

CLASS 4

Sprint 3

Technical Requirements

- Sprint 2 PBIs updated descriptions, and test scenarios
- Sprint 3 PBIs descriptions, updated and new attributes, test data, and test cases

Product Backlog

1	VIEW-TASK-TABLE VIEW	1
2	VIEW-TASK-DETAIL	
3	ADD-TASK-BASIC	2
4	DELETE-TASK	
5	EDIT-TASK-BASIC	
6	ADD-STATUS	3
7	EDIT-STATUS	
8	DELETE-STATUS	
9	NESTED-PUBLIC-PATH	
10	SORT-TASK-BY-STATUS-TABLE-VIEW	4
11	FILTER-TASK-BY-STATUS-TABLE-VIEW	
12	LIMIT-TASKS-IN-STATUS	
13	VALIDATE-INPUT-FE-BASIC	
14	VALIDATE-INPUT-BE-BASIC	

- Release-Level Priority
- The priority is modified to accommodate for the level of difficulties of each PBI

● Must-have

● Should-have

● Could-have

- in-complete tasks from previous Sprint may be carried forward to current Sprint backlog
- (at the beginning of the Sprint) commit PBI (Sprint goal) to advisor
- share what the team have learned on this Sprint Spike
- development of committed PBI in Sprint backlog
[each PBI should be broken down to tasks such as create DB schema, create entity class, create controller, test endpoint with postman, deploy on server, ...]
- review and UAT **completed** PBI with advisor
- next Sprint Spike

PBI #	PBI NAME	PBI DESCRIPTION / CONDITIONS
3	ADD-TASK-BASIC	<ol style="list-style-type: none">1. When the user click on “Add” button, the task detail modal window is shown with options to “Save” and “Cancel”. "Save" is enabled only if the taskTitle is NOT empty.<ol style="list-style-type: none">1.1 The user can add the Task’s details: Title [REQUIRED], Description [OPTIONAL], Assignees [OPTIONAL], Status [default: No Status]1.2 FE does NOT trim any field while the user enters the data on these fields. FE must trim task details before sending to BE.1.3 On “Save” FE sends request to BE, On “Cancel” returns to previous page2. BE assumes that the request body contains valid fields (validation will be implemented in PBI14.)3. DB: inserts data and auto inserts createdOn and updatedOn (current date, time), dateTime data are NOT provided by BE4. If DB insert is successful, BE returns status code 201 and FE shows “The task has been successfully added”5. FE Shows updated TableView with added task

Sprint 2 PBI Descriptions

PBI #	PBI NAME	PBI DESCRIPTION / CONDITIONS
4	DELETE-TASK	<ol style="list-style-type: none">1. When the user click on “Delete” for a task, a confirmation modal window is shown showing showing task running number (in the table) and taskTitle to be deleted. “Confirm” and “Cancel” options are provided.2. When “Confirm”, FE sends request to BE to delete the specified task.3. On DB deletion success, BE returns status code 200, FE shows “The task has been deleted”<ol style="list-style-type: none">3.1 On failed deletion of invalid taskID, BE returns status code 404, FE shows “An error has occurred, the task does not exist.”4. FE shows updated task table

Sprint 2 PBI Descriptions

PBI #	PBI NAME	PBI DESCRIPTION / CONDITIONS
5	EDIT-TASK-BASIC	<ol style="list-style-type: none">1. When the user clicks on “Edit” button for a task, a modal window with Task-Detail is shown and all fields, except DateTime fields are editable, with options to “Save” or “Cancel”. "Save" is enabled only if the form is modified by the user.2. FE does NOT trim any field while the user enters the data on these fields. FE must trim task details before sending to BE.3. On “Save”, FE sends request to BE. On “Cancel” no request is made to BE, FE then returns to previous page4. On successful update, FE shows “The task has been updated” and returns to the previous page5. On failed update (invalid taskID), BE returns 404 status code and FE shows “An error has occurred, the task does not exist.”

Sprint 2 Test Scenarios

PBI	TEST ID	Test Scenario	Expected Results
3	TC-PBI3-ADD-TASK-BASIC-1	normal – no taskTitle and cancel	1. “Save” button is disable 2. FE does not send request to BE and returns to previous page
	TC-PBI3-ADD-TASK-BASIC-2	normal – no OPTIONAL fields and status is not specified	1. FE does not send createdOn and updatedOn to BE 2. New task is created successfully. 3. FE shows OPTIONAL fields correctly. The status is “No Status”, createdOn and updatedOn are the same.
	TC-PBI3-ADD-TASK-BASIC-3	normal – max size in every field; “To Do”	1. New task is created successfully with every character stored in the database and the status is “To Do”.
	TC-PBI3-ADD-TASK-BASIC-4	normal – with leading and trailing whitespace; “Done”	1. New task is created successfully. 2. FE has trimmed leading and trailing whitespace 3. BE also trim leading and trailing whitespace (if exist)

Sprint 2 Test Scenarios

PBI	TEST ID	Test Scenario	Expected Results
4	TC-PBI4-DELETE-TASK-1	normal – cancel	<ol style="list-style-type: none">1. FE shows confirmation message with task running number and title2. When click “Cancel”, FE does not send request to BE and returns to previous page. The task is shown in the task table.
	TC-PBI4-DELETE-TASK-2	normal – successful delete	<ol style="list-style-type: none">1. When click “Confirm”, FE send request to BE2. BE successfully delete the task and returns 2003. FE shows “The task has been deleted” and the deleted task is not shown in the task table
	TC-PBI4-DELETE-TASK-3	failed – task not exist (deleted by someone else)	<ol style="list-style-type: none">1. FE shows confirmation message with task running number and title2. BE returns 4043. FE shows “The task does not exist” and the task is not shown in the task table

Sprint 2 Test Scenarios

PBI	TEST ID	Test Scenario	Expected Results
4	TC-PBI5-EDIT-TASK-BASIC-1	normal – no modification and cancel	<ol style="list-style-type: none">1. All fields are editable except createdOn and updatedOn2. “Save” button is disable at the beginning and when taskTitle is empty3. FE does not send request to BE and returns to previous page with task details unchanged
	TC-PBI5-EDIT-TASK-BASIC-2	normal – change all editable fields	<ol style="list-style-type: none">1. FE shows “The task has been updated”2. The task is updated in DB and task table view correctly, createdOn is unchanged, and updatedOn has been changed to the current time
	TC-PBI5-EDIT-TASK-BASIC-3	normal – change OPTIONAL fields to null	<ol style="list-style-type: none">1. FE shows “The task has been updated”2. The task is updated in DB and task table view correctly, createdOn is unchanged, and updatedOn has been changed to the current time
	TC-PBI5-EDIT-TASK-BASIC-4	failed – no task	<ol style="list-style-type: none">1. BE returns 4042. FE shows “The task does not exist” and on close of message window, returns to previous page

Attribute Name	Description / Constraints	Example
taskTitle	Task name : UTF-8 TEXT; Limit to 100 characters; NOT NULL; NOT EMPTY; Leading and trailing whitespaces are trimmed;	design API specs for PBI1–2
taskDescription	Task description : UTF-8 TEXT; Limit to 500 characters; OPTIONAL; NOT EMPTY; Leading and trailing whitespaces are trimmed;	need to specify endpoints, request body, response body, status
taskAssignees	Task assignees : UTF-8 TEXT; Limit to 30 characters; OPTIONAL; NOT EMPTY; Leading and trailing whitespaces are trimmed.	Pang; Pong;
taskStatus	Task status : applicable statuses are stored in status table; NOT NULL; The default value is 'No Status';	To Review
createdOn	Task creation date/time (on DBMS) : DATETIME; NOT NULL;	2024-04-19 01:00:00
updatedOn	Task last update date/time (on DBMS) : DATETIME; When a task is created, updatedOn is the same as createdOn; NOT NULL;	2024-04-20 18:00:00

status Attributes : v1 [PBI6–]

Attribute Name	Description / Constraints	Example
statusName	Status name : UTF-8 TEXT; Limit to 50 characters; NOT NULL; NOT EMPTY; UNIQUE Leading and trailing whitespaces are trimmed;	To Do
statusDescription	Status description : UTF-8 TEXT; Limit to 200 characters; OPTIONAL; NOT EMPTY; Leading and trailing whitespaces are trimmed;	This item hasn't been started

PBI #	PBI NAME	PBI DESCRIPTION / CONDITIONS
6	ADD-STATUS	<ol style="list-style-type: none">1. When the user clicks on “Manage Status” menu, the Status table listing existing statuses is shown with statusName and statusDescription.2. User has choice to Add, Edit, or Delete Status. The “No Status” status cannot be edited or deleted.3. When “Add” is clicked, a modal window to add a new status is shown, with options to “Save” or “Cancel”. "Save" is enabled only if the statusName is NOT empty.4. User enter status Name [Required] and Description [Optional] . The statusName must be unique (uniqueness is NOT tested in this PBI)<ol style="list-style-type: none">4.1 FE does NOT trim any field while the user enters the data on these fields. FE must trim status details before sending to BE.5. When user click "Save", FE sends request to BE<ol style="list-style-type: none">5.1 On success, BE returns 200, FE shows a message “The status has been added” and then returns the previous page to show the updated Status Table5.2 On failure, BE returns 400 or 500, FE shows a message “An error has occurred, the status could not be added.”

PBI #	PBI NAME	PBI DESCRIPTION / CONDITIONS
7	EDIT-STATUS	<ol style="list-style-type: none">1. When the user clicks on “Edit” button for a Status, the Status-Detail modal window is shown and all fields are editable, with options to “Save” or “Cancel”.<ol style="list-style-type: none">1.1 the “No Status” status cannot be edited1.2 “Save” is enabled only if the form is modified by the user and not empty1.3 FE does NOT trim any field while the user enters the data on these fields. FE must trim status details before sending to BE.2. On “Save”, FE sends request to BE. On “Cancel”, no request is made to BE, FE then returns to previous page.3. On successful update, FE shows “The status has been updated” and returns to the previous page. The status table and task table are shown with the updated name or description.4. On failed update (invalid statusID), BE returns 404 status code and FE shows “An error has occurred, the status does not exist.”5. User can click “Close” or “Back” to return to previous page.

PBI #	PBI NAME	PBI DESCRIPTION / CONDITIONS
8	DELETE-STATUS	<ol style="list-style-type: none">1. The “No Status” status cannot be deleted2. If there is no task under the specified status, FE opens modal window asking the user “Do you want to delete <statusName> status?” “Confirm” and “Cancel” options are provided.<ol style="list-style-type: none">2.1 When “Confirm”, FE sends request to BE to delete the status. If the deletion is successful, BE returns 200 and FE shows “The status has been deleted”. In the case the status does not exist, BE returns 404 and FE shows “An error has occurred, the status does not exist.”2.2 When “Cancel”, FE does not send request to BE and returns to previous page

PBI #	PBI NAME	PBI DESCRIPTION / CONDITIONS
8	DELETE-STATUS	<p>3. If there are tasks in the specified status, FE opens modal window asking user “There are tasks in <statusName> status. In order to delete this status, the system must transfer tasks in this status to existing status. Transfer tasks to [drop-down of existing statuses] ” “Transfer and Delete” and “Cancel” options are provided.</p> <p>3.1 “Transfer and Delete” is enable only when destination status is selected (there is no default)</p> <p>3.2 When “Transfer and Delete”, FE sends request to BE to transfer tasks in <i>to-delete</i> status to destination status and delete the <i>to-delete</i> status. If the transfer and deletion is successful, BE returns 200 and FE shows “The task(s) have been transferred and the status has been deleted”. In the case the status is no longer exist, BE returns 404 and FE shows “An error has occurred, the status does not exist.”</p> <p>3.3 When “Cancel”, FE does not send request to BE and returns to previous page</p> <p>4. FE shows updated Status table</p>

PBI #	PBI NAME	PBI DESCRIPTION / CONDITIONS
9	NESTED-PUBLIC-PATH	<ol style="list-style-type: none">1. The system must be accessible from internet (without VPN).2. This includes frontend and backend, but not database.3. The database must be accessible within the server only.4. The public url for each system is ‘http://intproj23.sit.kmutt.ac.th/<TEAMCODE>/’, e.g., ‘http://intproj23.sit.kmutt.ac.th/sy1/’5. ‘intproj23.sit.kmutt.ac.th’ is maintained by the school admin.6. The team may configure reverse proxy on the host machine to handle CORS issue better.

API Specifications for Sprint 3

Server:

`http://intproj23.sit.kmutt.ac.th/<TEAMCODE>`

URI:

POST

/v2/statuses Create a new status

PUT

/v2/statuses/{id} Update a status with id

GET

/v2/statuses Gets all status

DELETE

/v2/statuses/{id} Delete a status with id

DELETE

/v2/statuses/{id}/{newId} Delete and transfer

POST /v2/statuses

POST /v2/statuses Create a new status

Status name must unique

Parameters

No parameters

Request body required

Example Value | Schema

```
{
  "name": "string",
  "description": "string"
}
```

Responses

Code	Description
------	-------------

201	Successful Operation
-----	----------------------

Media type

application/hal+json

Controls Accept header.

Example Value | Schema

```
{
  "id": 0,
  "name": "string",
  "description": "string"
}
```

400	Failed Operation for invalid input
-----	------------------------------------

Media type

application/json

Example Value | Schema

```
{
  "timestamp": "2024-04-23T08:29:03.133+00:00",
  "status": 404,
  "message": "NOT FOUND",
  "instance": "/v1/tasks/99"
}
```

500	Failed Operation
-----	------------------

Curl

```
curl -X 'POST' \  
  'http://localhost:8080/v2/statuses' \  
  -H 'accept: application/hal+json' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "name": "Waiting Some Task Finished",  
    "description": "Test status description"  
  }'
```

Request URL

```
http://localhost:8080/v2/statuses
```

Server response

Code	Details
------	---------

201	
-----	--

Response body

```
{  
  "id": 9,  
  "name": "Waiting Some Task Finished",  
  "description": "Test status description"  
}
```

Curl

```
curl -X 'POST' \  
  'http://localhost:8080/v2/statuses' \  
  -H 'accept: application/hal+json' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "name": "POC",  
    "description": "Test status description"  
  }'
```

Request URL

http://localhost:8080/v2/statuses

Server response

Code

Details

500

Error: response status is 500

Response body

```
{  
  "timestamp": "2024-05-06T20:14:00.570+00:00",  
  "status": 500,  
  "error": "Internal Server Error",  
  "message": "could not execute statement [Duplicate entry 'POC' for key 'statuses.name_UNIQUE']  
(description,name) values (?,?); constraint [statuses.name_UNIQUE]",  
  "path": "/v2/statuses"  
}
```

PUT /v2/statuses{id}

PUT **/v2/statuses/{id}** Update a status with id

Status id to be update must exist in db

Parameters

Name	Description
id <small>* required</small>	
<code>integer(\$int32)</code>	id of status to be update
<code>(path)</code>	
	<input type="text" value="id"/>

Request body required

Example Value | Schema

```
{  "name": "string",  "description": "string"}
```

Responses	
Code	Description
200	<div>Successful Operation</div> <div>Media type</div> <div>application/hal+json</div> <div>Controls Accept header.</div> <div>Example Value Schema</div> <div><pre>{ "id": 0, "name": "string", "description": "string"}</pre></div>
404	<div>Failed Operation (Resource Not Found)</div> <div>Media type</div> <div>application/json</div> <div>Example Value Schema</div> <div><pre>{ "timestamp": "2024-04-23T08:29:03.133+00:00", "status": 404, "message": "NOT FOUND", "instance": "/v1/tasks/99"}</pre></div>

Responses Operation Success Example

Curl

```
curl -X 'PUT' \
  'http://localhost:8080/v2/statuses/7' \
  -H 'accept: application/hal+json' \
  -H 'Content-Type: application/json' \
  -d '{
    "name": "Waiting",
    "description": "Waiting somethings"
  }'
```

Request URL

http://localhost:8080/v2/statuses/7

Server response

Code	Details
------	---------

200

Response body

```
{
  "id": 7,
  "name": "Waiting",
  "description": "Waiting somethings"
}
```

Operation Failed Example

Responses

Curl

```
curl -X 'PUT' \
  'http://localhost:8080/v2/statuses/7' \
  -H 'accept: application/hal+json' \
  -H 'Content-Type: application/json' \
  -d '{
    "name": "Doing",
    "description": "Task still in processing"
  }'
```

Request URL

http://localhost:8080/v2/statuses/7

Server response

Code	Details
------	---------

500

Undocumented Error: response status is 500

Response body

```
{
  "timestamp": "2024-05-06T20:25:53.113+00:00",
  "status": 500,
  "error": "Internal Server Error",
  "message": "could not execute statement [Duplicate entry 'Doing' for key 'statuses.name_UNIQUE'; constraint [statuses.name_UNIQUE]",
  "path": "/v2/statuses/7"
}
```

DELETE /v2/statuses/{id}

DELETE /v2/statuses/{id} Delete a status with id

Delete status by id

Parameters

Name	Description
id * required	
integer(\$int32)	id of task to be delete
(path)	

id

Responses

Code

Description

200

Successful Operation

Media type

application/hal+json

Controls Accept header.

Example Value | Schema

```
{}
```

404

Failed Operation (Resource Not Found)

Media type

application/json

Example Value | Schema

```
{
  "timestamp": "2024-04-23T08:29:03.133+00:00",
  "status": 404,
  "message": "NOT FOUND",
  "instance": "/v1/tasks/99"
}
```


DELETE /v2/statuses{id}

DELETE /v2/statuses/{id}/{newId} Delete and transfer

Delete a specific status id and transfer task with this status to another status

Parameters

Name	Description
id * required integer(\$int32) (path)	id of status to be delete
newId * required integer(\$int32) (path)	id of transfer to status

Code	Description
200	<p>Successful Operation</p> <p>Media type</p> <p>application/hal+json ▾</p> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{}</pre>
404	<p>Failed Operation (Resource Not Found)</p> <p>Media type</p> <p>application/json ▾</p> <p>Example Value Schema</p> <pre>{ "timestamp": "2024-04-23T08:29:03.133+00:00", "status": 404, "message": "NOT FOUND", "instance": "/v1/tasks/99" }</pre>

IT-Bangmod Kradan Kanban

itbkk-manage-status

Manage Status

+	Title	Assignees	Status
1	TaskTitle1TaskTitle2TaskTitle3TaskTitle4TaskTitle5TaskTitle6TaskTitle7TaskTitle8TaskTitle9TaskTitle0	Assignees1Assignees2Assignees3	No Status
2	Repository	Unassigned	To Do
3	ดาต้าเบส	あなた、彼、彼女 (私ではありません)	Doing
4	_Infrastructure_	ไถ่จวง กับ เพนกวิ้น	Done
5	Frontend UX/UI Design	vuejs	To Do

IT-Bangmod Kradan Kanban

itbkk-button-home

itbkk-button-add

Home > Task Status

Add Status

	Name	Description	itbkk-item		Action
1	No Status	The default status			
2	To Do				Edit Delete
3	Doing				Edit Delete
4	Done				Edit Delete

itbkk-status-name

itbkk-status-description

itbkk-button-edit

itbkk-button-delete

SAMPLE UI : Add Status

Add Status

itbkk-modal-status

Name

itbkk-status-name

Description

itbkk-status-description

Save

Cancel

Edit Status

itbkk-modal-status

Name

itbkk-status-name

Doing

Description

itbkk-status-description

TimeZone: Asia/Bangkok

Created On: 22/04/2024, 16:00:00

Updated On: 22/04/2024, 16:00:00

itbkk-timezone

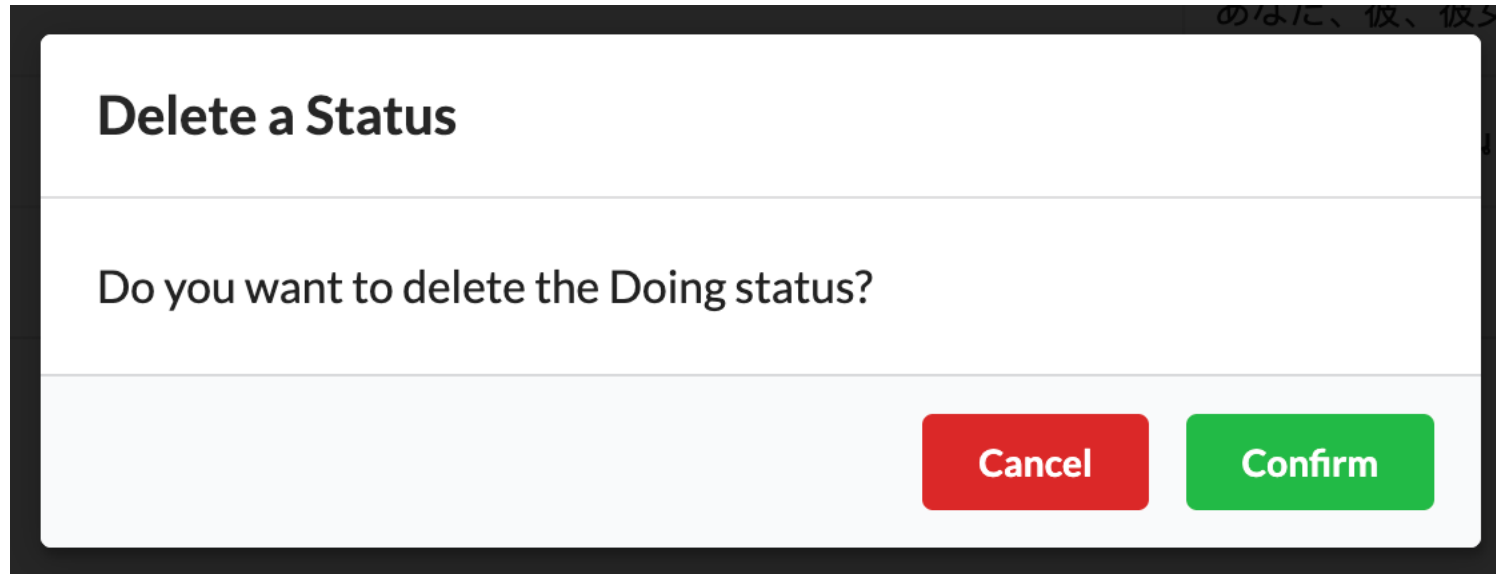
itbkk-created-on

itbkk-updated-on

Save

Cancel

itbkk-message

A dialog box titled "Delete a Status" with a light gray background and a dark gray border. It contains a message "Do you want to delete the Doing status?" and two buttons: "Cancel" (red) and "Confirm" (green).

Delete a Status

Do you want to delete the Doing status?

Cancel Confirm

itbkk-button-cancel

itbkk-button-confirm

itbkk-message

Transfer a Status

There is some task associated with the Doing status.

Transfer to

CancelTransfer

itbkk-button-cancel

itbkk-button-confirm

Transfer to

No Status

To Do

Doing

Done

PBI	TEST ID	Test Scenario	Expected Results
6	TC-PBI6-ADD-STATUS-1	normal – cancel	<ol style="list-style-type: none">1. List of statuses is shown in status table2. “Save” button is disable initially and enable when user add status name3. FE does not send request to BE and returns to previous page
	TC-PBI6-ADD-STATUS-2	normal – no OPTIONAL fields	<ol style="list-style-type: none">1. New status is created successfully2. BE returns 2013. FE shows OPTIONAL fields correctly
	TC-PBI6-ADD-STATUS-3	normal – max size in every field	<ol style="list-style-type: none">1. New status is created successfully with every character stored in the database
	TC-PBI6-ADD-STATUS-4	normal – with leading and trailing whitespace	<ol style="list-style-type: none">1. New status is created successfully.2. FE has trimmed leading and trailing whitespace3. BE also trim leading and trailing whitespace (if exist)

Sprint 3 Test Scenarios

PBI	TEST ID	Test Scenario	Expected Results
7	TC-PBI7-EDIT-STATUS-1	normal – edit “No Status”, no modification and cancel	<ol style="list-style-type: none">1. FE shows “This status is the default status and cannot be modified”2. “Save” button is disable at the beginning and when taskName is empty. The button is enabled with statusName is edited.3. On “Cancel”, FE does not send request to BE and returns to previous page with status details unchanged
	TC-PBI7-EDIT-STATUS-2	normal – change status name only	<ol style="list-style-type: none">1. FE shows “The status has been updated”2. The status is updated in DB and status table view correctly3. The task with updated status name is shown with new status name
	TC-PBI7-EDIT-STATUS-3	normal – change all fields	<ol style="list-style-type: none">1. FE shows “The status has been updated”2. The status is updated in DB and status table view correctly
	TC-PBI7-EDIT-STATUS-4	failed – no status	<ol style="list-style-type: none">1. BE returns 4042. FE shows “An error has occurred, the status does not exist” and on close of message window, returns to previous page