PLANNING POKER API

VERSION 1.0.0 – 10/12/2023: API DOCUMENTATION

Table of Contents

Chapter 1. Overview

The documentation of the API

1.1. Version information

Version : 1.0

1.2. URI scheme

Host : localhost:8080
BasePath : /
Schemes : HTTP

1.3. Tags

- Backlog : All about backlogs
- Discussion : All about discussions
- Estimation : All about estimations
- Session : All about sessions
- Team : All about teams
- User : All about users

Chapter 2. Resources

2.1. Backlog

All about backlogs

2.1.1. Returns a backlog

```
GET /api/backlog
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the backlog	string

Responses

HTTP Code	Description	Schema
200	Backlog found	Backlog
400	Invalid session ID	No Content
401	User not authenticated	No Content
404	Backlog not found	No Content

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/backlog?sessionId=string
```

Example HTTP response

Response 200

```
{
  "items" : [ {
    "description" : "As an authorized User I want to see the list of items so that I can select one",
    "estimation" : "3",
    "number" : "US001",
    "priority" : "5",
    "title" : "List Items"
  } ],
  "name" : "MyBacklog"
}
```

2.1.2. Delete all backlog item

```
DELETE /api/backlog/clear
```

Parameters

Type	Name	Schema
Query	**sessionId** *required*	string

Responses

HTTP Code	Description	Schema
200	Backlog items removed	No Content
401	User not authenticated	No Content
404	Backlog not found	No Content

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/backlog/clear?sessionId=string
```

2.1.3. Export a backlog as CSV file

```
GET /api/backlog/export
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the backlog	string

Responses

HTTP Code	Description	Schema
200	Successfully exported the backlog	string (binary)
400	Invalid session ID	No Content
401	User not authenticated	No Content
404	Backlog not found	No Content

Produces

- `text/csv`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/backlog/export?sessionId=string
```

Example HTTP response

Response 200

```
{
  "text/csv" : "Number;Title;Description;Estimation;Priority\nUS007;Create Account;As an unauthorized User I want to create a new account;3;1\n"
}
```

2.1.4. Import a backlog from CSV file

```
POST /api/backlog/import
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the backlog	string
FormData	**file** *required*	The CSV file with the backlog	string (binary)

Responses

HTTP Code	Description	Schema
200	Successfully imported the backlog	Backlog
400	Invalid session ID	No Content
401	User not authenticated	No Content
404	Backlog not found	No Content
405	Invalid input	No Content

Consumes

- `multipart/form-data`

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/backlog/import?sessionId=string
```

Request formData

```
"string"
```

Example HTTP response

Response 200

```
{
  "items" : [ {
    "description" : "As an authorized User I want to see the list of items so that I can select one",
    "estimation" : "3",
    "number" : "US001",
    "priority" : "5",
    "title" : "List Items"
  } ],
  "name" : "MyBacklog"
}
```

2.1.5. Add a backlog item

```
POST /api/backlog/item
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the backlog	string
Body	**body** *required*		BacklogItemAdd

Responses

HTTP Code	Description	Schema
201	Backlog item added	BacklogItem
400	Invalid session ID	No Content

HTTP Code	Description	Schema
401	User not authenticated	No Content
404	Backlog not found	No Content
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/backlog/item?sessionId=string
```

Request body

```
{
  "description" : "As an authorized User I want to see the list of items so that I can select one",
  "title" : "List Items"
}
```

Example HTTP response

Response 201

```
{
  "description" : "As an authorized User I want to see the list of items so that I can select one",
  "estimation" : "3",
  "number" : "US001",
  "priority" : "5",
  "title" : "List Items"
}
```

2.1.6. Returns a backlog item by number

```
GET /api/backlog/item/{number}
```

Parameters

Type	Name	Description	Schema
Path	**number** *required*	Number of the backlog item	string
Query	**sessionId** *required*		string

Responses

HTTP Code	Description	Schema
200	Backlog item removed	No Content
400	Invalid number	No Content
401	User not authenticated	No Content
404	Backlog item not found	No Content

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/backlog/item/string?sessionId=string
```

2.1.7. Update a backlog item by number

```
PUT /api/backlog/item/{number}
```

Parameters

Type	Name	Description	Schema
Path	**number** *required*	Number of the backlog item	string
Query	**sessionId** *required*		string
Body	**body** *required*		BacklogItemUpdate

Responses

HTTP Code	Description	Schema
200	Backlog item updated	BacklogItem
400	Invalid number	No Content
401	User not authenticated	No Content
404	Backlog item not found	No Content

Consumes

- `application/json`

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/backlog/item/string?sessionId=string
```

Request body

```
{
  "description" : "As an authorized User I want to see the list of items so that I can select one",
  "estimation" : "3"
  "priority" : "5",
  "title" : "List Items"
}
```

Example HTTP response

Response 200

```
{
  "description" : "As an authorized User I want to see the list of items so that I can select one",
  "estimation" : "3",
  "number" : "US001",
  "priority" : "5",
  "title" : "List Items"
}
```

2.1.8. Delete a backlog item by number

```
DELETE /api/backlog/item/{number}
```

Parameters

Type	Name	Description	Schema
Path	**number** *required*	Number of the backlog item	string
Query	**sessionId** *required*		string

Responses

HTTP Code	Description	Schema
200	Backlog item removed	No Content
400	Invalid number	No Content
401	User not authenticated	No Content
404	Backlog item not found	No Content

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/backlog/item/string?sessionId=string
```

2.1.9. Move a backlog item

```
POST /api/backlog/item/{number}/move
```

Parameters

Type	Name	Description	Schema
Path	**number** *required*	Number of the backlog item	string
Query	**sessionId** *required*		string
Body	**body** *required*		BacklogItemMove

Responses

HTTP Code	Description	Schema
200	Backlog item moved	< BacklogItem > array
400	Invalid number	No Content
401	User not authenticated	No Content
404	Backlog item not found	No Content

Consumes

- `application/json`

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/backlog/item/string/move?sessionId=string
```

Request body

```
{
  "newIndex" : 1
}
```

Example HTTP response

Response 200

```
[ {
  "description" : "As an authorized User I want to see the list of items so that I can select one",
  "estimation" : "3",
  "number" : "US001",
  "priority" : "5",
  "title" : "List Items"
} ]
```

2.1.10. Rename a backlog

```
PATCH /api/backlog/rename
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the backlog	string
Body	**body** *required*		BacklogRename

Responses

HTTP Code	Description	Schema
200	Backlog renamed	Backlog

HTTP Code	Description	Schema
400	Invalid session ID	No Content
401	User not authenticated	No Content
404	Backlog not found	No Content

Consumes

- `application/json`

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/backlog/rename?sessionId=string
```

Request body

```
{
  "name" : "MyBacklog"
}
```

Example HTTP response

Response 200

```
{
  "items" : [ {
    "description" : "As an authorized User I want to see the list of items so that I can select one",
    "estimation" : "3",
    "number" : "US001",
    "priority" : "5",
    "title" : "List Items"
  } ],
  "name" : "MyBacklog"
}
```

2.2. Discussion

All about discussions

2.2.1. Returns the discussion

```
GET /api/discussion
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the discussion	string

Responses

HTTP Code	Description	Schema
200	Discussion found	Discussion
400	Invalid session ID	No Content
401	User not authenticated	No Content

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/discussion?sessionId=string
```

Example HTTP response

Response 200

```
{
  "active" : true,
  "duration" : "00:03:00",
  "endedAt" : "2023-05-16 10:00:00",
  "posts" : [ {
    "author" : "Max",
    "content" : "Let's estimate together.",
    "createdAt" : "2023-05-16 10:00:00",
    "updatedAt" : "2023-05-16 10:00:00"
  } ],
  "startedAt" : "2023-05-16 10:00:00",
  "topic" : "General"
}
```

2.2.2. End a discussion

```
POST /api/discussion/end
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the discussion	string

Responses

HTTP Code	Description	Schema
200	Discussion ended	Discussion
400	Invalid session ID	No Content
401	User not authenticated	No Content
405	Invalid input	No Content

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/discussion/end?sessionId=string
```

Example HTTP response

Response 200

```
{
  "active" : true,
  "duration" : "00:03:00",
  "endedAt" : "2023-05-16 10:00:00",
  "posts" : [ {
    "author" : "Max",
    "content" : "Let's estimate together.",
    "createdAt" : "2023-05-16 10:00:00",
    "updatedAt" : "2023-05-16 10:00:00"
  } ],
  "startedAt" : "2023-05-16 10:00:00",
  "topic" : "General"
}
```

2.2.3. Create a discussion post

```
POST /api/discussion/post
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the discussion	string
Body	**body** *required*		DiscussionPostCreate

Responses

HTTP Code	Description	Schema
201	Discussion post created	DiscussionPost
400	Invalid session ID	No Content
401	User not authenticated	No Content

HTTP Code	Description	Schema
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/discussion/post?sessionId=string
```

Request body

```
{
  "content" : "Let's estimate together."
}
```

Example HTTP response

Response 201

```
{
  "author" : "Max",
  "content" : "Let's estimate together.",
  "createdAt" : "2023-05-16 10:00:00",
  "updatedAt" : "2023-05-16 10:00:00"
}
```

2.2.4. Start a discussion

```
POST /api/discussion/start
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the discussion	string
Body	**body** *required*		DiscussionStart

Responses

HTTP Code	Description	Schema
200	Discussion started	Discussion
400	Invalid session ID	No Content
401	User not authenticated	No Content
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/discussion/start?sessionId=string
```

Request body

```
{
  "topic" : "General"
}
```

Example HTTP response

Response 200

```
{
  "active" : true,
  "duration" : "00:03:00",
  "endedAt" : "2023-05-16 10:00:00",
  "posts" : [ {
    "author" : "Max",
    "content" : "Let's estimate together.",
    "createdAt" : "2023-05-16 10:00:00",
    "updatedAt" : "2023-05-16 10:00:00"
  } ],
  "startedAt" : "2023-05-16 10:00:00",
  "topic" : "General"
}
```

2.3. Estimation

All about estimations

2.3.1. Give an estimation

```
POST /api/estimation/give
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the estimation round	string
Body	**body** *required*		EstimationGive

Responses

HTTP Code	Description	Schema
201	Estimation given	Estimation
400	Invalid session ID	No Content
401	User not authenticated	No Content
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Security

Type	Name
apiKey	bearer-auth

Example HTTP request

Request path

```
/api/estimation/give?sessionId=string
```

Request body

```
{
  "estimationValue" : "3"
}
```

Example HTTP response

Response 201

```
{
  "createdAt" : "2023-05-16 10:00:00",
  "estimationValue" : "3",
  "estimator" : "Max",
  "updatedAt" : "2023-05-16 10:00:00"
}
```

2.3.2. Finish the estimation round

```
POST /api/estimation/round/finish
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the estimation round	string

Responses

HTTP Code	Description	Schema
200	Estimation round finished	EstimationRound
400	Invalid session ID	No Content
401	User not authenticated	No Content
405	Invalid input	No Content

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/estimation/round/finish?sessionId=string
```

Example HTTP response

Response 200

```
{
  "backlogItemNumber" : "US001",
  "estimations" : [ {
    "createdAt" : "2023-05-16 10:00:00",
    "estimationValue" : "3",
    "estimator" : "Max",
    "updatedAt" : "2023-05-16 10:00:00"
  } ],
  "finishedAt" : "2023-05-16 10:00:00",
  "startedAt" : "2023-05-16 10:00:00"
}
```

2.3.3. Returns the current estimation round

```
GET /api/estimation/round/info
```

Parameters

Type	Name	Schema
Query	**sessionId** *required*	string

Responses

HTTP Code	Description	Schema
200	Estimation round found	EstimationRound
400	Invalid session ID	No Content
401	User not authenticated	No Content

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/estimation/round/info?sessionId=string
```

Example HTTP response

Response 200

```
{
  "backlogItemNumber" : "US001",
  "estimations" : [ {
    "createdAt" : "2023-05-16 10:00:00",
    "estimationValue" : "3",
    "estimator" : "Max",
    "updatedAt" : "2023-05-16 10:00:00"
  } ],
  "finishedAt" : "2023-05-16 10:00:00",
  "startedAt" : "2023-05-16 10:00:00"
}
```

2.3.4. Start the next estimation round

```
POST /api/estimation/round/next
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the estimation round	string

Responses

HTTP Code	Description	Schema
201	Estimation round started	EstimationRound
400	Invalid session ID	No Content
401	User not authenticated	No Content
405	Invalid input	No Content

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/estimation/round/next?sessionId=string
```

Example HTTP response

Response 201

```
{
  "backlogItemNumber" : "US001",
  "estimations" : [ {
    "createdAt" : "2023-05-16 10:00:00",
    "estimationValue" : "3",
    "estimator" : "Max",
    "updatedAt" : "2023-05-16 10:00:00"
  } ],
  "finishedAt" : "2023-05-16 10:00:00",
  "startedAt" : "2023-05-16 10:00:00"
}
```

2.3.5. Start a estimation round

```
POST /api/estimation/round/start
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the estimation round	string
Body	**body** *required*		EstimationRoundStart

Responses

HTTP Code	Description	Schema
201	Estimation round started	EstimationRound
400	Invalid session ID	No Content
401	User not authenticated	No Content
404	Backlog item not found	No Content
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Security

Type	Name
apiKey	bearer-auth

Example HTTP request

Request path

```
/api/estimation/round/start?sessionId=string
```

Request body

```
{
  "backlogItemNumber" : "US001"
}
```

Example HTTP response

Response 201

```
{
  "backlogItemNumber" : "US001",
  "estimations" : [ {
    "createdAt" : "2023-05-16 10:00:00",
    "estimationValue" : "3",
    "estimator" : "Max",
    "updatedAt" : "2023-05-16 10:00:00"
  } ],
  "finishedAt" : "2023-05-16 10:00:00",
  "startedAt" : "2023-05-16 10:00:00"
}
```

2.3.6. Returns a summary of the estimation round

```
GET /api/estimation/round/summary
```

Parameters

Type	Name	Schema
Query	**sessionId** *required*	string

Responses

HTTP Code	Description	Schema
200	Estimation summary found	EstimationSummery
400	Invalid session ID	No Content
401	User not authenticated	No Content

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/estimation/round/summary?sessionId=string
```

Example HTTP response

Response 200

```
{
  "backlogItemNumber" : "US001",
  "consensusReached" : true,
  "duration" : "00:05:00",
  "estimationResults" : [ {
    "estimationValue" : "3",
    "estimators" : [ "Max", "John", "Noha" ]
  } ],
  "finishedAt" : "2023-05-16 10:00:00",
  "numberOfEstimators" : 6,
  "recommendation" : "3",
  "startedAt" : "2023-05-16 10:00:00",
  "totalEstimators" : 8
}
```

2.4. Session

All about sessions

2.4.1. Create a session

```
POST /api/session
```

Parameters

Type	Name	Schema
Body	**body** *required*	SessionCreate

Responses

HTTP Code	Description	Schema
201	Session created	Session
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Example HTTP request

Request path

```
/api/session
```

Request body

```
{
  "deck" : {
    "cards" : [ {
      "colors" : {
        "backTextColor" : "#DC3545",
        "backgroundColor" : "#F8F9FA",
        "frontTextColor" : "#000000",
        "hoverColor" : "#FFF0C1"
      },
      "value" : "1"
    } ],
    "name" : "MyDeck"
  },
  "team" : "MyTeam",
  "username" : "MyName"
}
```

Example HTTP response

Response 201

```
{
  "backlog" : {
    "items" : [ {
      "description" : "As an authorized User I want to see the list of items so that I can select one",
      "estimation" : "3",
      "number" : "US001",
      "priority" : "5",
      "title" : "List Items"
    } ],
    "name" : "MyBacklog"
  },
  "createdAt" : "2023-05-16 10:00:00",
  "deck" : {
    "cards" : [ {
      "colors" : {
        "backTextColor" : "#DC3545",
        "backgroundColor" : "#F8F9FA",
        "frontTextColor" : "#000000",
        "hoverColor" : "#FFF0C1"
      },
      "value" : "1"
    } ],
    "name" : "MyDeck"
  },
  "discussion" : {
    "active" : true,
    "duration" : "00:03:00",
    "endedAt" : "2023-05-16 10:00:00",
    "posts" : [ {
      "author" : "Max",
      "content" : "Let's estimate together.",
      "createdAt" : "2023-05-16 10:00:00",
      "updatedAt" : "2023-05-16 10:00:00"
    } ],
    "startedAt" : "2023-05-16 10:00:00",
    "topic" : "General"
  },
  "estimationRound" : {
    "backlogItemNumber" : "US001",
    "estimations" : [ {
      "createdAt" : "2023-05-16 10:00:00",
      "estimationValue" : "3",
      "estimator" : "Max",
      "updatedAt" : "2023-05-16 10:00:00"
    } ],
    "finishedAt" : "2023-05-16 10:00:00",
    "startedAt" : "2023-05-16 10:00:00"
  },
  "estimationSummary" : {
    "backlogItemNumber" : "US001",
    "consensusReached" : true,
    "duration" : "00:05:00",
    "estimationResults" : [ {
      "estimationValue" : "3",
      "estimators" : [ "Max", "John", "Noha" ]
    } ],
    "finishedAt" : "2023-05-16 10:00:00",
    "numberOfEstimators" : 6,
    "recommendation" : "3",
    "startedAt" : "2023-05-16 10:00:00",
    "totalEstimators" : 8
  },
  "id" : "0ea43720-2256-46de-8854-0359ef5c4977",
  "team" : {
    "members" : [ {
      "active" : true,
      "name" : "John",
      "role" : "MODERATOR"
    } ],
    "name" : "MyTeam"
  }
}
```

2.4.2. Returns a paginated list of sessions

```
GET /api/session
```

Parameters

Type	Name	Description	Schema	Default
Query	**page** *optional*	Zero-based page index (0..N)	integer	`0`
Query	**size** *optional*	The size of the page to be returned	integer	`10`
Query	**sort** *optional*	Sorting criteria in the format: property,(asc|desc). Default sort order is ascending. Multiple sort criteria are supported.	< string > array(multi)	

Responses

HTTP Code	Description	Schema
200	OK	SessionPage

Produces

- `application/json`

Example HTTP request

Request path

```
/api/session
```

Example HTTP response

Response 200

```
{
  "info" : {
    "NumberOfItems" : 10,
    "TotalItems" : 1000,
    "TotalPage" : 100,
    "offset" : 0,
    "pageNumber" : 1,
    "pageSize" : 10
  },
  "items" : [ {
    "activeMembers" : 3,
    "createdAt" : "2023-05-16 10:00:00",
    "deckCards" : "1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ?, ∞, □",
    "deckName" : "Fibonacci",
    "estimatedBacklogItems" : 3,
    "id" : "0ea43720-2256-46de-8854-0359ef5c4977",
    "teamName" : "MyTeam",
    "totalBacklogItems" : 10,
    "totalDiscussionPosts" : 36,
    "totalEstimationRounds" : 12,
    "totalEstimations" : 22,
    "totalMembers" : 6
  } ]
}
```

2.4.3. Returns a session by public ID

```
GET /api/session/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	Public ID of the session	string

Responses

HTTP Code	Description	Schema
200	Session found	Session
400	Invalid ID	No Content
401	User not authenticated	No Content
404	Session not found	No Content

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/session/string
```

Example HTTP response

Response 200

```
{
  "backlog" : {
    "items" : [ {
      "description" : "As an authorized User I want to see the list of items so that I can select one",
      "estimation" : "3",
      "number" : "US001",
      "priority" : "5",
      "title" : "List Items"
    } ],
    "name" : "MyBacklog"
  },
  "createdAt" : "2023-05-16 10:00:00",
  "deck" : {
    "cards" : [ {
      "colors" : {
        "backTextColor" : "#DC3545",
        "backgroundColor" : "#F8F9FA",
        "frontTextColor" : "#000000",
        "hoverColor" : "#FFF0C1"
      },
      "value" : "1"
    } ],
    "name" : "MyDeck"
  },
  "discussion" : {
    "active" : true,
    "duration" : "00:03:00",
    "endedAt" : "2023-05-16 10:00:00",
    "posts" : [ {
      "author" : "Max",
      "content" : "Let's estimate together.",
      "createdAt" : "2023-05-16 10:00:00",
      "updatedAt" : "2023-05-16 10:00:00"
    } ],
    "startedAt" : "2023-05-16 10:00:00",
    "topic" : "General"
  },
  "estimationRound" : {
    "backlogItemNumber" : "US001",
    "estimations" : [ {
      "createdAt" : "2023-05-16 10:00:00",
      "estimationValue" : "3",
      "estimator" : "Max",
      "updatedAt" : "2023-05-16 10:00:00"
    } ],
    "finishedAt" : "2023-05-16 10:00:00",
    "startedAt" : "2023-05-16 10:00:00"
  },
  "estimationSummary" : {
    "backlogItemNumber" : "US001",
    "consensusReached" : true,
    "duration" : "00:05:00",
    "estimationResults" : [ {
      "estimationValue" : "3",
      "estimators" : [ "Max", "John", "Noha" ]
    } ],
    "finishedAt" : "2023-05-16 10:00:00",
    "numberOfEstimators" : 6,
    "recommendation" : "3",
    "startedAt" : "2023-05-16 10:00:00",
    "totalEstimators" : 8
  },
  "id" : "0ea43720-2256-46de-8854-0359ef5c4977",
  "team" : {
    "members" : [ {
      "active" : true,
      "name" : "John",
      "role" : "MODERATOR"
    } ],
    "name" : "MyTeam"
  }
}
```

2.4.4. Delete a session by public ID

```
DELETE /api/session/{id}
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	Public ID of the session	string

Responses

HTTP Code	Description	Schema
200	Session deleted	No Content
400	Invalid ID	No Content
401	User not authenticated	No Content
404	Session not found	No Content

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/session/string
```

2.4.5. Join a session by public ID

```
POST /api/session/{id}/join
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	Public ID of the session	string
Body	**body** *required*		SessionJoin

Responses

HTTP Code	Description	Schema
200	Session joined	Session
405	Invalid input	No Content

Consumes

- `application/json`

Produces

- `application/json`

Example HTTP request

Request path

```
/api/session/string/join
```

Request body

```
{
  "spectator" : false,
  "username" : "MyName"
}
```

Example HTTP response

Response 200

```
{
  "backlog" : {
    "items" : [ {
      "description" : "As an authorized User I want to see the list of items so that I can select one",
      "estimation" : "3",
      "number" : "US001",
      "priority" : "5",
      "title" : "List Items"
    } ],
    "name" : "MyBacklog"
  },
  "createdAt" : "2023-05-16 10:00:00",
  "deck" : {
    "cards" : [ {
      "colors" : {
        "backTextColor" : "#DC3545",
        "backgroundColor" : "#F8F9FA",
        "frontTextColor" : "#000000",
        "hoverColor" : "#FFF0C1"
      },
      "value" : "1"
    } ],
    "name" : "MyDeck"
  },
  "discussion" : {
    "active" : true,
    "duration" : "00:03:00",
    "endedAt" : "2023-05-16 10:00:00",
    "posts" : [ {
      "author" : "Max",
      "content" : "Let's estimate together.",
      "createdAt" : "2023-05-16 10:00:00",
      "updatedAt" : "2023-05-16 10:00:00"
    } ],
    "startedAt" : "2023-05-16 10:00:00",
    "topic" : "General"
  },
  "estimationRound" : {
    "backlogItemNumber" : "US001",
    "estimations" : [ {
      "createdAt" : "2023-05-16 10:00:00",
      "estimationValue" : "3",
      "estimator" : "Max",
      "updatedAt" : "2023-05-16 10:00:00"
    } ],
    "finishedAt" : "2023-05-16 10:00:00",
    "startedAt" : "2023-05-16 10:00:00"
  },
  "estimationSummary" : {
    "backlogItemNumber" : "US001",
    "consensusReached" : true,
    "duration" : "00:05:00",
    "estimationResults" : [ {
      "estimationValue" : "3",
      "estimators" : [ "Max", "John", "Noha" ]
    } ],
    "finishedAt" : "2023-05-16 10:00:00",
    "numberOfEstimators" : 6,
    "recommendation" : "3",
    "startedAt" : "2023-05-16 10:00:00",
    "totalEstimators" : 8
  },
  "id" : "0ea43720-2256-46de-8854-0359ef5c4977",
  "team" : {
    "members" : [ {
      "active" : true,
      "name" : "John",
      "role" : "MODERATOR"
    } ],
    "name" : "MyTeam"
  }
}
```

2.4.6. Leave a session by public ID

```
POST /api/session/{id}/leave
```

Parameters

Type	Name	Description	Schema
Path	**id** *required*	Public ID of the session	string

Responses

HTTP Code	Description	Schema
200	Session leaved	Session
401	User not authenticated	No Content

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/session/string/leave
```

Example HTTP response

Response 200

```
{
  "backlog" : {
    "items" : [ {
      "description" : "As an authorized User I want to see the list of items so that I can select one",
      "estimation" : "3",
      "number" : "US001",
      "priority" : "5",
      "title" : "List Items"
    } ],
    "name" : "MyBacklog"
  },
  "createdAt" : "2023-05-16 10:00:00",
  "deck" : {
    "cards" : [ {
      "colors" : {
        "backTextColor" : "#DC3545",
        "backgroundColor" : "#F8F9FA",
        "frontTextColor" : "#000000",
        "hoverColor" : "#FFF0C1"
      },
      "value" : "1"
    } ],
    "name" : "MyDeck"
  },
  "discussion" : {
    "active" : true,
    "duration" : "00:03:00",
    "endedAt" : "2023-05-16 10:00:00",
    "posts" : [ {
      "author" : "Max",
      "content" : "Let's estimate together.",
      "createdAt" : "2023-05-16 10:00:00",
      "updatedAt" : "2023-05-16 10:00:00"
    } ],
    "startedAt" : "2023-05-16 10:00:00",
    "topic" : "General"
  },
  "estimationRound" : {
    "backlogItemNumber" : "US001",
    "estimations" : [ {
      "createdAt" : "2023-05-16 10:00:00",
      "estimationValue" : "3",
      "estimator" : "Max",
      "updatedAt" : "2023-05-16 10:00:00"
    } ],
    "finishedAt" : "2023-05-16 10:00:00",
    "startedAt" : "2023-05-16 10:00:00"
  },
  "estimationSummary" : {
    "backlogItemNumber" : "US001",
    "consensusReached" : true,
    "duration" : "00:05:00",
    "estimationResults" : [ {
      "estimationValue" : "3",
      "estimators" : [ "Max", "John", "Noha" ]
    } ],
    "finishedAt" : "2023-05-16 10:00:00",
    "numberOfEstimators" : 6,
    "recommendation" : "3",
    "startedAt" : "2023-05-16 10:00:00",
    "totalEstimators" : 8
  },
  "id" : "0ea43720-2256-46de-8854-0359ef5c4977",
  "team" : {
    "members" : [ {
      "active" : true,
      "name" : "John",
      "role" : "MODERATOR"
    } ],
    "name" : "MyTeam"
  }
}
```

2.5. Team

All about teams

2.5.1. Returns a team

```
GET /api/team
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the team	string

Responses

HTTP Code	Description	Schema
200	Team found	Team
400	Invalid session ID	No Content
401	User not authenticated	No Content
404	Team not found	No Content

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/team?sessionId=string
```

Example HTTP response

Response 200

```
{
  "members" : [ {
    "active" : true,
    "name" : "John",
    "role" : "MODERATOR"
  } ],
  "name" : "MyTeam"
}
```

2.5.2. Rename a team

```
PATCH /api/team/rename
```

Parameters

Type	Name	Description	Schema
Query	**sessionId** *required*	Session ID of the team	string
Body	**body** *required*		TeamRename

Responses

HTTP Code	Description	Schema
200	Team renamed	Team
400	Invalid session ID	No Content

HTTP Code	Description	Schema
401	User not authenticated	No Content
404	Team not found	No Content

Consumes

- `application/json`

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/team/rename?sessionId=string
```

Request body

```
{
  "name" : "MyTeam"
}
```

Example HTTP response

Response 200

```
{
  "members" : [ {
    "active" : true,
    "name" : "John",
    "role" : "MODERATOR"
  } ],
  "name" : "MyTeam"
}
```

2.6. User

All about users

2.6.1. Returns the current user

```
GET /api/user/info
```

Responses

HTTP Code	Description	Schema
200	User found	User
401	User not authenticated	No Content

Produces

- `application/json`

Security

Type	Name
apiKey	**bearer-auth**

Example HTTP request

Request path

```
/api/user/info
```

Example HTTP response

Response 200

```
{
  "active" : true,
  "name" : "MyName",
  "role" : "MODERATOR"
}
```

Chapter 3. Definitions

3.1. Backlog

Name	Description	Schema
items *required*	Set of backlog items **Example** : [" BacklogItem "]	< BacklogItem > array
name *required*	The name of the backlog **Length** : 1 - 32 **Pattern** : "^[A-Za-z0-9 _\\-]{1,32}$" **Example** : "MyBacklog"	string

3.2. BacklogItem

A backlog item

<table>
<tr><th>Name</th><th>Description</th><th>Schema</th></tr>
<tr><td><b>description</b>
<i>required</i></td><td>The description of the backlog item
<b>Length</b> : 0 - 256
<b>Pattern</b> : "^[A-Za-z0-9 !\"#$%&'()*+,-./:;<⇒?@\\[\\]^_{|}~]{0,256}$"`
<b>Example</b> : "As an authorized User I want to see the list of items so that I can select one"</td><td>string</td></tr>
<tr><td><b>estimation</b>
<i>required</i></td><td>The estimation of the backlog item
<b>Length</b> : 0 - 3
<b>Pattern</b> : "^[A-Za-z0-9½? ∞ \\p{So}]{0,3}$"
<b>Example</b> : "3"</td><td>string</td></tr>
<tr><td><b>number</b>
<i>required</i></td><td>The number of the backlog item
<b>Length</b> : 5
<b>Pattern</b> : "^US[0-9]{3}$"
<b>Example</b> : "US001"</td><td>string</td></tr>
<tr><td><b>priority</b>
<i>required</i></td><td>The priority of the backlog item
<b>Length</b> : 0 - 32
<b>Pattern</b> : "^[A-Za-z0-9 \\-]{0,32}$"
<b>Example</b> : "5"</td><td>string</td></tr>
<tr><td><b>title</b>
<i>required</i></td><td>The title of the backlog item
<b>Length</b> : 1 - 64
<b>Pattern</b> : "^[A-Za-z0-9 _\\-]{1,64}$"
<b>Example</b> : "List Items"</td><td>string</td></tr>
</table>

3.3. BacklogItemAdd

Name	Description	Schema
description *optional*	The description of the backlog item **Length** : 0 - 256 **Pattern** : "^[A-Za-z0-9 !\"#$%&'()*+,-./:;<=>?@\\[\\]^_{\|}~]{0,256}$"` **Example** : "As an authorized User I want to see the list of items so that I can select one"	string
title *required*	The title of the backlog item **Length** : 1 - 64 **Pattern** : "^[A-Za-z0-9 _\\-]{1,64}$" **Example** : "List Items"	string

3.4. BacklogItemMove

Name	Description	Schema
newIndex *optional*	The new index of the backlog item **Example** : 1	integer (int32)

3.5. BacklogItemUpdate

Name	Description	Schema
description *required*	The description of the backlog item **Length** : 0 - 256 **Pattern** : "^[A-Za-z0-9 !\"#$%&'()*+,-./:;<=>?@\\[\\]^_{\|}~]{0,256}$"` **Example** : "As an authorized User I want to see the list of items so that I can select one"	string
estimation *required*	The estimation of the backlog item **Length** : 0 - 3 **Pattern** : "^[A-Za-z0-9½? ∞ \\p{So}]{0,3}$" **Example** : "3"	string
priority *required*	The priority of the backlog item **Length** : 0 - 32 **Pattern** : "^[A-Za-z0-9 \\-]{0,32}$" **Example** : "5"	string
title *required*	The title of the backlog item **Length** : 1 - 64 **Pattern** : "^[A-Za-z0-9 _\\-]{1,64}$" **Example** : "List Items"	string

3.6. BacklogRename

Name	Description	Schema
name *required*	The name of the backlog **Length** : `1 - 32` **Pattern** : `"^[A-Za-z0-9 _\\-]{1,32}$"` **Example** : `"MyBacklog"`	string

3.7. Deck

Name	Description	Schema
cards *optional*	Set of cards **Example** : `[ "DeckCard" ]`	< DeckCard > array
name *required*	The name of the deck **Length** : `1 - 32` **Pattern** : `"^[A-Za-z0-9 _\\-]{1,32}$"` **Example** : `"MyDeck"`	string

3.8. DeckCard

A card

Name	Description	Schema
colors *required*	**Example** : DeckCardColors	DeckCardColors
value *required*	The estimation value **Length** : `1 - 3` **Example** : `"1"`	string

3.9. DeckCardColors

The colors of the card

Name	Description	Schema
backTextColor *required*	The back text color of the card **Length** : `3 - 7` **Pattern** : `"^#?([a-fA-F0-9]{3}|[a-fA-F0-9]{6})$"` **Example** : `"#DC3545"`	string
backgroundColor *required*	The background color of the card **Length** : `3 - 7` **Pattern** : `"^#?([a-fA-F0-9]{3}|[a-fA-F0-9]{6})$"` **Example** : `"#F8F9FA"`	string

Name	Description	Schema
frontTextColor *required*	The front text color of the card **Length** : `3 - 7` **Pattern** : `"^#?([a-fA-F0-9]{3}|[a-fA-F0-9]{6})$"` **Example** : `"#000000"`	string
hoverColor *required*	The hover color of the card **Length** : `3 - 7` **Pattern** : `"^#?([a-fA-F0-9]{3}|[a-fA-F0-9]{6})$"` **Example** : `"#FFF0C1"`	string

3.10. Discussion

Name	Description	Schema
active *optional*	Is discussion active **Example** : `true`	boolean
duration *required*	The duration of the estimation round **Pattern** : `"^([0-1][0-9]|2[0-3]):([0-5][0-9]):([0-5][0-9])$"` **Example** : `"00:03:00"`	string
endedAt *optional*	The end date of the discussion **Example** : `"2023-05-16 10:00:00"`	string
posts *required*	**Example** : `[ "DiscussionPost" ]`	< DiscussionPost > array
startedAt *optional*	The start date of the discussion **Example** : `"2023-05-16 10:00:00"`	string
topic *required*	The topic of the discussion **Length** : `1 - 32` **Pattern** : `"^[A-Za-z0-9 _\\-]{1,32}$"` **Example** : `"General"`	string

3.11. DiscussionPost

A discussion post

Name	Description	Schema
author *required*	The name of the author **Length** : `1 - 32` **Pattern** : `"^[A-Za-z ]{1,32}$"` **Example** : `"Max"`	string

Name	Description	Schema
content *required*	The content of the discussion post **Length** : 1 - 512 **Pattern** : "^[A-Za-z0-9 !\"#$%&'()*+,-./:;<⇒?@\\[\\]^_{\|}~\\r\\n\\p{So}]{0,256}$"` **Example** : "Let’s estimate together."	string
createdAt *required*	The creation date of the discussion post **Example** : "2023-05-16 10:00:00"	string
updatedAt *required*	The update date of the discussion post **Example** : "2023-05-16 10:00:00"	string

3.12. DiscussionPostCreate

Name	Description	Schema
content *required*	The content of the discussion post **Length** : 1 - 512 **Pattern** : "^[A-Za-z0-9 !\"#$%&'()*+,-./:;<⇒?@\\[\\]^_{\|}~\\r\\n\\p{So}]{0,256}$"` **Example** : "Let’s estimate together."	string

3.13. DiscussionStart

Name	Description	Schema
topic *required*	The topic of the discussion **Length** : 1 - 32 **Pattern** : "^[A-Za-z0-9 _\\-]{1,32}$" **Example** : "General"	string

3.14. Estimation

A estimation

Name	Description	Schema
createdAt *required*	The creation date of the estimation **Example** : "2023-05-16 10:00:00"	string
estimationValue *required*	The estimation value of the backlog item **Length** : 1 - 3 **Example** : "3"	string

Name	Description	Schema
estimator *required*	The name of the estimator **Length** : `1 - 32` **Pattern** : `"^[A-Za-z ]{1,32}$"` **Example** : `"Max"`	string
updatedAt *required*	The update date of the estimation **Example** : `"2023-05-16 10:00:00"`	string

3.15. EstimationGive

Name	Description	Schema
estimationValue *required*	The estimation of the backlog item **Length** : `1 - 3` **Example** : `"3"`	string

3.16. EstimationResult

A estimation result

Name	Description	Schema
estimationValue *required*	The estimation value of the backlog item **Length** : `1 - 3` **Example** : `"3"`	string
estimators *required*	Set of estimators **Example** : `[ "Max", "John", "Noha" ]`	< string > array

3.17. EstimationRound

Name	Description	Schema
backlogItemNumber *required*	The number of the backlog item **Length** : `5` **Pattern** : `"^US[0-9]{3}$"` **Example** : `"US001"`	string
estimations *required*	**Example** : `[ "Estimation" ]`	< Estimation > array
finishedAt *optional*	The finish date of the estimation round **Example** : `"2023-05-16 10:00:00"`	string
startedAt *required*	The start date of the estimation round **Example** : `"2023-05-16 10:00:00"`	string

3.18. EstimationRoundStart

Name	Description	Schema
backlogItemNumber *required*	The number of the backlog item **Length** : `5` **Pattern** : `"^US[0-9]{3}$"` **Example** : `"US001"`	string

3.19. EstimationSummery

Name	Description	Schema
backlogItemNumber *required*	The number of the backlog item **Length** : `5` **Pattern** : `"^US[0-9]{3}$"` **Example** : `"US001"`	string
consensusReached *optional*	Is a consensus reached and all estimations are the same **Example** : `true`	boolean
duration *required*	The duration of the estimation round **Pattern** : `"^([0-1][0-9]|2[0-3]):([0-5][0-9]):([0-5][0-9])$"` **Example** : `"00:05:00"`	string
estimationResults *required*	Set of estimation results **Example** : `[ "EstimationResult" ]`	< EstimationResult > array
finishedAt *optional*	The finish date of the estimation round **Example** : `"2023-05-16 10:00:00"`	string
numberOfEstimators *optional*	The number of estimators that give an estimation in the estimation round **Minimum value** : `0` **Maximum value** : `8` **Example** : `6`	integer (int32)
recommendation *required*	The recommended estimation value of the estimation round **Length** : `0 - 3` **Example** : `"3"`	string
startedAt *required*	The start date of the estimation round **Example** : `"2023-05-16 10:00:00"`	string
totalEstimators *optional*	The total count of estimators in the estimation round **Minimum value** : `0` **Maximum value** : `8` **Example** : `8`	integer (int32)

3.20. Session

Name	Description	Schema
backlog *required*	**Example** : Backlog	Backlog
createdAt *required*	The creation date of the session **Example** : `"2023-05-16 10:00:00"`	string
deck *required*	**Example** : Deck	Deck
discussion *required*	**Example** : Discussion	Discussion
estimationRound *optional*	**Example** : EstimationRound	EstimationRound
estimationSummary *optional*	**Example** : EstimationSummery	EstimationSummery
id *required*	The public ID of the session **Example** : `"0ea43720-2256-46de-8854-0359ef5c4977"`	string
team *required*	**Example** : Team	Team

3.21. SessionCreate

The data to create a session

Name	Description	Schema
deck *required*	**Example** : Deck	Deck
team *required*	The name of the team **Length** : `1 - 32` **Pattern** : `"^[A-Za-z0-9 _\\-]{1,32}$"` **Example** : `"MyTeam"`	string
username *required*	The name of the user that moderate the session **Length** : `1 - 32` **Pattern** : `"^[A-Za-z ]{1,32}$"` **Example** : `"MyName"`	string

3.22. SessionJoin

Name	Description	Schema
spectator *optional*	Spectate or participate the session **Default** : `false` **Example** : `false`	boolean
username *required*	The name of the user that participate or spectate the session **Length** : `1 - 32` **Pattern** : `"^[A-Za-z ]{1,32}$"` **Example** : `"MyName"`	string

3.23. SessionPage

Name	Description	Schema
info *required*	**Example** : SessionPageInfo	SessionPageInfo
items *required*	List of session page items **Example** : `[ "SessionPageItem" ]`	< SessionPageItem > array

3.24. SessionPageInfo

Info of the session page

Name	Description	Schema
NumberOfItems *optional*	The number of session page items **Example** : `10`	integer (int32)
TotalItems *optional*	The total number of session page items **Example** : `1000`	integer (int64)
TotalPage *optional*	The total number of session pages **Example** : `100`	integer (int32)
offset *optional*	The offset of session page **Example** : `0`	integer (int64)
pageNumber *optional*	The page number of session page **Example** : `1`	integer (int32)
pageSize *optional*	The page size of session page **Example** : `10`	integer (int32)

3.25. SessionPageItem

The session page item of the session page

Name	Description	Schema
activeMembers *optional*	The count of active members **Minimum value** : `0` **Maximum value** : `8` **Example** : `3`	integer (int32)
createdAt *required*	The creation date of the session **Example** : `"2023-05-16 10:00:00"`	string
deckCards *required*	The cards of the deck **Pattern** : `"` `\\s*[` `,]{1,3}\\s*(\\s*,\\s*[^,]{0,3}\\s*)*$"` **Example** : `"1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ?,` `∞` `, ▯"`	string
deckName *required*	The name of the deck **Length** : `1 - 32` **Pattern** : `"^[A-Za-z0-9 _\\-]{1,32}$"` **Example** : `"Fibonacci"`	string
estimatedBacklogItems *optional*	The total count of estimated backlog items **Minimum value** : `0` **Maximum value** : `1000` **Example** : `3`	integer (int32)
id *required*	The public ID of the session **Example** : `"0ea43720-2256-46de-8854-0359ef5c4977"`	string
teamName *required*	The name of team **Length** : `1 - 32` **Pattern** : `"^[A-Za-z0-9 _\\-]{1,32}$"` **Example** : `"MyTeam"`	string
totalBacklogItems *optional*	The total count of backlog items in the backlog **Minimum value** : `0` **Maximum value** : `1000` **Example** : `10`	integer (int32)
totalDiscussionPosts *optional*	The total count of discussion posts **Minimum value** : `0` **Maximum value** : `10000` **Example** : `36`	integer (int32)
totalEstimationRounds *optional*	The total count of estimation rounds **Example** : `12`	integer (int32)
totalEstimations *optional*	The total count of estimations **Example** : `22`	integer (int32)
totalMembers *optional*	The total count of members in the team **Minimum value** : `1` **Maximum value** : `8` **Example** : `6`	integer (int32)

3.26. Team

Name	Description	Schema
members *required*	Set of team members **Example** : `[ " TeamMember " ]`	< TeamMember > array
name *required*	The name of team **Length** : `1 - 32` **Pattern** : `"^[A-Za-z0-9 _\\-]{1,32}$"` **Example** : `"MyTeam"`	string

3.27. TeamMember

A team member

Name	Description	Schema
active *optional*	Is team member active **Example** : `true`	boolean
name *required*	The name of team member **Length** : `1 - 32` **Pattern** : `"^[A-Za-z ]{1,32}$"` **Example** : `"John"`	string
role *required*	The role of the team member **Example** : `"MODERATOR"`	enum (MODERATOR, PARTICIPANT, SPECTATOR)

3.28. TeamRename

Name	Description	Schema
name *required*	The name of the team **Length** : `1 - 32` **Pattern** : `"^[A-Za-z0-9 _\\-]{1,32}$"` **Example** : `"MyTeam"`	string

3.29. User

Name	Description	Schema
active *optional*	Is user active **Example** : `true`	boolean

Name	Description	Schema
name *required*	The name of the user **Length** : `1 - 32` **Pattern** : `"^[A-Za-z ]{1,32}$"` **Example** : `"MyName"`	string
role *required*	The role of the user **Example** : `"MODERATOR"`	enum (MODERATOR, PARTICIPANT, SPECTATOR)

Chapter 4. Security

4.1. bearer-auth

Bearer token

Type : apiKey
Name : Authorization
In : HEADER