TaskFlow

Task & Project Manager

Tesina Tecnologie Web
UNIMORE - 2026

Colince Tcheussieu Mendji

Cos'è TaskFlow?

Applicazione web per la gestione di progetti e task

✅ Obiettivi

- Organizzare progetti in modo efficiente
- Tracciare lo stato dei task
- Visualizzare i progressi
- Interfaccia semplice e intuitiva

🎯 Target

- Studenti e professionisti
- Team di lavoro
- Gestione personale
- Organizzazione progetti

Stack Tecnologico

Backend

- 🐍 **Python 3** - Linguaggio di programmazione
- 🌶️ **Flask** - Framework web minimalista
- 📊 **CSV** - Storage dati semplice e portabile

API

- 🔌 **REST API** - Endpoints JSON per accesso dati

Frontend

- 📝 **HTML5** - Struttura delle pagine
- 🎨 **CSS3** - Styling e responsive design
- 🔧 **Jinja2** - Template engine di Flask
- ⚛️ **React** - Dashboard interattiva

Struttura dell'Applicazione

```
TaskFlow/
├── app.py                      # Backend Flask (270 righe)
├── requirements.txt            # Dipendenze Python
├── static/
│   ├── data/                  # Database CSV
│   │   ├── projects.csv       # Progetti
│   │   └── tasks.csv          # Task
│   └── css/
│       └── style.css          # Stili (700+ righe)
└── templates/                 # Template Jinja2
    ├── base.html              # Template base
    ├── index.html             # Homepage
    ├── project_detail.html    # Dettaglio progetto
    ├── create_project.html    # Form nuovo progetto
    ├── create_task.html       # Form nuovo task
    └── dashboard.html         # Dashboard React
```

Pattern Architetturale: MVC (Model-View-Controller)

Le 5 Pagine dell'Applicazione

1. Homepage (/)

Lista di tutti i progetti con statistiche, progress bar e bottone "Nuovo Progetto"

2. Dettaglio Progetto

Visualizza tutti i task del progetto con badge status/priorità e azioni rapide

3. Nuovo Progetto

Form con nome, descrizione e selezione colore personalizzato

4. Nuovo Task

Form completo con status, priorità e descrizione del task

5. Dashboard React

Statistiche generali e Kanban board interattiva con dati caricati via API REST

Funzionalità CRUD Complete

PROGETTI

- ✅ **CREATE** - Form di creazione con validazione
- ✅ **READ** - Lista progetti e dettaglio singolo
- ✅ **UPDATE** - Modifica status task associati
- ✅ **DELETE** - Eliminazione progetto e task collegati

TASK

- ✅ **CREATE** - Form con tutti i campi
- ✅ **READ** - Visualizzazione lista task
- ✅ **UPDATE** - Cambio status (Da Fare → In Corso → Completato)
- ✅ **DELETE** - Eliminazione task

Storage: Dati salvati in file CSV persistenti

API REST in Formato JSON

3 Endpoint Disponibili

1. GET /api/projects

Restituisce tutti i progetti in formato JSON

2. GET /api/tasks

Restituisce tutti i task in formato JSON

3. GET /api/projects/<id>/tasks

Restituisce i task di un progetto specifico in formato JSON filtrato

La Dashboard React consuma queste API per visualizzare i dati in tempo reale

Dashboard Interattiva con React

📊 Statistiche Generali

- Progetti totali
- Task totali
- Task completati
- Percentuale completamento

📋 Kanban Board

- 3 colonne: Da Fare, In Corso, Completati
- Card colorate per progetto
- Badge priorità visibili
- Aggiornamento real-time

Tecnologia: React 18 con Hooks

useState e useEffect per gestione stato e side effects

Design Moderno e Responsive

Caratteristiche del Design

- 🎨 **Colori personalizzabili** per progetti
- 📱 **Responsive Design** - Mobile, tablet, desktop
- ⚡ **Animazioni fluide** e transizioni
- 🎯 **UI intuitiva** - Facile da usare
- 🏷️ **Badge visivi** per status e priorità
- 📊 **Progress bar** per completamento
- 🎨 **Hover effects** per feedback
- ✨ **Gradient moderni** e ombre

700+ righe di CSS

Variabili CSS • Grid Layout • Media Queries • Mobile-First

Gestione Dati con CSV

Format projects.csv

```
id,name,description,color,created_at
1,Sito Web,Descrizione progetto,#3b82f6,2024-01-15T10:00:00
```

Format tasks.csv

```
id,project_id,title,description,status,priority,created_at
1,1,Task 1,Descrizione task,todo,high,2024-01-16T10:00:00
```

Vantaggi

- ✅ Nessun database esterno necessario
- ✅ Dati human-readable
- ✅ Facile backup e modifica
- ✅ Portabilità massima

Requisiti Tesina Soddisfatti

Requisito	Status	Implementazione
Almeno 4 pagine	✅	5 pagine HTML complete
Template	✅	Jinja2 con ereditarietà
CSS	✅	700+ righe responsive
JavaScript	✅	React Dashboard interattiva
API	✅	3 endpoint REST JSON
CRUD	✅	Completo per progetti e task
Codice funzionante	✅	Testato e operativo

Snippet di Codice - Backend Flask

Esempio: Route per creare un progetto

```
@app.route('/create-project', methods=['POST'])
def create_project():
    name = request.form.get('name')
    description = request.form.get('description', '')
    color = request.form.get('color', '#3b82f6')

    # Genera nuovo ID
    projects = get_projects()
    new_id = str(max([int(p['id']) for p in projects]) + 1)

    # Salva nel CSV
    csv_path = os.path.join('static', 'data', 'projects.csv')
    with open(csv_path, 'a', encoding='utf-8') as file:
        writer = csv.writer(file)
        writer.writerow([new_id, name, description, color,
                         datetime.now().isoformat()])

    return redirect(url_for('index'))
```

Snippet di Codice - Frontend React

Esempio: Componente Kanban con React

```
function KanbanBoard() {
    const [tasks, setTasks] = useState([]);

    useEffect(() => {
        fetch('/api/tasks')
            .then(res => res.json())
            .then(data => setTasks(data));
    }, []);

    const todoTasks = tasks.filter(t => t.status === 'todo');
    const inProgressTasks = tasks.filter(t => t.status === 'in_progress');
    const doneTasks = tasks.filter(t => t.status === 'done');

    return (
        <div className="kanban-board">
            <KanbanColumn title="Da Fare" tasks={todoTasks} />
            <KanbanColumn title="In Corso" tasks={inProgressTasks} />
            <KanbanColumn title="Completati" tasks={doneTasks} />
        </div>
    );
}
```

Competenze Acquisite

Cosa Ho Imparato

- ✅ Sviluppo web full-stack
- ✅ Design di API REST
- ✅ Template engine Jinja2
- ✅ CSS avanzato e responsive
- ✅ Gestione stato con React Hooks
- ✅ Architettura MVC
- ✅ Storage dati con CSV
- ✅ Best practices web development

Risultato

Un'applicazione web completa, funzionante e pronta all'uso che dimostra padronanza delle tecnologie web moderne

Grazie per l'Attenzione!

TaskFlow - Task & Project Manager

Tesina Tecnologie Web

UNIMORE - 2026

Colince Tcheussieu Mendji

Demo Live: http://127.0.0.1:5000

Domande?