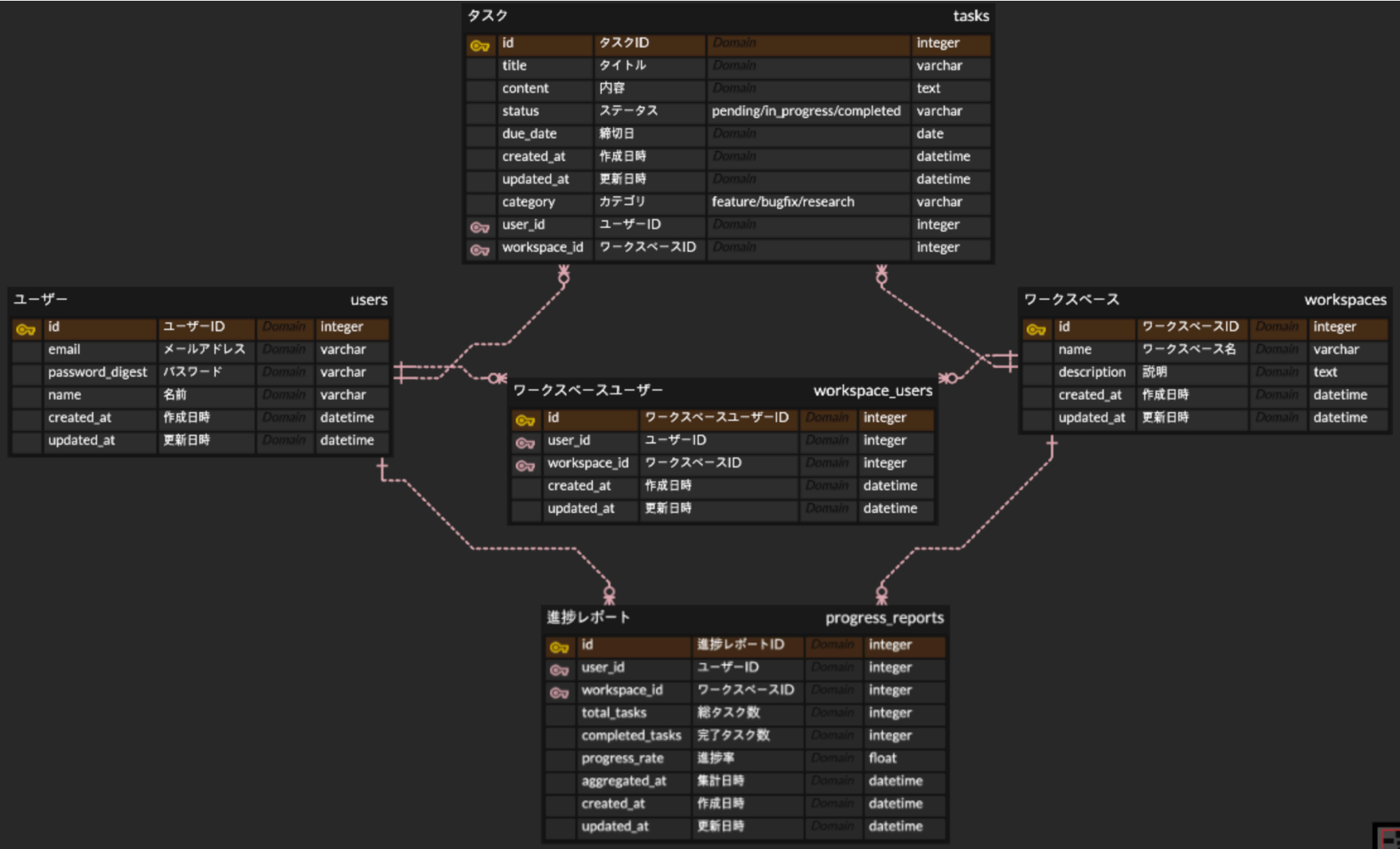


設計書

[モデル設計書]

< ER図 >



<テーブル・カラム・リレーション構成>

-データベース全体構成について

本システムでは、ユーザーが所属するワークスペース内でタスクを管理できるように、以下の4つの主要なテーブルを設計しています：

- users：ユーザー情報を管理
- workspaces：作業空間（ワークスペース）情報を管理
- tasks：タスク情報を管理
- workspace_users：ユーザーとワークスペースの多対多関係を管理する中間テーブル
- progress_reports：ユーザー別・ワークスペース別のタスク進捗集計結果を管理

-各テーブルの構成

- **users** テーブル

ユーザーの基本情報を管理します。

カラム名	データ型	内容
id	integer	ユーザーID（PK）
email	varchar	メールアドレス

カラム名	データ型	内容
password_digest	varchar	パスワード
name	varchar	名前
created_at	datetime	作成日時
updated_at	datetime	更新日時

- `workspaces` テーブル

ユーザーが参加するワークスペース情報を管理します。

カラム名	データ型	内容
id	integer	ワークスペースID (PK)
name	varchar	ワークスペース名
description	text	説明
created_at	datetime	作成日時
updated_at	datetime	更新日時

- `tasks` テーブル

各ユーザーがワークスペース内で作成するタスク情報を管理します。

カラム名	データ型	内容
id	integer	タスクID (PK)
title	varchar	タイトル
content	text	詳細内容
status	varchar	ステータス (例: pending, in_progress, completed)
due_date	date	締切日
created_at	datetime	作成日時
updated_at	datetime	更新日時
category	varchar	カテゴリ (例: feature, bugfix, research)
user_id	integer	作成者のユーザーID (外部キー)
workspace_id	integer	関連ワークスペースID (外部キー)

- `workspace_users` テーブル

1人のユーザーが複数のワークスペースに所属できるようにする中間テーブルです。

カラム名	データ型	内容
id	integer	ワークスペースユーザーID (PK)
user_id	integer	ユーザーID (外部キー)
workspace_id	integer	ワークスペースID (外部キー)
created_at	datetime	作成日時
updated_at	datetime	更新日時

- `progress_reports` テーブル

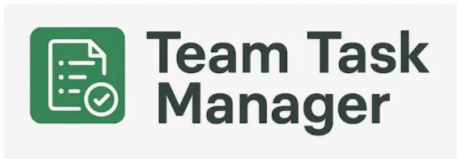
タスク進捗集計結果を管理します。

カラム名	データ型	内容
id	integer	進捗レポートID (PK)
user_id	integer	ユーザーID (外部キー)

カラム名	データ型	内容
workspace_id	integer	ワークスペースID（外部キー）
total_tasks	integer	総タスク数
completed_tasks	integer	完了タスク数
progress_rate	float	進捗率（completed_tasks ÷ total_tasks × 100 等）
aggregated_at	datetime	集計日時
created_at	datetime	レコード作成日時
updated_at	datetime	レコード更新日時

[画面仕様書]

<会員登録>



[会員登録](#) | [ログイン](#) | [ログアウト](#)

会員登録

メールアドレス

名前

パスワード

パスワード（確認）

登録する

すでにアカウントをお持ちですか？ [こちらからログイン](#)

• 入力項目

項目名	v-model	form要素	type	必須/任意
メールアドレス	form.email	<input id="email" ... />	email	必須
名前	form.name	<input id="name" ... />	text	必須
パスワード	form.password	<input id="password" ... />	password	必須
パスワード（確認）	form.password_confirmation	<input id="password_confirmation" ... />	password	必須

• エラー処理

-ブラウザ標準の必須チェック

条件：required属性付きフィールドが空

動作：ネイティブなインラインメッセージが表示され、onSubmitは呼ばれない

例)

メールアドレス

o

!

이메일 주소에 '@'를 포함해 주세요. 'o'에 '@'가 없습니다.

-メールアドレスに「@」を含めてください。「O」に「@」がありません。

メールアドレス

!

이 입력란을 작성하세요.

-この入力欄を入力してください。

<ログイン>



ログイン

メールアドレス

パスワード

ログイン

アカウントをお持ちでない方は[こちらから会員登録](#)

• 入力項目

項目名	v-model	form要素	type	必須/任意
メールアドレス	form.email	<input id="email" type="email" v-model="form.email" required>	email	必須
パスワード	form.password	<input id="password" type="password" v-model="form.password" required>	password	必須

• エラー処理

-ブラウザ標準の必須チェック

条件：required属性付きフィールドが空

動作：ネイティブなインラインメッセージが表示され、onSubmitは呼ばれない

-処理フロー

1. ユーザー検索

- `User.find_by(email: email)` で該当メールのユーザーを検索

2. パスワード検証

- `user&.authenticate(password)` でハッシュ化済みパスワードをチェック

3. 認証結果を返却

- 成功時 → `AuthResult.new(true, user, nil)`
- 失敗時 → `AuthResult.new(false, nil, ['メールアドレスまたはパスワードが正しくありません。'])`

<ログアウト>



[会員登録](#) | [ログイン](#) | [ログアウト](#)

ログアウト完了

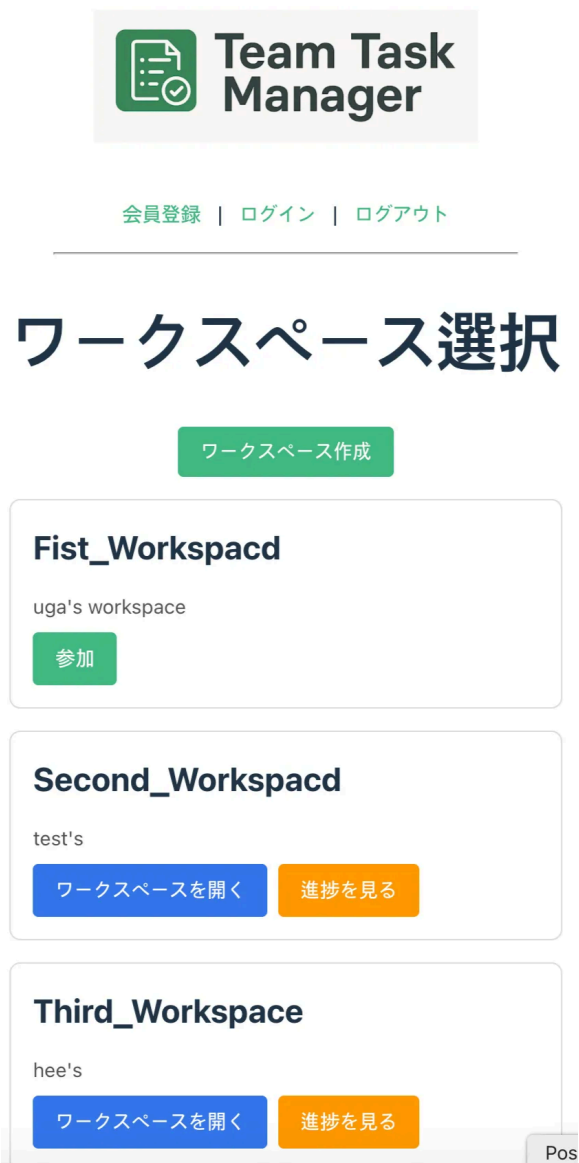
ログアウトしました。

[再度ログイン](#)

ログアウトボタンをクリックすると、ログアウトが完了します。

<ワークスペース選択>

ログイン後、作成済みのワークスペース一覧画面が表示, ユーザーは各ワークスペースに参加できる



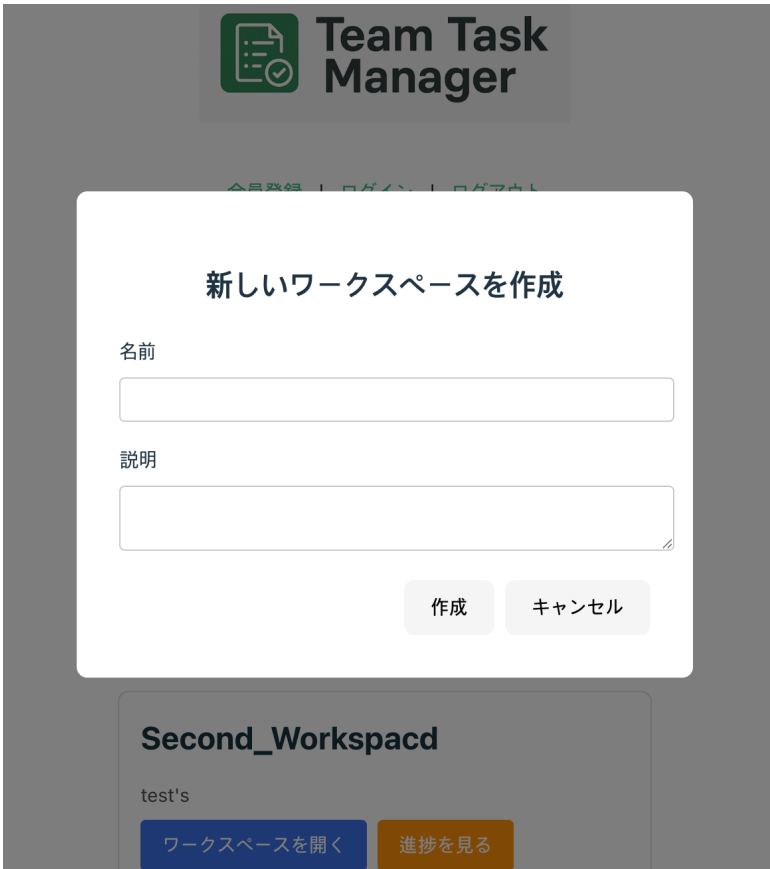
-未参加のワークスペースには「参加」ボタンが表示される

-参加済みのワークスペースには「ワークスペースを開く」ボタンと「進捗を見る」ボタンが表示される

-画面上部の「ワークスペース作成」 ボタンから新規ワークスペースを作成できる

<ワークスペース作成>

「ワークスペース作成」 ボタンをクリックすると、ワークスペースを作成できるモーダルウィンドウが表示される。



- 入力項目

項目名	v-model	form要素	type	必須／任意
名前	newWorkspace.name	<input id="name" type="text" v-model="newWorkspace.name" required>	text	必須
説明	newWorkspace.description	<textarea id="description" v-model="newWorkspace.description">	textarea	任意

- エラー処理

-ブラウザ標準の必須チェック

条件：required属性付きフィールドが空

動作：ネイティブなインラインメッセージが表示され、onSubmitは呼ばれない

<タスク一覧>

ワークスペース選択画面で「ワークスペースを開く」 ボタンを押すと、該当ワークスペース内のタスク一覧画面が表示される。

Fist_Workspacd のタスク一覧

— 担当者と絞り込み —

— カテゴリで絞り込み —

— ステータスで絞り込み —

戻る

新しいタスクを作成

VIEW setting

編集

削除

set

担当者: uga カテゴリ: feature ステータス: completed 締切: 2025-05-01

test working

編集

削除

test

担当者: uga カテゴリ: research ステータス: pending 締切: 2025-05-03

Ruby setting

編集

削除

setting

担当者: uga カテゴリ: research ステータス: pending 締切: 2025-05-07

UX setting

編集

削除

Visual Studio

<タスクのフィルタリング>

担当者・カテゴリ・ステータス別にタスクをフィルタリングできます。

- 担当者

✓ — 担当者と絞り込み —

uga

YU

heeEun

uga

— カテゴリで絞り込み —

— ステータスで絞り込み —

戻る

新しいタスクを作成

VIEW setting

編集

削除

set

担当者: uga カテゴリ: feature ステータス: completed 締切: 2025-05-01

test working

編集

削除

test

担当者: uga カテゴリ: research ステータス: pending 締切: 2025-05-03

Ruby setting

編集

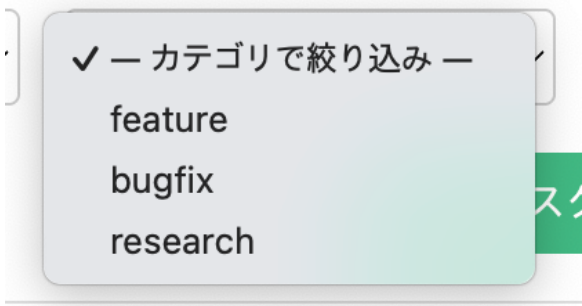
削除

setting

担当者: uga カテゴリ: research ステータス: pending 締切: 2025-05-07

各ワークスペースでは、タスク担当者の名前が表示され、担当者ごとにタスクを絞り込んで閲覧できます。

- カテゴリ



— 担当者で絞り込み — ▾ **bugfix** ▾ — ステータスで絞り込み — ▾

[戻る](#) [新しいタスクを作成](#)

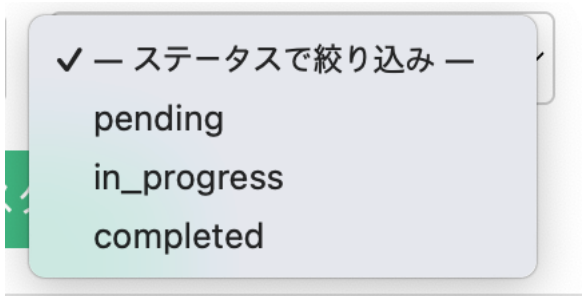
UX setting	編集 削除
setting	
担当者: YU カテゴリ: bugfix ステータス: completed 締切: 2025-05-18	

screen setting	編集 削除
it's screen setting	
担当者: heeEun カテゴリ: bugfix ステータス: in_progress 締切: 2025-05-31	

UI update	編集 削除
update	
担当者: YU カテゴリ: bugfix ステータス: completed 締切: 2025-06-06	

カテゴリ別にタスクを絞り込んで閲覧できます。

- ステータス



— 担当者で絞り込み — ▾ — カテゴリで絞り込み — ▾ **completed** ▾

[戻る](#) [新しいタスクを作成](#)

VIEW setting	編集 削除
set	
担当者: uga カテゴリ: feature ステータス: completed 締切: 2025-05-01	

UX setting	編集 削除
setting	
担当者: YU カテゴリ: bugfix ステータス: completed 締切: 2025-05-18	

EC2 setting	編集 削除
AWS	
担当者: heeEun カテゴリ: research ステータス: completed 締切: 2025-05-26	

UI update	編集 削除
update	
担当者: YU カテゴリ: bugfix ステータス: completed 締切: 2025-06-06	

ステータス別にタスクを絞り込んで閲覧できます。

<タスク作成>

「新しいタスク作成」 ボタンをクリックすると、タスクを作成できるモーダルウィンドウが表示される。



• 入力項目

項目名	v-model	form要素	type	必須／任意
タイトル	newTask.title	<input id="title" type="text" v-model="newTask.title" required placeholder="タスクのタイトルを入力">	text	必須
詳細	newTask.content	<textarea id="content" v-model="newTask.content" placeholder="詳細を入力">	text	任意
締切日	newTask.due_date	<input id="due_date" type="date" v-model="newTask.due_date" required>	date	必須
カテゴリ	newTask.category	<select id="category" v-model="newTask.category" required>...</select>	select	必須

• エラー処理

-ブラウザ標準の必須チェック

条件：required属性付きフィールドが空

動作：ネイティブなインラインメッセージが表示され、onSubmitは呼ばれない

<タスク編集>

各タスクの「編集」 ボタンをクリックすると、既存の値が入力されたタスク編集モーダルウィンドウが表示される。

編集モーダルでは、ステータスを変更できるようになっている。

タスクを編集

タイトル

VIEW setting

詳細

set

締切日

2025. 05. 01.

カテゴリ

feature

ステータス

completed

キャンセル

更新

- 入力項目

項目名	v-model	form要素	type	必須／任意
タイトル	editTask.title	<input id="edit-title" type="text" v-model="editTask.title" required>	text	必須
詳細	editTask.content	<textarea id="edit-content" v-model="editTask.content">	textarea	任意
締切日	editTask.due_date	<input id="edit-due_date" type="date" v-model="editTask.due_date" required>	date	必須
カテゴリ	editTask.category	<select id="edit-category" v-model="editTask.category" required>...</select>	select	必須
ステータス	editTask.status	<select id="edit-status" v-model="editTask.status" required>...</select>	select	必須

- エラー処理

-ブラウザ標準の必須チェック

条件：required属性付きフィールドが空

動作：ネイティブなインラインメッセージが表示され、onSubmitは呼ばれない

<タスク削除>

各タスクの「削除」ボタンをクリックすると、確認ダイアログ「本当に削除しますか？」が表示される
ユーザーが「確認」を押した場合にのみ、削除処理を実行する, 「キャンセル」を押すと、何もせずダイアログを閉じる

本当に削除しますか？

취소

확인

VIEW setting

<進捗レポート>

各ワークスペースの担当者別タスク進捗を一覧表示する「進捗レポート画面」
本画面の集計データは、毎日深夜2時に実行されるバッチ処理（Rakeタスク）によって自動で更新される。



[会員登録](#) | [ログイン](#) | [ログアウト](#)

Fist_Workspacd の進捗レポート

担当者	全タスク数	完了タスク数	進捗率
YU	2	2	100%
heeEun	2	1	50%
uga	3	1	33.3%

戻る

[機能設計書]

≡ Index	Aa feat	⌵ http	🔗 URL	≡ リクエスト例
User	会員登録	POST	/api/v1/users	{ "user": { "email": " <u>ugauga@example.com</u> ", "password": "qweasd", "password_confirmation": "qweasd", "name": "uga" } }
	<u>ログイン</u>	POST	/api/v1/login	{ "email": " <u>ugauga@example.com</u> ", "password": "qweasd" }
	<u>ログアウト</u>	DELETE	/api/v1/logout	
Workspace	<u>全ワークスペースの一覧取得</u>	GET	/api/v1/workspaces/public	
	<u>ワークスペース作成</u>	POST	/api/v1/workspaces	{ "workspace": { "name": "first_workspace", "description": "description" } }
	<u>ワークスペース参加</u>	POST	/api/v1/workspaces/:id/join	

≡ Index	Aa feat	📶 http	🔗 URL	≡ リクエスト例
Task	<u>タスク一覧取得</u>	GET	<u>/api/v1/workspaces/:workspace_id/tasks</u>	
	<u>タスク作成</u>	POST	<u>/api/v1/workspaces/:workspace_id/tasks</u>	{ "task": { "title": "Title!", "content": "it's ~~", "status": "pending", "due_date": "2025-06-01", "category": "feature" } }
	제목 없음			
	<u>タスク編集</u>	PATCH	<u>/api/v1/tasks/:id</u>	{ "task": { "status": "in_progress", "category": "research", "content": "content", "due_date": "2025-06-15", "title" : "change title" } }
	<u>タスク削除</u>	DELETE	<u>/api/v1/tasks/:id</u>	
	<u>担当者別タスク一覧の取得</u>	GET	<u>/api/v1/workspaces/:workspace_id/tasks?assignee_id=</u>	
	<u>カテゴリ別タスク一覧の取得</u>	GET	<u>/api/v1/workspaces/:workspace_id/tasks?&category=</u>	
	<u>ステータス別タスク一覧の取得</u>	GET	<u>/api/v1/workspaces/:workspace_id/tasks?&status=</u>	

[使用技術・ライブラリー覧]

- **バックエンド**
 - Ruby on Rails （APIモード）
- **フロントエンド**
 - Vue.js 3
 - Vite
 - Axios
- **テンプレートエンジン**
 - Pug
- **データベース**
 - SQLite3
- **認証方式**
 - Railsセッション方式
- **バッチ処理**
 - Rakeタスク（担当者別タスク進捗集計用）

