2"  
 }  
 ]  
 }  
 ],  
 "clazz": [  
 "/classes/contributors"  
 ]  
}

ERROR

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project InventedProject not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**3. Obtain Contributor Details of Project**

**REQUEST**

* **Uri:** GET /projects/{name}/contributors/{ctrName}
* **Headers:**
  + Accept: application/vnd.siren+json (optional)
* **Path Parameters:** 
  + name – name of the project where contributor is contained
  + ctrName – name of the contributor whose details will be displayed

**RESPONSE**

SUCCESS

* **Status:** 200 OK
* **Headers:**
  + Content-type: application/json, application/vnd.siren+json
* **Body (application/json):**

{  
 "userID": 1,  
 "username": "aluno1",  
 "pname": "Grupo 3"  
}

* **Body (application/vnd.siren+json):**

{  
 "properties": **{** "userID": 1,  
 "ProjectName": "Grupo 3"  
**}**,  
 "links": [  
 **{** "rel": [  
 "self"  
 ],  
 "href": "/projects/Grupo%203/contributors/aluno1",  
 "title": "Contributor 1"  
 **}**,  
 **{** "rel": [  
 "/rels/project"  
 ],  
 "href": "/projects/Grupo%203",  
 "title": "Project Grupo 3"  
 **}** ],  
 "actions": [  
 **{** "name": "/actions/delete/contributor",  
 "href": "/projects/Grupo%203/contributors/aluno1",  
 "title": "Remove contributor",  
 "class": [  
 "/classes/contributor"  
 ],  
 "method": "DELETE"  
 **}** ],  
 "entities": [],  
 "clazz": [  
 "/classes/comment"  
 ]  
}

ERROR

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project InventedProject not found.",

"status": 404

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User nonexistinguser not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}

**4. Remove Contributor from Project**

**REQUEST**

* **Uri**: DELETE /projects/{name}/contributors/{ctrName}
* **Headers**: Authorization: Basic Auth
* **Path Parameters:**
  + name - name of the project containing the contributor being deleted
  + ctrName – name of the contributor being deleted

**RESPONSE**

SUCCESS

* **Status:** 204 NO-CONTENT

ERROR

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "Project InventedProject not found.",

"status": 404

}

* **Status**: 404 NOT FOUND
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "EntityNotFoundException",

"title": "Entity not found exception.",

"detail": "User not found.",

"status": 404

}

* **Status**: 500 INTERNAL SERVER ERROR
* **Headers**: Content-type: application/problem+json
* **Body**:

{

"type": "DatabaseAccessException",

"title": "Database access exception.",

"detail": "Error accessing database: …",

"status": 500

}