Kanmind API Endpoint Dokumentation

**POST /api/registration/**

**Description:** Erstellt einen neuen Benutzer.

**Request Body**

{

"fullname": "Example Username",

"email": "example@mail.de",

"password": "examplePassword",

"repeated\_password": "examplePassword"

}

**Success Response**

Erfolgreicher Erstellung gibt dies ein Token sowie die Benutzerinformationen zurück, inklusive die einzigartige Nutzer-ID.

{

"token": "83bf098723b08f7b23429u0fv8274",

"fullname": "Example Username",

"email": "example@mail.de",

"user\_id": 123

}

**Status Codes**

* **201:** Der Benutzer wurde erfolgreich erstellt.
* **400:** Ungültige Anfragedaten.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**No Permissions required**

**POST /api/login/**

**Description:** Authentifiziert einen Benutzer und liefert ein Authentifizierungs-Token zurück, das für weitere API-Anfragen genutzt wird.

**Request Body**

{

"email": "example@mail.de",

"password": "examplePassword"

}

**Success Response**

Erfolgreiche Authentifizierung gibt ein Token sowie Benutzerinformationen zurück.

{

"token": "83bf098723b08f7b23429u0fv8274",

"fullname": "Example Username",

"email": "example@mail.de",

"user\_id": 123

}

**Status Codes**

* **200:** Erfolgreiche Anmeldung.
* **400:** Ungültige Anfragedaten.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**No Permissions required**

**Boards**

Alles zur Bearbeitung, Erstellung und Abruf von Boards

**GET**

**/api/boards/**

**Description:** Ruft eine Liste von Boards ab, die der angemeldete Benutzer entweder erstellt hat oder zu denen er Mitglied ist.

**Success Response**

Die Antwort enthält eine Liste der Boards mit den grundlegenden Informationen: Titel, ID des Eigentümers und die Anzahl an Mitglieder, die generelle Tasks Anzahl und die Anzahl der Tasks in 'to-do' und mit Priorität 'high'.

[

{

"id": 1,

"title": "Projekt X",

"member\_count": 2,

"ticket\_count": 5,

"tasks\_to\_do\_count": 2,

"tasks\_high\_prio\_count": 1,

"owner\_id": 12

},

{

"id": 1,

"title": "Projekt Y",

"member\_count": 12,

"ticket\_count": 43,

"tasks\_to\_do\_count": 12,

"tasks\_high\_prio\_count": 1,

"owner\_id": 3

}

]

**Status Codes**

* **200:** Erfolgreich. Gibt eine Liste der Boards zurück.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Der Benutzer muss Mitglied eines der Boards oder der Eigentümer eines Boards sein, um es anzuzeigen.

**Extra Information:** Die Liste der Boards enthält nur die Boards, zu denen der authentifizierte Benutzer Zugriff hat.

**POST**

**/api/boards/**

**Description:** Erstellt ein neues Board und fügt Mitglieder hinzu. Der Benutzer, der das Board erstellt, wird automatisch als Eigentümer festgelegt.

**Request Body**

{

"title": "Neues Projekt",

"members": [

12,

5,

54

]

}

**Success Response**

Die Antwort enthält das neu erstellte Board mit den grundlegenden Informationen: Titel, ID des Eigentümers und eine Liste der Mitglieder.

{

"id": 18,

"title": "neu",

"member\_count": 3,

"ticket\_count": 0,

"tasks\_to\_do\_count": 0,

"tasks\_high\_prio\_count": 0,

"owner\_id": 2

}

**Status Codes**

* **201:** Das Board wurde erfolgreich erstellt.
* **400:** Ungültige Anfragedaten. Möglicherweise sind einige Benutzer-Email-Adressen ungültig.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Der Benutzer muss angemeldet und berechtigt sein, ein neues Board zu erstellen.

**GET**

**/api/boards/{board\_id}/**

**Description:** Ruft die Informationen eines bestimmten Boards ab, zusammen mit den zugehörigen Tasks.

**URL Parameters**

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| board\_id | - | Die ID des Boards, dessen Informationen und zugewiesene Tasks abgerufen werden sollen. |

**Request Body**

{

}

**Success Response**

Die Antwort enthält die Board-Informationen (Titel, Mitglieder) sowie die Tasks, die dem Board zugewiesen sind.

{

"id": 1,

"title": "Projekt X",

"owner\_id": 12,

"members": [

{

"id": 1,

"email": "max.mustermann@example.com",

"fullname": "Max Mustermann"

},

{

"id": 54,

"email": "max.musterfrau@example.com",

"fullname": "Maxi Musterfrau"

}

],

"tasks": [

{

"id": 5,

"title": "API-Dokumentation schreiben",

"description": "Die API-Dokumentation für das Backend vervollständigen",

"status": "to-do",

"priority": "high",

"assignee": null,

"reviewer": {

"id": 1,

"email": "max.mustermann@example.com",

"fullname": "Max Mustermann"

},

"due\_date": "2025-02-25",

"comments\_count": 0

},

{

"id": 8,

"title": "Code-Review durchführen",

"description": "Den neuen PR für das Feature X überprüfen",

"status": "review",

"priority": "medium",

"assignee": {

"id": 1,

"email": "max.mustermann@example.com",

"fullname": "Max Mustermann"

},

"reviewer": null,

"due\_date": "2025-02-27",

"comments\_count": 0

}

]

}

**Status Codes**

* **200:** Erfolgreich. Gibt das Board mit den zugehörigen Tasks zurück.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein, um auf die Ressource zuzugreifen.
* **403:** Verboten. Der Benutzer muss entweder Mitglied des Boards oder der Eigentümer des Boards sein.
* **404:** Board nicht gefunden. Die angegebene Board-ID existiert nicht.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Der Benutzer muss entweder Mitglied des Boards oder der Eigentümer des Boards sein, um die Informationen und Tasks abzurufen.

**Extra Information:** Die Antwort enthält das Board mit allen Mitgliedern sowie die zugehörigen Tasks.

**PATCH**

**/api/boards/{board\_id}/**

**Description:** Aktualisiert die Mitglieder eines bestehenden Boards. Mitglieder können hinzugefügt oder entfernt werden. Der Benutzer, der die Anfrage stellt, muss entweder der Eigentümer des Boards oder ein Mitglied des Boards sein. Dieser Endpoint ist nicht zum ändern der Tasks gedacht!

**URL Parameters**

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| board\_id | - | Die ID des Boards, dessen Mitglieder aktualisiert werden sollen. |

**Request Body**

{

"title": "Changed title",

"members": [

1,

54

]

}

**Success Response**

Die Antwort enthält das aktualisierte Board mit den neuen Mitgliedern und entfernt nicht benannte Mitglieder.

{

"id": 3,

"title": "Changed title",

"owner\_data": {

"id": 1,

"email": "max.mustermann@example.com",

"fullname": "Max Mustermann"

},

"members\_data": [

{

"id": 1,

"email": "max.mustermann@example.com",

"fullname": "Max Mustermann"

},

{

"id": 54,

"email": "max.musterfrau@example.com",

"fullname": "Maxi Musterfrau"

}

]

}

**Status Codes**

* **200:** Das Board wurde erfolgreich aktualisiert. Mitglieder wurden hinzugefügt und/oder entfernt.
* **400:** Ungültige Anfragedaten. Möglicherweise sind einige Benutzer ungültig.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein, um auf die Ressource zuzugreifen.
* **403:** Verboten. Der Benutzer muss entweder der Eigentümer oder ein Mitglied des Boards sein.
* **404:** Board nicht gefunden. Die angegebene Board-ID existiert nicht.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Der Benutzer muss entweder der Eigentümer oder ein Mitglied des Boards sein, um Mitglieder hinzuzufügen oder zu entfernen.

**DELETE**

**/api/boards/{board\_id}/**

**Description:** Löscht ein Board. Nur der Eigentümer (Owner) des Boards hat die Berechtigung, das Board zu löschen.

**URL Parameters**

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| board\_id | - | Die ID des Boards, das gelöscht werden soll. |

**Request Body**

{

}

**Success Response**

Die Antwort bestätigt, dass das Board erfolgreich gelöscht wurde.

null

**Status Codes**

* **204:** Das Board wurde erfolgreich gelöscht.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein, um auf die Ressource zuzugreifen.
* **403:** Verboten. Der Benutzer muss der Eigentümer des Boards sein, um es zu löschen.
* **404:** Board nicht gefunden. Die angegebene Board-ID existiert nicht.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Der Benutzer muss der Eigentümer des Boards sein, um es zu löschen.

**Extra Information:** Wenn der Benutzer nicht der Eigentümer des Boards ist, wird die Anfrage mit einem `401 Unauthorized`-Fehler abgelehnt. Das Löschen eines Boards entfernt alle zugehörigen Tasks und Kommentare.

**GET**

**/api/email-check/**

**Description:** Prüft, ob eine bestimmte E-Mail-Adresse bereits einem registrierten Benutzer zugeordnet ist.

**Query Parameters**

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| email | email | Die E-Mail-Adresse, die überprüft werden soll. |

**Request Body**

{

}

**Success Response**

Die Antwort gibt den User zurück falls dieser existiert.

{

"id": 1,

"email": "max.mustermann@example.com",

"fullname": "Max Mustermann"

}

**Status Codes**

* **200:** Erfolgreich. Gibt zurück, ob die E-Mail existiert.
* **400:** Ungültige Anfrage. Die E-Mail-Adresse fehlt oder hat ein falsches Format.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein.
* **404:** Email nicht gefunden. Die Email exestiert nicht.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Der Benutzer muss angemeldet sein

**Tasks**

Alles zur Bearbeitung, Erstellung und Abruf von Tasks

**Description:** Ruft alle Tasks ab, die dem aktuell authentifizierten Benutzer entweder als Bearbeiter (`assignee`) zugewiesen sind. Der Benutzer muss eingeloggt sein, um auf diese Tasks zuzugreifen.

**Request Body**

{

}

**Success Response**

Die Antwort enthält eine Liste der Tasks, die entweder dem aktuell authentifizierten Benutzer zugewiesen wurden. Jede Task enthält grundlegende Informationen wie Titel, Status, Priorität und Fälligkeitsdatum.

[

{

"id": 1,

"board": 1,

"title": "Task 1",

"description": "Beschreibung der Task 1",

"status": "to-do",

"priority": "high",

"assignee": {

"id": 13,

"email": "marie.musterfraun@example.com",

"fullname": "Marie Musterfrau"

},

"reviewer": {

"id": 1,

"email": "max.mustermann@example.com",

"fullname": "Max Mustermann"

},

"due\_date": "2025-02-25",

"comments\_count": 0

},

{

"id": 2,

"board": 12,

"title": "Task 2",

"description": "Beschreibung der Task 2",

"status": "in-progress",

"priority": "medium",

"assignee": {

"id": 13,

"email": "marie.musterfraun@example.com",

"fullname": "Marie Musterfrau"

},

"reviewer": null,

"due\_date": "2025-02-20",

"comments\_count": 0

}

]

**Status Codes**

* **200:** Erfolgreich. Gibt eine Liste der Tasks zurück, die dem aktuell authentifizierten Benutzer entweder als Bearbeiter oder als Prüfer zugewiesen sind.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein, um auf diese Tasks zugreifen zu können.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Der Benutzer muss eingeloggt und authentifiziert sein, um auf die Tasks zuzugreifen, die ihm als Bearbeiter (`assignee`) zugewiesen sind.

**GET**

**/api/tasks/reviewing/**

**Description:** Ruft alle Tasks ab, bei denen der aktuell authentifizierte Benutzer als Prüfer (`reviewer`) eingetragen ist. Der Benutzer muss eingeloggt sein, um auf diese Tasks zuzugreifen.

**Request Body**

{

}

**Success Response**

Die Antwort enthält eine Liste der Tasks, die dem authentifizierten Benutzer zur Überprüfung zugewiesen wurden. Jede Task enthält grundlegende Informationen wie Titel, Status, Priorität und Fälligkeitsdatum.

[

{

"id": 1,

"board": 1,

"title": "Task 1",

"description": "Beschreibung der Task 1",

"status": "to-do",

"priority": "high",

"assignee": null,

"reviewer": {

"id": 1,

"email": "max.mustermann@example.com",

"fullname": "Max Mustermann"

},

"due\_date": "2025-02-25",

"comments\_count": 0

},

{

"id": 2,

"board": 12,

"title": "Task 2",

"description": "Beschreibung der Task 2",

"status": "in-progress",

"priority": "medium",

"assignee": {

"id": 13,

"email": "marie.musterfraun@example.com",

"fullname": "Marie Musterfrau"

},

"reviewer": {

"id": 1,

"email": "max.mustermann@example.com",

"fullname": "Max Mustermann"

},

"due\_date": "2025-02-20",

"comments\_count": 0

}

]

**Status Codes**

* **200:** Erfolgreich. Gibt eine Liste der Tasks zurück, bei denen der Benutzer als Prüfer (`reviewer`) eingetragen ist.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein, um auf diese Tasks zugreifen zu können.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Der Benutzer muss eingeloggt und authentifiziert sein, um auf die Tasks zuzugreifen, die ihm als Prüfer (`reviewer`) zugewiesen sind.

**POST**

**/api/tasks/**

**Description:** Erstellt eine neue Task innerhalb eines Boards. Der Benutzer muss einen der folgenden Werte für den Status nutzen: `to-do`, `in-progress`, `review` oder `done` und einen der folgenden Werte für die Priority: `low`, `medium` oder `high`.

**Request Body**

{

"board": 12,

"title": "Code-Review durchführen",

"description": "Den neuen PR für das Feature X überprüfen",

"status": "review",

"priority": "medium",

"assignee\_id": 13,

"reviewer\_id": 1,

"due\_date": "2025-02-27"

}

**Success Response**

Die Antwort enthält die erstellte Task mit allen zugehörigen Informationen.

{

"id": 10,

"board": 12,

"title": "Code-Review durchführen",

"description": "Den neuen PR für das Feature X überprüfen",

"status": "review",

"priority": "medium",

"assignee": {

"id": 13,

"email": "marie.musterfraun@example.com",

"fullname": "Marie Musterfrau"

},

"reviewer": {

"id": 1,

"email": "max.mustermann@example.com",

"fullname": "Max Mustermann"

},

"due\_date": "2025-02-27",

"comments\_count": 0

}

**Status Codes**

* **201:** Die Task wurde erfolgreich erstellt.
* **400:** Ungültige Anfragedaten. Möglicherweise fehlen erforderliche Felder oder enthalten ungültige Werte.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein.
* **403:** Verboten. Der Benutzer muss Mitglied des Boards sein, um eine Task zu erstellen.
* **404:** Board nicht gefunden. Die angegebene Board-ID existiert nicht.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Der Benutzer muss Mitglied des Boards sein, um eine Task zu erstellen.

**Extra Information:** Sowohl `assignee` als auch `reviewer` müssen Mitglieder des Boards sein. Falls kein `assignee` oder `reviewer` angegeben wird, bleibt das Feld leer.

**PATCH**

**/api/tasks/{task\_id}/**

**Description:** Aktualisiert eine bestehende Task. Nur Mitglieder des Boards, zu dem die Task gehört, können sie bearbeiten.

**URL Parameters**

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| task\_id | - | Die ID der zu aktualisierenden Task. |

**Request Body**

{

"title": "Code-Review abschließen",

"description": "Den PR fertig prüfen und Feedback geben",

"status": "done",

"priority": "high",

"assignee\_id": 13,

"reviewer\_id": 1,

"due\_date": "2025-02-28"

}

**Success Response**

Die Antwort enthält die aktualisierte Task mit allen geänderten Werten.

{

"id": 10,

"title": "Code-Review abschließen",

"description": "Den PR fertig prüfen und Feedback geben",

"status": "done",

"priority": "high",

"assignee": {

"id": 13,

"email": "marie.musterfraun@example.com",

"fullname": "Marie Musterfrau"

},

"reviewer": {

"id": 1,

"email": "max.mustermann@example.com",

"fullname": "Max Mustermann"

},

"due\_date": "2025-02-28"

}

**Status Codes**

* **200:** Die Task wurde erfolgreich aktualisiert.
* **400:** Ungültige Anfragedaten. Möglicherweise sind einige Werte ungültig oder nicht erlaubt.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein.
* **403:** Verboten. Der Benutzer muss Mitglied des Boards sein, zu dem die Task gehört.
* **404:** Task nicht gefunden. Die angegebene Task-ID existiert nicht.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Der Benutzer muss Mitglied des Boards sein, um eine Task zu aktualisieren. Das ändern der Board-Id(board) ist nicht erlaubt!

**Extra Information:** Felder, die nicht aktualisiert werden sollen, können weggelassen werden. `assignee` und `reviewer` müssen weiterhin Mitglieder des Boards sein.

**DELETE**

**/api/tasks/{task\_id}/**

**Description:** Löscht eine bestehende Task. Nur der Ersteller der Task oder der Eigentümer des Boards kann die Task löschen.

**URL Parameters**

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| task\_id | - | Die ID der zu löschenden Task. |

**Request Body**

{

}

**Success Response**

Wenn die Task erfolgreich gelöscht wurde, wird eine Bestätigung ohne Inhalt zurückgegeben.

null

**Status Codes**

* **204:** Die Task wurde erfolgreich gelöscht.
* **400:** Ungültige Anfragedaten. Die übermittelte Task-ID ist fehlerhaft.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein.
* **403:** Verboten. Nur der Ersteller der Task oder der Board-Eigentümer kann die Task löschen.
* **404:** Task nicht gefunden. Die angegebene Task-ID existiert nicht.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Nur der Ersteller der Task oder der Eigentümer des Boards kann eine Task löschen.

**Extra Information:** Die Löschung ist dauerhaft und kann nicht rückgängig gemacht werden.

**GET**

**/api/tasks/{task\_id}/comments/**

**Description:** Ruft alle Kommentare ab, die einer bestimmten Task zugeordnet sind.

**URL Parameters**

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| task\_id | - | Die ID der Task, zu der die Kommentare abgerufen werden sollen. |

**Request Body**

{

}

**Success Response**

Die Antwort enthält eine Liste aller Kommentare zur angegebenen Task. Jeder Kommentar enthält das Erstellungsdatum, den vollständigen Namen des Autors und den Inhalt.

[

{

"id": 1,

"created\_at": "2025-02-20T14:30:00Z",

"author": "Max Mustermann",

"content": "Das ist ein Kommentar zur Task."

},

{

"id": 2,

"created\_at": "2025-02-21T09:15:00Z",

"author": "Erika Musterfrau",

"content": "Ein weiterer Kommentar zur Diskussion."

}

]

**Status Codes**

* **200:** Erfolgreich. Gibt eine Liste der Kommentare zurück.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein.
* **403:** Verboten. Der Benutzer muss Mitglied des Boards sein, zu dem die Task gehört.
* **404:** Task nicht gefunden. Die angegebene Task-ID existiert nicht.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Der Benutzer muss Mitglied des Boards sein, zu dem die Task gehört.

**Extra Information:** Die Kommentare sind chronologisch nach Erstellungsdatum sortiert.

**POST**

**/api/tasks/{task\_id}/comments/**

**Description:** Erstellt einen neuen Kommentar zu einer bestimmten Task. Der Autor wird automatisch anhand der Authentifizierung bestimmt.

**URL Parameters**

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| task\_id | - | Die ID der Task, zu der der Kommentar hinzugefügt werden soll. |

**Request Body**

{

"content": "Das ist ein neuer Kommentar zur Task."

}

**Success Response**

Die Antwort enthält die erstellte Kommentarinstanz mit ID, Erstellungsdatum, vollständigem Namen des Autors und dem Inhalt.

{

"id": 15,

"created\_at": "2025-02-20T15:00:00Z",

"author": "Max Mustermann",

"content": "Das ist ein neuer Kommentar zur Task."

}

**Status Codes**

* **201:** Der Kommentar wurde erfolgreich erstellt.
* **400:** Ungültige Anfragedaten. Möglicherweise ist der `content`-Wert leer.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein.
* **403:** Verboten. Der Benutzer muss Mitglied des Boards sein, zu dem die Task gehört.
* **404:** Task nicht gefunden. Die angegebene Task-ID existiert nicht.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Der Benutzer muss Mitglied des Boards sein, zu dem die Task gehört.

**Extra Information:** Der Autor des Kommentars wird aus der Authentifizierung des aktuellen Benutzers bestimmt.

**DELETE**

**/api/tasks/{task\_id}/comments/{comment\_id}/**

**Description:** Löscht einen Kommentar einer bestimmten Task. Nur der Ersteller des Kommentars kann ihn löschen.

**URL Parameters**

| **Name** | **Type** | **Description** |
| --- | --- | --- |
| task\_id | - | Die ID der Task, zu der der Kommentar gehört. |
| comment\_id | - | Die ID des zu löschenden Kommentars. |

**Request Body**

{

}

**Success Response**

Bei erfolgreicher Löschung wird eine leere Antwort mit Statuscode `204` zurückgegeben.

null

**Status Codes**

* **204:** Der Kommentar wurde erfolgreich gelöscht.
* **400:** Ungültige Anfragedaten.
* **401:** Nicht autorisiert. Der Benutzer muss eingeloggt sein.
* **403:** Verboten. Nur der Ersteller des Kommentars darf ihn löschen.
* **404:** Kommentar oder Task nicht gefunden.
* **500:** Interner Serverfehler.

**Rate Limits**

* No limit

**Permissions required:** Nur der Benutzer, der den Kommentar erstellt hat, darf ihn löschen.

**Extra Information:** Falls der Kommentar oder die Task nicht existiert, wird ein `404`-Fehler zurückgegeben.