Approvals

Entity - Approvals - Handles instruction approval

Page version: 21, last updated: 8:45PM, 25-Sep-20

Verb	Request	Notes	Permissions required
POST	/Consumer /Approvals	This endpoint can be used to post approval message to the API which will either approve or reject given instruction Post body: • InstructionId (int) - id of the instruction • Comment (string) - any comment to go with the approval or rejection • Approved (bool flag) - true if the instruction is to be approved or false if it is to be rejected. ⊘ This is obsolete since v3.1 to support scheduled instructions. Use /Consumer/Approvals /Instruction instead.	Approver permission on the instruction user is attempting to approve
GET	/Consumer /Approvals /CanApprove/ {instructionId}	Can be used to check if calling user can approve given instruction Returns object with following fields: • CanApprove (bool flag) - true if yes, false if no • ErrorMessage (string) - actual error message • ErrorType - (string) for false flag denotes the type of issue encountered. Values supported so far are: • "workflow" - meaning instructions isn't in AvaitingApproval state • "ownership" - meaning the calling user is asking about his/her own action (and the override approver permission is off) • "permissions" - meaning the user failed the IsAuthenticated call either because they do not have the permission to approve or do not exist in the system at all. ⊘ This is obsolete since v3.1 to support scheduled instructions. Use /Consumer/Approvals /CanApprove/instruction/{id} instead.	None. Internally the method calls the same code as the authorization code but instead of returning Unauthorized code to the browser returns CanApprove = false and ErrorType = "permissions".
GET	/Consumer /Approvals /GetNotifications	Can be used to get the list of Instructions that the calling user can approve ⊘ This is obsolete since v3.1 to support scheduled instructions. Use /Consumer/Approvals /notifications/instructions instead.	Instructions are filtered based on user's permissions.
POST	/Consumer /Approvals /Instruction	To approve or reject an instruction approval request. Post body: • InstructionId (int) - id of the instruction • Comment (string) - any comment to go with the approval or rejection • Approved (bool flag) - true if the instruction is to be approved or false if it is to be rejected. ⓘ Since v3.1	Approver permission on the instruction user is attempting to approve

POST	/Consumer /Approvals /ScheduledInstruction	To approve or reject an scheduled instruction approval request. Post body: • ScheduledInstructionId(int) - id of the instruction • Comment (string) - any comment to go with the approval or rejection • Approved (bool flag) - true if the instruction is to be approved or false if it is to be rejected. ⓘ Since v3.1	Approver permission on the scheduled instruction user is attempting to approve
GET	/Consumer /Approvals /notifications /instructions	Can be used to get the list of Instructions that the calling user can approve ⓘ Since v3.1	Instructions are filters based on user's permissions.
GET	/Consumer /Approvals /notifications /scheduledinstructio ns	Can be used to get the list of scheduled Instructions that the calling user can approve ⓘ Since v3.1	Scheduled instructions are filters based on user's permissions.

GET	/Consumer /Approvals /CanApprove /instruction/{id}	Can be used to check if calling user can approve given instruction
		
		Returns object with following fields:
		
		• CanApprove (bool flag) - true if yes, false if no
		• ErrorMessage (string) - actual error message
		• ErrorType - (string) for false flag denotes the type of issue encountered. Values supported so far are:
		  • "workflow" - meaning instructions isn't in AvaitingApproval state
		  • "ownership" - meaning the calling user is asking about his/her own action (and the override approver permission is off)
		  • "permissions" - meaning the user failed the IsAuthenticated call either because they do not have the permission to approve or do not exist in the system at all.
		
		ⓘ Since v3.1
			None. Internally the method calls the same code as the authorization code but instead of returning Unauthorized code to the browser returns CanApprove = false and ErrorType = "permissions".

GET	/Consumer /Approvals /canapprove /scheduledinstructio n/{id}	Can be used to check if calling user can approve given instruction
		
		Returns object with following fields:
		
		• CanApprove (bool flag) - true if yes, false if no
		• ErrorMessage (string) - actual error message
		• ErrorType - (string) for false flag denotes the type of issue encountered. Values supported so far are:
		  • "workflow" - meaning instructions isn't in AvaitingApproval state
		  • "ownership" - meaning the calling user is asking about his/her own action (and the override approver permission is off)
		  • "permissions" - meaning the user failed the IsAuthenticated call either because they do not have the permission to approve or do not exist in the system at all.
		
		ⓘ Since v3.1
			None. Internally the method calls the same code as the authorization code but instead of returning Unauthorized code to the browser returns CanApprove = false and ErrorType = "permissions".

GET	/Consumer /Approvals /notifications	Can be used to get all approval requests that the calling user can approve. These include	All the items are filtered based on user permissions. For example, if a user is not allowed to approve instructions then the call will return empty instructions array.

- Instructions
- Scheduled instructions
- Device authorization for two factor authentication

Sample Response Payload

```
{
    "Instructions": [
        {
            "Id": 2,
            "Sequence": 0,
            "Name": "1E-Explorer-
1ECore-MyInstruction1",
            // properties removed for
brevity
        },
        {
            "Id": 3,
            "Sequence": 0,
            "Name": "1E-Explorer-
1ECore-MyInstruction2",
            // properties removed for
brevity
        }
    ],
    "ScheduledInstructions": [
        {
            "Id": 1,
            "Name": "1E-Explorer-
Examples-MyInstruction1",
            "Cmd": "SendAll",
            "KeepRaw": true,
            "Scope": null,
            "InstructionTtlMinutes":
60,
            "ResponseTtlMinutes": 120,
            "CreatedTimestampUtc":
"2017-11-21T16:37:42.997Z",
            "Export": false,
            "ExportLocation":
null,
            // properties removed for
brevity
        },
        {
            "Id": 1,
            "Name": "1E-Explorer-
Examples-MyInstruction2",
            "Cmd": "SendAll",
            "KeepRaw": true,
            "Scope": null,
            "InstructionTtlMinutes":
60,
            "ResponseTtlMinutes": 120,
            "CreatedTimestampUtc":
"2017-11-21T16:37:42.997Z",
            "Export": false,
            "ExportLocation":
null,
            // properties removed for
brevity
        }
    ],
    "Devices": [
        {
            "Id": 1,
            "UserName": "acme\\jane.
doe",
            "UserEmail": "jane.
doe@acme.com",
```

```
                "RequestGuid": "085fb65f-
f653-4336-a767-17fc826cd453 ",
                "DeviceType": null,
                "CreatedTimestampUtc":
"2017-11-21T16:39:15.723Z",
                "ModifiedTimestampUtc":
"2017-11-21T16:39:15.723Z",
                "Manufacturer": null,
                "Model": null,
                "OsVersion": null
        }
    ]
}
```

ⓘ Since v3.2