**API Specification Doc**

***Think Time App* **

| **Version** | **Date** | **Author** | **Description** |
| --- | --- | --- | --- |
| 3.0.0 | 03-Feb-2023 | Abhay | Updated version 3  API specification for Think time app |

## 

Index

[1. User Onboarding](#_bit7idv2ancx)

[Request](#_kd6tjl5w5wjo)

[Response](#_8b7ijrpuamb8)

[2. Login](#_9t094e45sg90)

[Request](#_38pypefkb82)

[Response](#_47awg09ehquu)

[3. Create Timesheet](#_9mc75gzfq3m3)

[Request](#_xtszbus9yomc)

[Response](#_xuhh3y4ufz7)

[4. Update Timesheet](#_p8sbwmpv82e8)

[Request](#_iaps2dh7l2m2)

[Response](#_fvf6lulvf6wv)

[5. Get Timesheet By ID](#_p8sbwmpv82e8)

[Request](#_iaps2dh7l2m2)

[Response](#_fvf6lulvf6wv)

[4. Update Timesheet](#_p8sbwmpv82e8)

[Request](#_iaps2dh7l2m2)

[Response](#_fvf6lulvf6wv)

[Conventions](#_uk42zio712fb)

[Status Codes](#_of6ke88xwee4)

## 

Methods

## 1. On Board a User

On board the user into the system

## URL

| **Method** | **URL** |
| --- | --- |
| **POST** | api/user/ |

## Request

| **Type** | **Body** |
| --- | --- |
| POST | {  "name":"Sayak De",  "password":"Pass@ABC1",  "email":"sayak@gmail.com",  "mobileNo":1020029282,  "role": "TEAM\_LEAD"  } |

## Response

| **Status** | **Response** |
| --- | --- |
| 201 | {  "success": **true**,  "message": "User has been created successfully",  "data": {  "name": "sayak",  "password": "xxxx",  "email": "sayak@gmail.com",  "mobileNo": 1020029282,  "role": "TEAM\_LEAD",  "hashedPassword": "xxxx",  "createdAt": "2023-02-09T05:56:56.945Z",  "id": 2  }  } |
| 403 | {"error":"API key is missing."} |
| 400 | {"error":"Please provide email."} |
| 400 | {"error":"Please provide mobile number."} |
| 400 | {"error":"Please provide username."} |
| 400 | {"error":"Please provide password."} |
| 500 | {"error":"Something went wrong. Please try again later."} |

## 2. Login

Authenticate the user with the system and obtain the auth\_token

## URL

| **Method** | **URL** |
| --- | --- |
| **POST** | api/login/ |

## 

## 

## Request

| **Type** | **Body** |
| --- | --- |
| POST | {  "email": "sayak@gmail.com",  "password": "Pass@ABC1"  } |

**Version 3.0.0**

## Response

| **Status** | **Response** |
| --- | --- |
| 200 | {  "success": **true**,  "message": "User logged in successfully",  "data": "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyUm9sZSI6IlRFQU1fTEVBRCIsImVtYWlsIjoic2F5YWsxQGdtYWlsLmNvbSIsInVzZXJJZCI6MywiaWF0IjoxNjc1OTI0NTM3LCJleHAiOjE2NzU5MzA1Mzd9.GlOgPrHzMQR\_IcMNF3MxE25PTFrnW6rl8Hy0y-uaIwc"  } |
| 401 | {  "success": **false**,  "message": "Invalid email or password",  "error\_code": 401,  "data": {}  } |

## 3. Create Timesheet

Create the timesheet.

## URL

| **Method** | **URL** |
| --- | --- |
| **POST** | api/timesheet/ |

| **Type** | **Params** | **Values** |
| --- | --- | --- |
| HEAD  POST | Authorization  version | Bearer Token  3.0.0 |

## Request

| **Status** | **Response** |
| --- | --- |
| 200 | {  "taskDetail": [  {  "date": "2022-12-02",  "hourMinute": "2:30",  "task": "Created document for Test API/s and shared with Kartik",  "notes": "test notes",  "status": "completed"  },  {  "date": "2022-12-02",  "hourMinute": "2:30",  "task": "Created Postman collection and attached on ticket",  "notes": "test notes",  "status": "completed"  }  ],  "engineerId": “b6dff8a7-7a2f-453d-b225-8e7d2b0b9017”,  "teamLeadId": “50e38ccc-fb37-4fda-b4ca-18a555dd91b1”,  "name": "Wazo\_timesheet",  "projectId": "f7a50527-1c40-47ab-b56b-6b30af826961",  "status": "Active",  "assignedTo": “f3ba8f83-5201-4ed9-8022-24ee6431aeda”  } |
| 400 | {"error":"Please specify valid engineerId"} |
| 400 | {"error":"Please specify valid projectId."} |

## Response

| **Status** | **Response** |
| --- | --- |
| 200 | **{**  **"success": true,**  **"message": "Time sheet has been created successfully",**  **"data": {**  "engineerId": “b6dff8a7-7a2f-453d-b225-8e7d2b0b9017”,  "teamLeadId": “50e38ccc-fb37-4fda-b4ca-18a555dd91b1”,  "name": "Wazo\_timesheet",  "projectId": "f7a50527-1c40-47ab-b56b-6b30af826961",  "status": "Active",  "assignedTo": “f3ba8f83-5201-4ed9-8022-24ee6431aeda”  **"createdAt": "2023-02-09T06:53:10.610Z",**  **"updatedAt": "2023-02-09T06:53:10.610Z",**  **"id": 6,**  **"taskDetails": [**  {  "date": "2022-12-02",  "hourMinute": "2:30",  "task": "Created document for Test API/s and shared with Kartik",  "notes": "test notes",  "status": "completed"  },  {  "date": "2022-12-02",  "hourMinute": "2:30",  "task": "Created Postman collection and attached on ticket",  "notes": "test notes",  "status": "completed"  }  **]**  **}**  **}** |
|  |  |

## 

## 4. Update Timesheet

Update the existing timesheet with updated tasks whenever requested

## URL

| **Method** | **URL** |
| --- | --- |
| **PATCH** | api/timesheet/ |

| **Type** | **Params** | **Values** |
| --- | --- | --- |
| HEAD  POST | Authorization  version | Bearer Token  3.0.0 |

## Request

| **Status** | **Response** |
| --- | --- |
| 200 | {  "taskDetail": [  {  "date": "2022-12-02",  "hourMinute": "2:30",  "task": "Created document for Test API/s and shared with Kartik",  "notes": "test notes",  "status": "completed"  },  {  "date": "2022-12-02",  "hourMinute": "2:30",  "task": "Created Postman collection and attached on ticket",  "notes": "test notes",  "status": "completed"  }  ],  "engineerId": “b6dff8a7-7a2f-453d-b225-8e7d2b0b9017”,  "teamLeadId": “50e38ccc-fb37-4fda-b4ca-18a555dd91b1”,  "name": "Wazo\_timesheet",  "projectId": "f7a50527-1c40-47ab-b56b-6b30af826961",  "status": "Active",  "assignedTo": “f3ba8f83-5201-4ed9-8022-24ee6431aeda”  } |
| 400 | {"error":"Please specify valid engineerId"} |
| 400 | {"error":"Please specify valid projectId."} |

## Response

| **Status** | **Response** |
| --- | --- |
| 200 | **{**  **"success": true,**  **"message": "Time sheet has been updated successfully",**  **"data": [ {**  "engineerId": “b6dff8a7-7a2f-453d-b225-8e7d2b0b9017”,  "teamLeadId": “50e38ccc-fb37-4fda-b4ca-18a555dd91b1”,  "name": "Wazo\_timesheet",  "projectId": "f7a50527-1c40-47ab-b56b-6b30af826961",  "status": "Active",  "assignedTo": “f3ba8f83-5201-4ed9-8022-24ee6431aeda”  **"createdAt": "2023-02-09T06:53:10.610Z",**  **"updatedAt": "2023-02-09T06:53:10.610Z",**  **"id": 6,**  **"taskDetails": [**  {  "date": "2022-12-02",  "hourMinute": "2:30",  "task": "Created document for Test API/s and shared with Kartik",  "notes": "test notes",  "status": "completed"  },  {  "date": "2022-12-02",  "hourMinute": "2:30",  "task": "Created Postman collection and attached on ticket",  "notes": "test notes",  "status": "completed"  }  **]**  **}]**  **}** |
|  |  |

## 5. Get Timesheet By Id

Get the individual timesheet information by id.

## Request

| **Method** | **URL** |
| --- | --- |
| **GET** | api/timesheet/:id |

| **Type** | **Params** | **Values** |
| --- | --- | --- |
| HEAD  Path Variable  POST | Authorization  id  version | Bearer Token  \3.0.0 |

## Request

| **Status** | **Response** |
| --- | --- |
| 200 | {  "taskDetail": [  {  "date": "2022-12-02",  "hourMinute": "2:30",  "task": "Created document for Test API/s and shared with Kartik",  "notes": "test notes",  "status": "completed"  },  {  "date": "2022-12-02",  "hourMinute": "2:30",  "task": "Created Postman collection and attached on ticket",  "notes": "test notes",  "status": "completed"  }  ],  "engineerId": “b6dff8a7-7a2f-453d-b225-8e7d2b0b9017”,  "teamLeadId": “50e38ccc-fb37-4fda-b4ca-18a555dd91b1”,  "name": "Wazo\_timesheet",  "projectId": "f7a50527-1c40-47ab-b56b-6b30af826961",  "status": "Active",  "assignedTo": “f3ba8f83-5201-4ed9-8022-24ee6431aeda”  } |
| 400 | {"error":"Please specify valid engineerId"} |
| 400 | {"error":"Please specify valid projectId."} |

## Response

| **Status** | **Response** |
| --- | --- |
| 200 | **{**  **"success": true,**  **"message": "Time sheet has been created successfully",**  **"data": {**  "engineerId": “b6dff8a7-7a2f-453d-b225-8e7d2b0b9017”,  "teamLeadId": “50e38ccc-fb37-4fda-b4ca-18a555dd91b1”,  "name": "Wazo\_timesheet",  "projectId": "f7a50527-1c40-47ab-b56b-6b30af826961",  "status": "Active",  "assignedTo": “f3ba8f83-5201-4ed9-8022-24ee6431aeda”  **"createdAt": "2023-02-09T06:53:10.610Z",**  **"updatedAt": "2023-02-09T06:53:10.610Z",**  **"id": 6,**  **"taskDetails": [**  {  "date": "2022-12-02",  "hourMinute": "2:30",  "task": "Created document for Test API/s and shared with Kartik",  "notes": "test notes",  "status": "completed"  },  {  "date": "2022-12-02",  "hourMinute": "2:30",  "task": "Created Postman collection and attached on ticket",  "notes": "test notes",  "status": "completed"  }  **]**  **}**  **}** |
|  |  |

## 

## 6. Delete Timesheet

Delete the particular individual timesheet  
Timesheets will be soft deleted from the system. All the details will be archived

## URL

| **Method** | **URL** |
| --- | --- |
| **DELETE** | api/timesheet/:id |

| **Type** | **Params** | **Values** |
| --- | --- | --- |
| HEAD  Path Variable  POST | Authorization  id  version | Bearer Token  \3.0.0 |

## Request

| **Status** | **Response** |
| --- | --- |
|  |  |

## Response

| **Status** | **Response** |
| --- | --- |
| 204 | **{**  **"success": true,**  **"message": "Time sheet has been deleted successfully",**  **"data": {**}  **}** |
|  |  |

## 

## 7. Summary

Glossary

## Conventions

* **Client** - Client application.
* **Status** - HTTP status code of response.
* All the possible responses are listed under ‘Responses’ for each method. Only one of them is issued per request server.
* All responses are in JSON format.
* All request parameters are mandatory unless explicitly marked as [optional]
* The type of values accepted for a *request* parameter are shown the the values column like this [**10**|<any number>] .The | symbol means *OR*. If the parameter is [optional], the default value is shown in blue bold text, as **10** is written in [**10**|<any number>].

## Status Codes

All status codes are standard HTTP status codes. The below ones are used in this API.

2XX - Success of some kind

4XX - Error occurred in client’s part

5XX - Error occurred in server’s part

| **Status Code** | **Description** |
| --- | --- |
| 200 | OK |
| 201 | Created |
| 202 | Accepted (Request accepted, and queued for execution) |
| 400 | Bad request |
| 401 | Authentication failure |
| 403 | Forbidden |
| 404 | Resource not found |
| 405 | Method Not Allowed |
| 409 | Conflict |
| 412 | Precondition Failed |
| 413 | Request Entity Too Large |
| 500 | Internal Server Error |
| 501 | Not Implemented |
| 503 | Service Unavailable |