

Testdurchführung

Delegator

Klassifizierung intern
Status in Arbeit
Programmname Delegator
Projektnummer 1
Projektleiter Tabinas Kenan
Version 0.1
Datum 26. März 2025
Auftraggeber Tabinas Kenan
Autor/Autoren Tabinas Kenan
Verteiler

Änderungsverzeichnis

Version	Datum	Änderung	Autor
0.1	03.06.2025	Usertest	TAK
1	06.06.2025	Pentest	TAK

Tabelle 1: Änderungsverzeichnis

1 Abnahme von Muss Kriterien

Test Bedingungen:

- Gerät: OnePlus EB2103.
- Testerin: Noëmi Tabinas, wusste nicht wie die App Funktioniert und erhielt nur das Erwartete Ergebnisse und musste die Tests Schritte absolvieren.
- Testvoraussetzungen: Diese wurde im Vorhinein sichergestellt.

1.1 Testfall

ID / Bezeichnung	T-01	User Funktionalität
Beschreibung	Erstellen eines Projektes	
Testvoraussetzung	Eingelogged, Organisation beigetreten	
Testschritte	<ul style="list-style-type: none">• Auf neues Projekt drücken	
Erwartetes Ergebnis	Es erscheint ein neues Projekt	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	Beim ersten Mal, habe ich nicht beachtet, dass es wir keine Org gegründet haben.
*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel	

1.2 Testfall

ID / Bezeichnung	T-05	Aufgaben
Beschreibung	Aufgabe erstellt	
Testvoraussetzung	Eingelogged, Projekt ist erstellt	
Testschritte	<ul style="list-style-type: none"> • Klickt auf neue Aufgabe • Gibt geforderten Informationen ein 	
Erwartetes Ergebnis	Es ermöglicht dem User eine Aufgabe zu erstellen	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	Im Hauptmenu Verlinkung zum Task funktioniert nicht. Gibt overflow
*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel	

1.3 Testfall

ID / Bezeichnung	T-06	Aufgaben
Beschreibung	Aufgabe einer anderen Person zu weisen	
Testvoraussetzung	Eingeloggnet, Projekt ist erstellt und es ist mehrere Personen in diesem Projekt	
Testschritte	<ul style="list-style-type: none"> • Aufgabe erstellen • Auf Zuweisung eine andere Person auswählen. • Speichern 	
Erwartetes Ergebnis	Es erscheint unter Aufgaben der Name der Ausgewählten Person bei der Aufgabe	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

1.4 Testfall

ID / Bezeichnung	T-07	Aufgaben
Beschreibung	Aufgabe Terminieren	
Testvoraussetzung	Eingelogged, Projekt ist erstellt	
Testschritte	<ul style="list-style-type: none"> • Aufgabe erstellen • Bei Deadline Datum auswählen • Speichern. 	
Erwartetes Ergebnis	Es erscheint bei der Aufgabe ein Datum	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

1.5 Testfall

ID / Bezeichnung	T-08	Aufgaben
Beschreibung	Kalender synchronisiert	
Testvoraussetzung	Eingelogged, Projekt ist erstellt und Kalender ist Synchronisiert	
Testschritte	<ul style="list-style-type: none"> • Aufgabe auf einen anderen Tag setzen. • Kalender App Synchronisieren 	
Erwartetes Ergebnis	Die Aufgabe erscheint an einem anderen Tag	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	Code ist nicht verfügbar aus der App heraus
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

1.6 Testfall

ID / Bezeichnung	T-10	Aufgaben
Beschreibung	Kalender zugriff	
Testvoraussetzung	Eingeloggged	
Testschritte	<ul style="list-style-type: none"> Man kann der Link von eigenen Kalender kopieren 	
Erwartetes Ergebnis	Link kann genutzt werden für die Kalender sychnronisa-tion	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Man-gel</i>	

1.7 Testfall

ID / Bezeichnung	T-11	Zugriff
Beschreibung	Projekt zugriff	
Testvoraussetzung	Eingeloggged, User hat Zugriff auf existierende Projekte	
Testschritte	<ul style="list-style-type: none"> • Projekte Liste anzeigen 	
Erwartetes Ergebnis	Projekte werden angezeigt	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

1.8 Testfall

ID / Bezeichnung	T-12	Zugriff
Beschreibung	Aufgaben anzeigen	
Testvoraussetzung	Eingelogged, User ist zugewiesen bei Projekten, Es existieren Aufgaben	
Testschritte	<ul style="list-style-type: none"> Aufgaben liste anzeigen 	
Erwartetes Ergebnis	Aufgaben werden angezeigt	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

1.9 Testfall

ID / Bezeichnung	T-25	Bug Report
Beschreibung	Jeder User ohne Anmeldung ein Bug Report erstellen	
Testvoraussetzung	-	
Testschritte	<ul style="list-style-type: none"> • Vor dem Login einen Bug Report ausfüllen • Titel, Inhalt und E-Mail eingeben • Absenden drücken 	
Erwartetes Ergebnis	Abstimmung verschwindet wieder.	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Tabinas
Mängelklasse*	0
Mangelbeschreibung	
Bemerkungen	keine
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

1.10 Testfall

ID / Bezeichnung	T-26	Bug Report
Beschreibung	Nach dem Bug Report gibt es eine Meldung an den User	
Testvoraussetzung	-	
Testschritte	<ul style="list-style-type: none"> • Vor dem Login einen Bug Report ausfüllen • Titel, Inhalt und E-Mail eingeben • Absenden drücken 	
Erwartetes Ergebnis	User erhält E-Mail auf die angegebene E-Mail	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	
Tester	
Mängelklasse*	4
Mangelbeschreibung	Dieses Feature wurde leider nichts umgesetzt.
Bemerkungen	keine
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

2 Abnahme von Kann Kriterien

2.1 Testfall

ID / Bezeichnung	T-19	Chat
Beschreibung	Projekte sind Kommentierbar	
Testvoraussetzung	Eingelogged, User ist zugewiesen bei Projekten	
Testschritte	<ul style="list-style-type: none"> • Bei einem Projekt • Auf Chats drücken • Nachricht absenden 	
Erwartetes Ergebnis	Nachrichten stehen im Chat	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

2.2 Testfall

ID / Bezeichnung	T-20	Chat
Beschreibung	Channels sind Kommentierbar	
Testvoraussetzung	Eingelogged, User ist zugewiesen bei Projekten	
Testschritte	<ul style="list-style-type: none"> • Bei Chats hin gehen • Einen Channel auswählen • Nachricht absenden 	
Erwartetes Ergebnis	Nachrichten stehen im Chat	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	03.06.2025
Tester	Noëmi Louise Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

3 Penetrationstest

Um die Sicherheit der Applikation zu gewährleisten, gehen wir auf den Sicherheit Aspekt ein. Wie aus dem Sicherheitskonzept hervorgeht, sind Schwache Endpoints die grösste Schwachstelle.

3.1 Testfall

ID / Bezeichnung	PT001	Token
Beschreibung	Testen, ob geschützte Endpunkte ohne Token erreichbar sind	
Testvoraussetzung	Keine Anmeldung	
Testschritte	GET /projects/ ohne Authorization Header aufrufen GET /calendars/ ohne Token versuchen POST /messages/ ohne Authentifizierung testen	
Erwartetes Ergebnis	401 Unauthorized für alle geschützten Endpunkte	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine
*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel	

3.2 Testfall

ID / Bezeichnung	PT002	Token
Beschreibung	Versuchen, JWT Tokens zu manipulieren oder zu fälschen	
Testvoraussetzung	Keine Anmeldung	
Testschritte	JWT Token dekodieren und Payload ändern (z.B. user_id) Token mit ungültiger Signatur verwenden Abgelaufenes Token testen Token mit geändertem Header verwenden	
Erwartetes Ergebnis	Alle manipulierten Tokens werden abgelehnt	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	keine
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

3.3 Testfall

ID / Bezeichnung	PT003	Token
Beschreibung	Testen, ob User auf fremde Daten zugreifen können	
Testvoraussetzung	Zwei verschiedene User-Accounts	
Testschritte	Mit User A einloggen Versuchen, Projekte von User B abzurufen (GET /projects/{userproject_id}/)	
Erwartetes Ergebnis	403 Forbidden bei Zugriff auf fremde Daten	

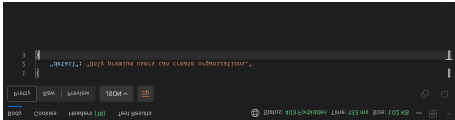
Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	Es kommt 404 not Found. Was mir sogar besser gefällt als 405.
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

3.4 Testfall

ID / Bezeichnung	PT004	Premium
Beschreibung	Testen, ob normale User Admin-Funktionen ausführen können	
Testvoraussetzung	Zwei verschiedene User-Accounts	
Testschritte	Versuchen, neue Organisation zu erstellen ohne Premium Versuchen, andere User aus Organisation zu entfernen Versuchen, auf /history/ zuzugreifen ohne Staff-Rechte Songs erstellen/bearbeiten ohne entsprechende Rolle	
Erwartetes Ergebnis	403 Forbidden bei Zugriff auf fremde Daten	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

3.5 Testfall

ID / Bezeichnung	PT005	SQL Injection
Beschreibung	SQL Injection in verschiedenen Endpunkten testen	
Testvoraussetzung	Gültiger API-Zugang	
Testschritte	SQL Injection in Suchfeldern: GET /songs/?search='; DROP TABLE-- SQL Injection in Filtern: GET /users/?username=admin' OR '1'='1' SQL Injection in POST/PUT Requests in JSON-Feldern SQL Injection in URL-Parametern	
Erwartetes Ergebnis	Keine SQL-Statements werden ausgeführt, Eingaben werden escaped	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	

**Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel*

3.6 Testfall

ID / Bezeichnung	PT006	Grenzwerte und Datentypen
Beschreibung	SQL Injection in verschiedenen Endpunkten testen	
Testvoraussetzung	Gültiger API-Zugang	
Testschritte	Extrem lange Strings in Text-Feldern (>10000 Zeichen) Negative Zahlen wo positive erwartet werden Float-Werte in Integer-Feldern Leere/null Werte in Required-Feldern Unicode/Emoji in verschiedenen Feldern	
Erwartetes Ergebnis	Angemessene Validierungsfehler, keine 500-Errors	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	Username max 150 Chars Floating Point numbers werden Ignoriert Json Parse Error bei Emojis
*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel	

3.7 Testfall

ID / Bezeichnung	PT007	iCal
Beschreibung	Sicherheit der iCal-Token testen	
Testvoraussetzung	Zugriff auf Calendar Subscription	
Testschritte	<p>Versuchen, fremde Calendar-Tokens zu erraten/bruteforcen</p> <p>Token-Enumeration: /ical/calendar/AAAA.../, /ical/calendar/AAAB.../</p> <p>Prüfen, ob Token in Logs oder URLs geleakt werden</p> <p>Testen, ob revoked Token noch funktionieren</p>	
Erwartetes Ergebnis	Angemessene Validierungsfehler, keine 500-Errors	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	2
Mangelbeschreibung	Nicht Ideal. Da das Token in der URL auftaucht. Eventuell wäre Token via Body besser. Aber das liegt eventuell an der Technologie. Es gäbe noch eine Art login.
Bemerkungen	keine
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

3.8 Testfall

ID / Bezeichnung	PT008	Organisation Invitation Security
Beschreibung	Sicherheit des Einladungssystems testen	
Testvoraussetzung	Organisationsadmin-Rechte	
Testschritte	Versuchen, Einladungen für andere Invite-Codes zu erstellen Mehrfache Verwendung von Einladungs-Tokens Einladungen für bereits existierende Mitglieder Abgelaufene Einladungen verwenden	
Erwartetes Ergebnis	Robuste Validierung aller Einladungsschritte	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	
Bemerkungen	
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

3.9 Testfall

ID / Bezeichnung	PT009	Project Access Control
Beschreibung	Zugriffskontrolle bei Projekten testen	
Testvoraussetzung	Mehrere User mit verschiedenen Projekt-Zugehörigkeiten	
Testschritte	User zu Projekt hinzufügen ohne Berechtigung Auf Projekt-Chat zugreifen ohne Projekt-Mitgliedschaft Tasks in fremden Projekten erstellen/bearbeiten Project-Level vs Organization-Level Berechtigungen testen	
Erwartetes Ergebnis	Strikte Trennung der Projekt-Zugriffe	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	

**Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel*

3.10 Testfall

ID / Bezeichnung	PT010	HTTP Method Testing
Beschreibung	Unerlaubte HTTP-Methoden testen	
Testvoraussetzung	Gültiger API-Zugang	
Testschritte	POST bei /chat/ DELETE an ReadOnly-Endpunkten /history/ Method Override Headers testen	
Erwartetes Ergebnis	Nur erlaubte HTTP-Methoden werden akzeptiert	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

3.11 Testfall

ID / Bezeichnung	PT011	Content-Type Validation
Beschreibung	Verschiedene Content-Types und Payload-Formate testen	
Testvoraussetzung	Gültiger API-Zugang	
Testschritte	XML-Payload an JSON-Endpunkt senden Malformed JSON senden Verschiedene Charset-Encodings testen Content-Type spoofing versuchen	
Erwartetes Ergebnis	Robuste Content-Type Validierung	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	keine
Bemerkungen	JSON Parse error
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

3.12 Testfall

ID / Bezeichnung	PT012	Error Message Information Leakage
Beschreibung	Testen, ob Fehlermeldungen sensitive Informationen preisgeben	
Testvoraussetzung	Verschiedene fehlerhafte Requests	
Testschritte	Nicht-existierende Endpunkte aufrufen Falsche Parameter-Typen senden Server-Errors provozieren Database-Errors auslösen	
Erwartetes Ergebnis	Generische Fehlermeldungen ohne technische Details	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	4
Mangelbeschreibung	Es ist noch Debug Level eingestellt.
Bemerkungen	
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

3.13 Testfall

ID / Bezeichnung	PT013	Session & Token Management
Beschreibung	Token-Lebensdauer und Session-Management testen	
Testvoraussetzung	Gültiger API-Zugang	
Testschritte	Token-Refresh-Mechanismus testen Token-Expiry Handling prüfen Concurrent Sessions testen Token-Invalidierung nach Logout prüfen	
Erwartetes Ergebnis	Sichere Token-Verwaltung	

Testdurchführung und Testergebnis (Mängelklasse)

Testdatum	06.06.2025
Tester	Kenan Tabinas
Mängelklasse*	0
Mangelbeschreibung	
Bemerkungen	
<i>*Mängelklasse: 0 = mängelfrei; 1 = belangloser Mangel; 2 = leichter Mangel; 3 = schwerer Mangel; 4 = kritischer Mangel</i>	

Pentest Fälle sind von Claude.

4 Integrationtest

4.1 Projects

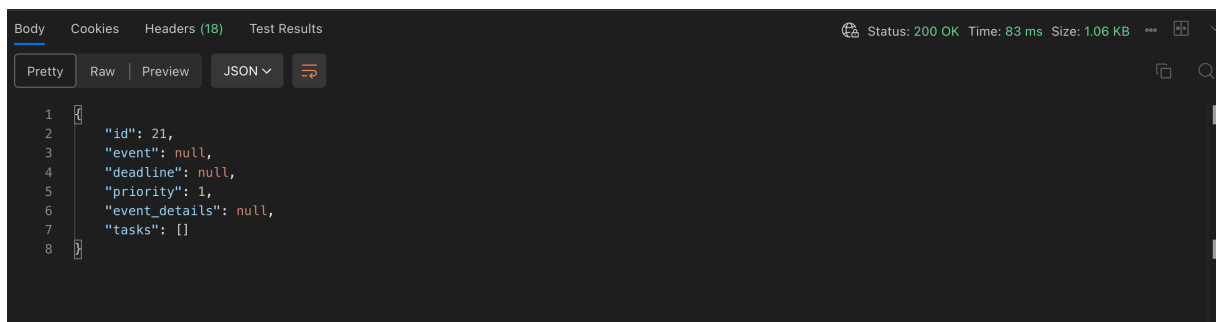
4.1.1 Project Get

```

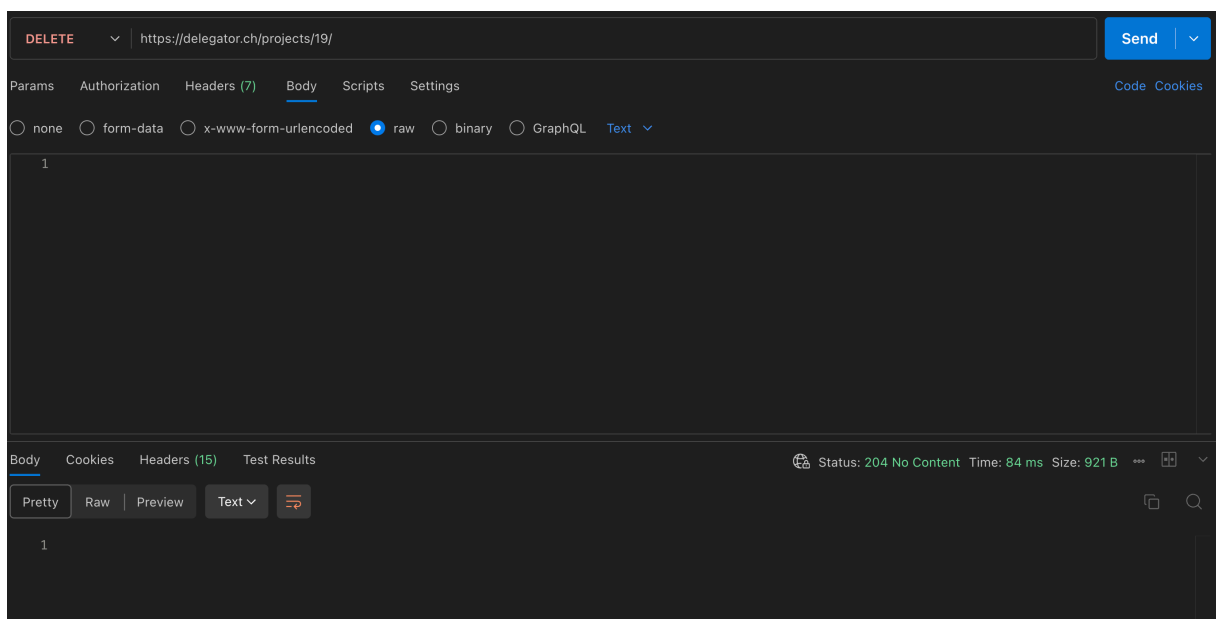
Attempting login for user: test_user_2
POST request to: https://delegator.ch/token/
Request data: {"username":"test_user_2","password":"sml12345"}
Response status: 200
Received login response: (refresh, access)
Tokens received successfully
Extracted user data from token: test_user_2
GET request to: https://delegator.ch/projects/
Response status: 200
Got 7 projects
Project ID: 20, Org ID: 5
Project ID: 15, Org ID: 5
Project ID: 14, Org ID: 5
Project ID: 16, Org ID: 5
Project ID: 13, Org ID: 5
Project ID: 21, Org ID: 5
Project ID: 17, Org ID: 5

```

4.1.2 Project Delete & Get by Id



4.1.3 Project Delete



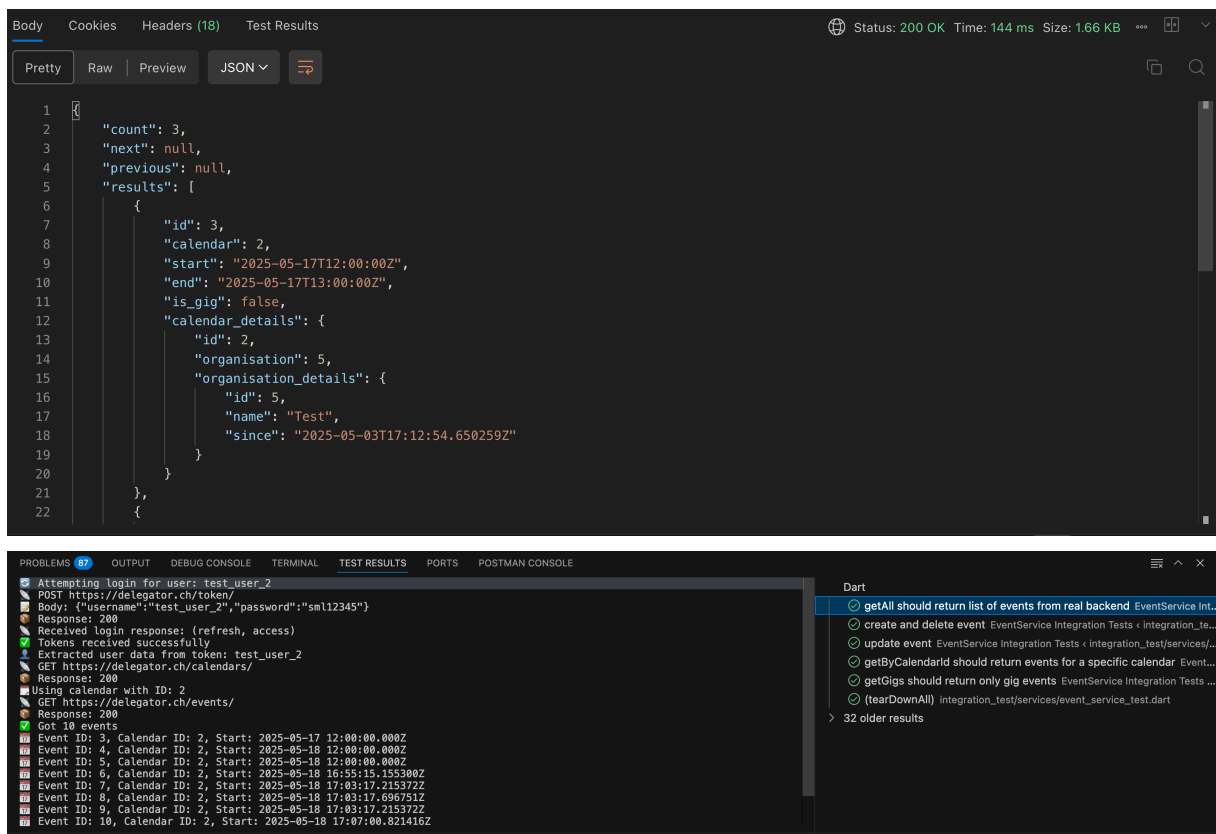
```

Attempting login for user: test_user_2
POST request to: https://delegator.ch/token/
Request data: {"username":"test_user_2","password":"sml12345"}
Response status: 200
Received login response: (refresh, access)
Tokens received successfully
Extracted user data from token: test_user_2
Creating new project for organisation 5
POST request to: https://delegator.ch/projects/
Request data: {"priority":1,"event_details":null,"organisation":5}
Response status: 201
Project created with ID: 23
Deleting project with ID: 23
DELETE request to: https://delegator.ch/projects/23/
Response status: 204
Project deleted successfully
GET request to: https://delegator.ch/projects/23/
Response status: 404
API Error: {detail: Not found.}
API Error (raw): {"detail":"Not found."}
Project no longer exists (expected error): ApiException: [404] {"detail":"Not found."}

```

4.2 Events

4.3 Get Events



The screenshot displays a web browser window showing the response of a GET request to `https://delegator.ch/events/`. The response is a JSON object with the following structure:

```

{
  "count": 3,
  "next": null,
  "previous": null,
  "results": [
    {
      "id": 3,
      "calendar": 2,
      "start": "2025-05-17T12:00:00Z",
      "end": "2025-05-17T13:00:00Z",
      "is_gig": false,
      "calendar_details": {
        "id": 2,
        "organisation": 5,
        "organisation_details": {
          "id": 5,
          "name": "Test",
          "since": "2025-05-03T17:12:54.650259Z"
        }
      }
    }
  ]
}

```

Below the browser window, a terminal window shows the test execution log, and a Dart IDE window shows the test results for 'EventService Integration Tests'.

4.4 Create and Delete

```

Body    Cookies    Headers (17)    Test Results
Pretty  Raw    Preview    JSON
3      "calendar": 2,
4      "start": "2025-05-18T17:03:17.215372Z",
5      "end": "2025-05-18T19:03:17.215372Z",
6      "is_gig": false,
7      "calendar_details": {
8        "id": 2,
9        "organisation": 5,
10       "organisation_details": {
11         "id": 5,
12         "name": "Test",
13         "since": "2025-05-03T17:12:54.650259Z"
14       }
15     }
16

```

```

Body    Cookies    Headers (16)    Test Results
Pretty  Raw    Preview    Text
1

```

```

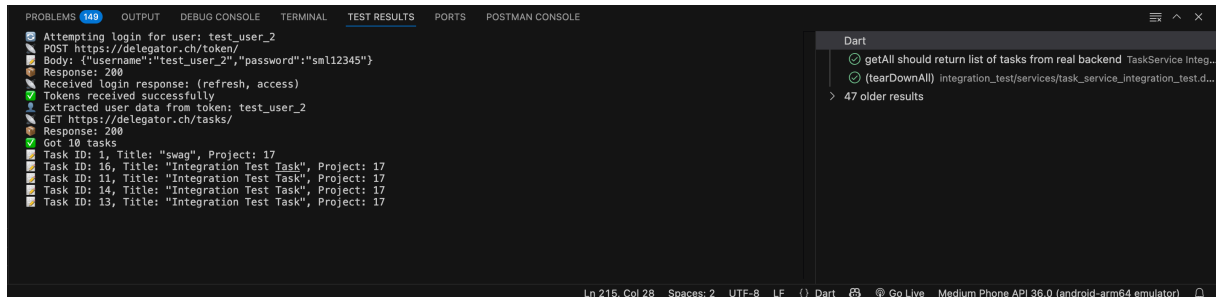
PROBLEMS (17)    OUTPUT    DEBUG CONSOLE    TERMINAL    TEST RESULTS    PORTS    POSTMAN CONSOLE
Attempting login for user: test_user_2
POST https://delegator.ch/token/
Body: {"username":"test_user_2","password":"sm12345"}
Response: 200
Received login response: (refresh, access)
Tokens received successfully
Extracted user data from token: test_user_2
GET https://delegator.ch/calendars/
Response: 200
Using calendar with ID: 2
Creating new event for calendar 2
POST https://delegator.ch/events/
Body: {"title":"Test","calendar":2,"start":"2025-05-18T17:13.319863","end":"2025-05-18T19:17:13.319863","is_gig":false}
Response: 201
Event created with ID: 15
Expected 2025-05-18 17:17:13.319863 and recieved 2025-05-18 17:17:13.319863Z
Deleting event with ID: 15
DELETE https://delegator.ch/events/15/
Response: 204
Event deleted successfully
GET https://delegator.ch/events/15/

Dart
get all should return list of events from real backend EventService Int...
create and delete event EventService Integration Tests < integration_te...
update event EventService Integration Tests < integration_test/services/...
getByCalendarid should return events for a specific calendar Event...
getGigs should return only gig events EventService Integration Tests ...
(tearDownAll) integration_test/services/event_service_test.dart
> 32 older results

```

4.5 Tasks

4.5.1 Get

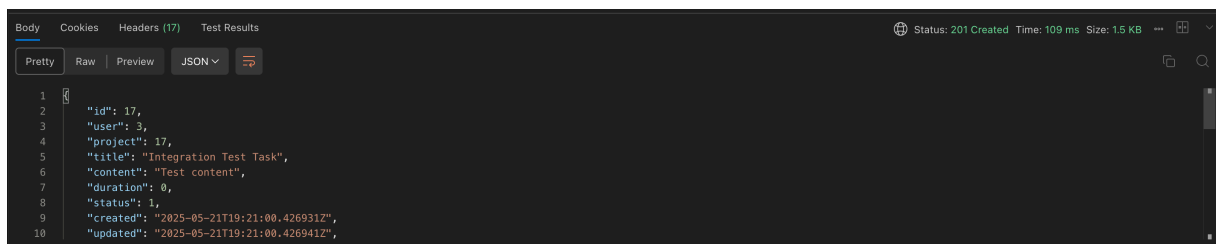


```

PROBLEMS (149) OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS POSTMAN CONSOLE
Attempting login for user: test_user_2
POST https://delegator.ch/token/
Body: {"username":"test_user_2","password":"sml12345"}
Response: 200
Received login response: (refresh, access)
Tokens received successfully
Extracted user data from token: test_user_2
GET https://delegator.ch/tasks/
Response: 200
Got 10 tasks
Task ID: 1, Title: "swag", Project: 17
Task ID: 16, Title: "Integration Test Task", Project: 17
Task ID: 11, Title: "Integration Test Task", Project: 17
Task ID: 14, Title: "Integration Test Task", Project: 17
Task ID: 13, Title: "Integration Test Task", Project: 17

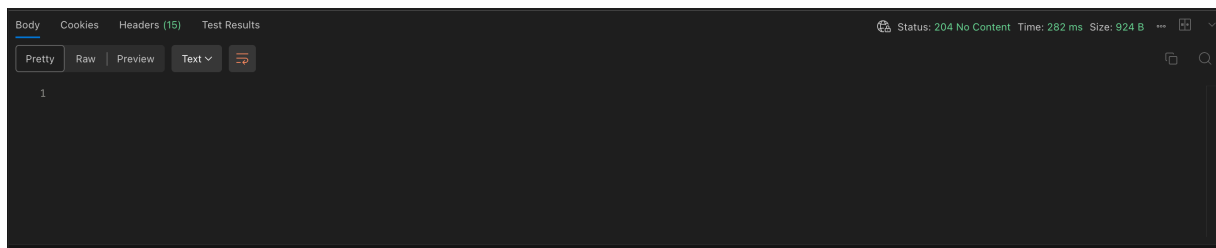
Dart
  ✓ getAll should return list of tasks from real backend TaskService Integ...
  ✓ (tearDownAll) integration_test/services/task_service_integration_test.d...
  > 47 older results
  
```

4.5.2 Create and Delete



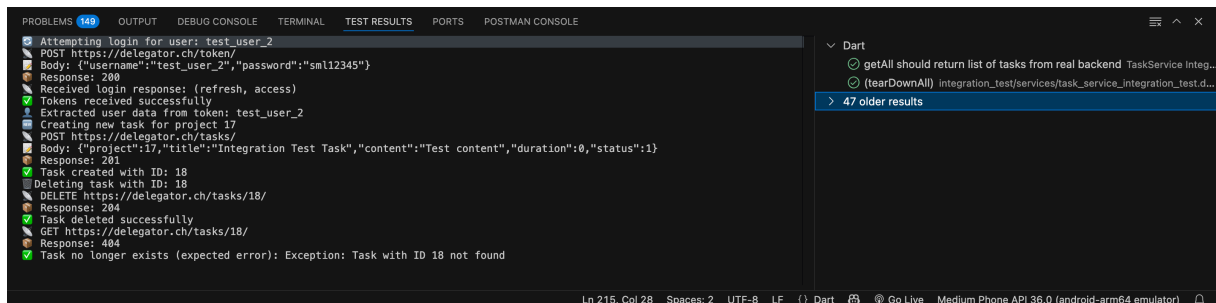
```

Body Cookies Headers (17) Test Results
Pretty Raw Preview JSON
1 2 3 4 5 6 7 8 9 10
{"id": 17,
 "user": 3,
 "project": 17,
 "title": "Integration Test Task",
 "content": "Test content",
 "duration": 0,
 "status": 1,
 "created": "2025-05-21T19:21:00.426931Z",
 "updated": "2025-05-21T19:21:00.426941Z",
}
Status: 201 Created Time: 109 ms Size: 1.5 KB
  
```



```

Body Cookies Headers (15) Test Results
Pretty Raw Preview Text
1
Status: 204 No Content Time: 282 ms Size: 924 B
  
```



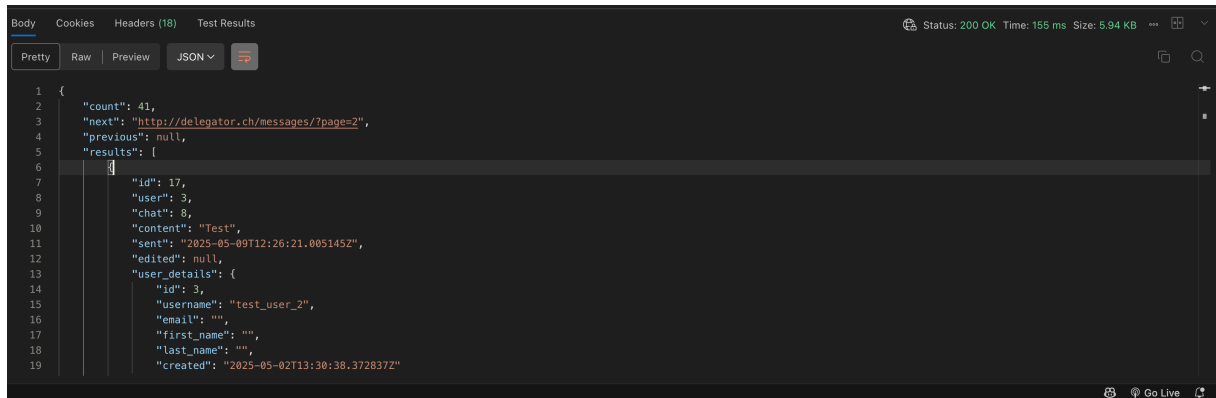
```

PROBLEMS (149) OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS POSTMAN CONSOLE
Attempting login for user: test_user_2
POST https://delegator.ch/token/
Body: {"username":"test_user_2","password":"sml12345"}
Response: 200
Received login response: (refresh, access)
Tokens received successfully
Extracted user data from token: test_user_2
Creating new task for project 17
POST https://delegator.ch/tasks/
Body: {"project":17,"title":"Integration Test Task","content":"Test content","duration":0,"status":1}
Response: 201
Task created with ID: 18
Deleting task with ID: 18
DELETE https://delegator.ch/tasks/18/
Response: 204
Task deleted successfully
GET https://delegator.ch/tasks/18/
Response: 404
Task no longer exists (expected error): Exception: Task with ID 18 not found

Dart
  ✓ getAll should return list of tasks from real backend TaskService Integ...
  ✓ (tearDownAll) integration_test/services/task_service_integration_test.d...
  > 47 older results
  
```

4.6 Messages

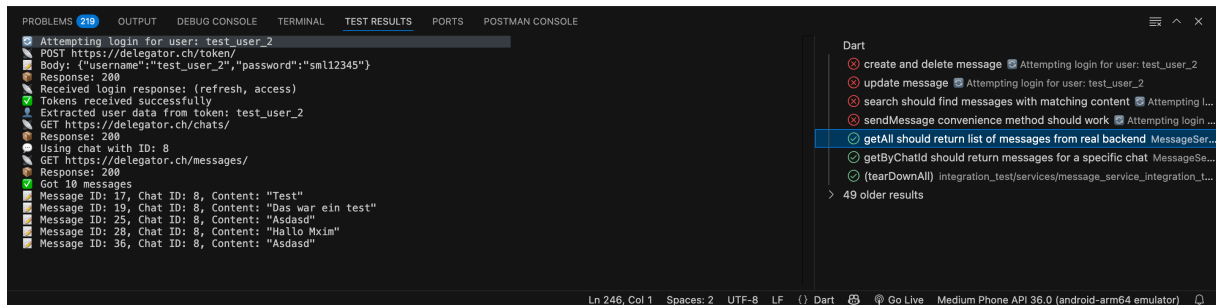
4.6.1 Get



```

1 {
2   "count": 41,
3   "next": "http://delegator.ch/messages/?page=2",
4   "previous": null,
5   "results": [
6     {
7       "id": 17,
8       "user": 3,
9       "chat": 8,
10      "content": "Test",
11      "sent": "2025-05-09T12:26:21.005145Z",
12      "edited": null,
13      "user_details": {
14        "id": 3,
15        "username": "test_user_2",
16        "email": "",
17        "first_name": "",
18        "last_name": "",
19        "created": "2025-05-02T13:30:38.372837Z"
20      }
21    }
22  ]
23 }

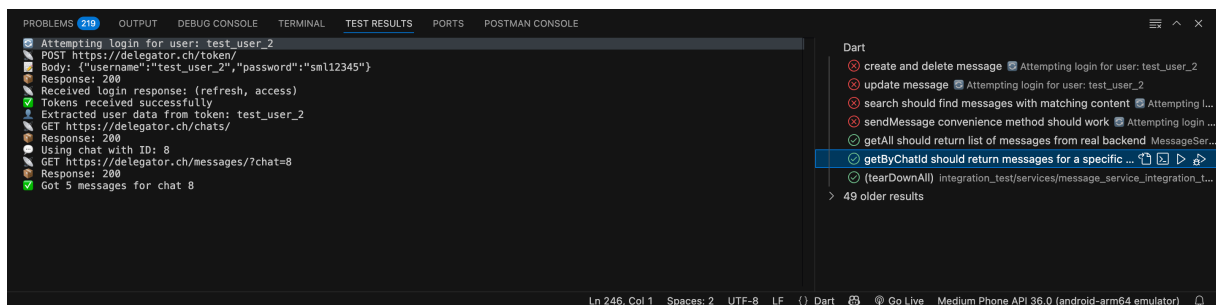
```



PROBLEMS (219) OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS POSTMAN CONSOLE

Attempting login for user: test_user_2
 POST https://delegator.ch/token/
 Body: {"username":"test_user_2","password":"sm12345"}
 Response: 200
 Received login response: (refresh, access)
 Tokens received successfully
 Extracted user data from token: test_user_2
 GET https://delegator.ch/chats/
 Response: 200
 Using chat with ID: 8
 GET https://delegator.ch/messages/
 Response: 200
 Got 10 messages
 Message ID: 17, Chat ID: 8, Content: "Test"
 Message ID: 19, Chat ID: 8, Content: "Das war ein test"
 Message ID: 29, Chat ID: 8, Content: "Asdasd"
 Message ID: 28, Chat ID: 8, Content: "Hallo Mxim"
 Message ID: 36, Chat ID: 8, Content: "Asdasd"

Dart
 create and delete message Attempting login for user: test_user_2
 update message Attempting login for user: test_user_2
 search should find messages with matching content Attempting login ...
 sendMessage convenience method should work Attempting login ...
 getAll should return list of messages from real backend MessageSer...
 getByChatId should return messages for a specific chat MessageSe...
 (tearDownAll) integration_test/services/message_service_integration_t...
 > 49 older results

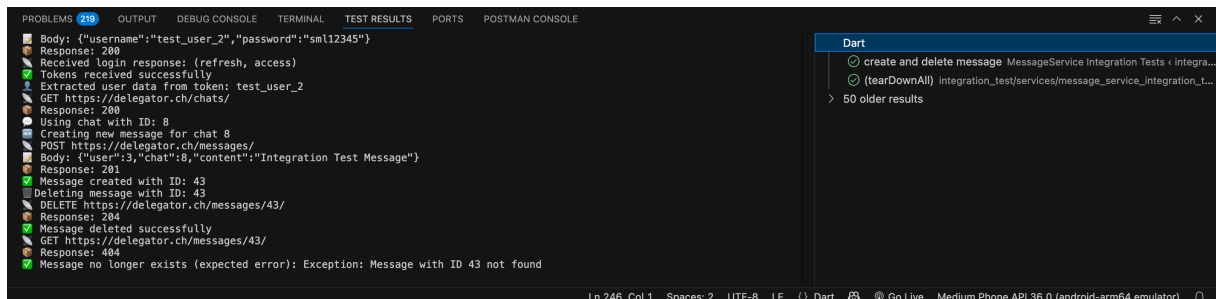


PROBLEMS (219) OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS POSTMAN CONSOLE

Attempting login for user: test_user_2
 POST https://delegator.ch/token/
 Body: {"username":"test_user_2","password":"sm12345"}
 Response: 200
 Received login response: (refresh, access)
 Tokens received successfully
 Extracted user data from token: test_user_2
 GET https://delegator.ch/chats/
 Response: 200
 Using chat with ID: 8
 GET https://delegator.ch/messages/?chat=8
 Response: 200
 Got 5 messages for chat 8

Dart
 create and delete message MessageService Integration Tests < integra...
 (tearDownAll) integration_test/services/message_service_integration_t...
 > 50 older results

4.6.2 Delete



PROBLEMS (219) OUTPUT DEBUG CONSOLE TERMINAL TEST RESULTS PORTS POSTMAN CONSOLE

Attempting login for user: test_user_2
 POST https://delegator.ch/token/
 Body: {"username":"test_user_2","password":"sm12345"}
 Response: 200
 Received login response: (refresh, access)
 Tokens received successfully
 Extracted user data from token: test_user_2
 GET https://delegator.ch/chats/
 Response: 200
 Using chat with ID: 8
 Creating new message for chat 8
 POST https://delegator.ch/messages/
 Body: {"user":3,"chat":8,"content":"Integration Test Message"}
 Response: 201
 Message created with ID: 43
 Deleting message with ID: 43
 DELETE https://delegator.ch/messages/43/
 Response: 204
 Message deleted successfully
 GET https://delegator.ch/messages/43/
 Response: 404
 Message no longer exists (expected error): Exception: Message with ID 43 not found

Dart
 create and delete message MessageService Integration Tests < integra...
 (tearDownAll) integration_test/services/message_service_integration_t...
 > 50 older results

DELETE
https://delegator.ch/messages/39/
Send

Params
Authorization
Headers (7)
Body
Scripts
Settings
Code
Cookies

Query Params

Key	Value
Key	Value

Body
Cookies
Headers (15)
Test Results

Status: 204 No Content
Time: 244 ms
Size: 930 B

Pretty
Raw
Preview
Text

1

PROBLEMS (219)
OUTPUT
DEBUG CONSOLE
TERMINAL
TEST RESULTS
PORTS
POSTMAN CONSOLE

```

Attempting login for users: test_user_2
POST https://delegator.ch/token/
Body: {"username":"test_user_2","password":"sml12345"}
Response: 200
Received login response: (refresh, access)
Tokens received successfully
Extracted user data from token: test_user_2
GET https://delegator.ch/chats/
Response: 200
Using chat with ID: 8
Sending message to chat 8
POST https://delegator.ch/messages/
Body: {"user":3,"chat":8,"content":"Sent with convenience method"}
Response: 201
Message sent with ID: 44
Cleaning up - deleting test message
DELETE https://delegator.ch/messages/44/
Response: 204

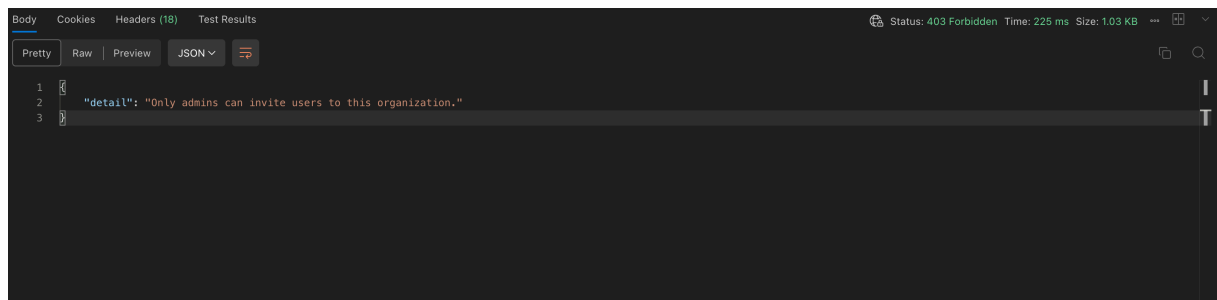
```

Dart
sendMessage convenience method should work: MessageService Int...
(tearDownAll) integration_test/services/message_service_integration_t...
51 older results

Ln 246, Col 1
Spaces: 2
UTF-8
LF
Dart
Go Live
Medium Phone API 36.0 (android-arm64 emulator)

4.7 Invitation

4.7.1 Org im not a part of



Abkürzungen und Glossar

Abkürzung / Fach- wort	Erläuterung
HERMES	Vorgehensmethodik für Projekte und Programme HERMES 5 ist ein eCH Standard

Tabelle 2: Abkürzungen und Glossar

Inhaltsverzeichnis

1	Testinterface	Error! Bookmark not defined.
2	Testen und so.....	Error! Bookmark not defined.
2.1	Projects.....	27
2.1.1	Project Get.....	27
2.1.2	Project Delete & Get By Id.....	27
2.1.3	Project Delete.....	27
2.2	Events	28
2.3	Get Events	28
2.4	Create and Delete.....	29
2.5	Tasks.....	30
2.5.1	Get.....	30
2.5.2	Create and Delete.....	30
2.6	Messages.....	31
2.6.1	Get.....	31
2.6.2	Delete.....	31
2.7	Invitation.....	33
2.7.1	Org im not a part of	33

Abbildungsverzeichnis

Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.

Tabellenverzeichnis

Tabelle 1: Änderungsverzeichnis	1
Tabelle 4: Abkürzungen und Glossar	34