

Software Architecture Diagram

Customer (External)

The System (Internal)

معماری سیستم مدیریت پروژه خودکار

راهنمای جامع معماری و پیاده‌سازی سیستم مدیریت پروژه خودکار مبتنی بر پایتون



یکپارچه‌سازی با
گیت‌هاب



REST API



رابط خط فرمان

Integration Service
(portal)

Reporting service

Data Storage
(ETL/OLAP)

The high-Level breakdown of a system into its parts

نسخه 2.0.0 | تاریخ به‌روزرسانی: ۱۶ اوت ۲۰۲۵

مرور کلی سیستم و معماری

سیستم مدیریت پروژه خودکار

یک سیستم جامع مبتنی بر پایتون با رابط خط فرمان و API برای مدیریت خودکار پروژه‌ها، پیگیری وظایف و یکپارچه‌سازی با گیت‌هاب.

✓ رابط خط فرمان قدرتمند

✓ REST API برای یکپارچه‌سازی

✓ ذخیره‌سازی داده‌ها در قالب JSON

✓ همگام‌سازی با گیت‌هاب

لایه رابط کاربری

< > افزونه VSCode

REST API ❖

📄 رابط خط فرمان

لایه سرویس

↻ مدیریت یکپارچه‌سازی

⚙️ دستورات CLI

↻ سرویس‌های خودکارسازی

منطق تجاری اصلی

📅 برنامه‌ریزی

📋 مدیریت وظایف

📁 مدیریت پروژه

📈 گزارش پیشرفت

لایه داده

📦 یکپارچه‌سازی گیت‌هاب

📄 پایگاه داده JSON

پشته فناوری

سیستم مدیریت پروژه خودکار با استفاده از مجموعه‌ای از **فناوری‌های مدرن** و **بهینه** توسعه یافته است

FastAPI



نسخه: +0.68

نوع: چارچوب API

هدف:

نقاط پایانی REST API

Click



نسخه: +8.0

نوع: چارچوب CLI

هدف:

رابط خط فرمان

پایتون



نسخه: +3.8

نوع: زبان اصلی

هدف:

پیاده‌سازی اصلی سیستم

pytest



نسخه: +7.0

نوع: چارچوب تست

هدف:

تست‌های واحد و یکپارچه

httpx



نسخه: +0.24

نوع: کلاینت HTTP

هدف:

فراخوانی‌های API گیت‌هاب

فایل‌های JSON



نسخه: بومی

نوع: ذخیره‌سازی داده

هدف:

مانایی داده‌ها

معماری رابط خط فرمان

سیستم مدیریت پروژه خودکار دارای **رابط خط فرمان قدرتمند** با استفاده از چارچوب Click است که امکان مدیریت کامل پروژه‌ها را فراهم می‌کند

create-project

ایجاد پروژه جدید

پارامتر: project_name

مثال: "apm create-project "MyProject"

init

مقداردهی اولیه تنظیمات سیستم

پارامتر: config-path--

مثال: apm init --config-path ./config

status

نمایش وضعیت پروژه

پارامتر: project--

مثال: apm status --project MyProject

add-task

افزودن وظیفه به پروژه

پارامتر: project, --title--

مثال: apm add-task --project MyProject --
"title "Implement feature"

sync-github

همگام‌سازی با گیت‌هاب

پارامتر: project--

مثال: apm sync-github --project MyProject

report

تولید گزارش‌ها

پارامتر: type, --format--

مثال: apm report --type progress --format
markdown

سیستم مدیریت پروژه اصلی

قابلیت‌های اصلی ⚙️

مدیریت پروژه 📁

ایجاد، خواندن، به‌روزرسانی و حذف پروژه‌ها

`()remove_project` `()add_project`

`()get_project` `()update_project`

مدیریت وظایف 📝

مدیریت وظایف درون پروژه‌ها

`()add_task_to_project`

`()remove_task_from_project`

`()update_task_in_project`

اعتبارسنجی داده‌ها ✅

تضمین یکپارچگی داده‌ها

`بررسی فیلدها` `اعتبارسنجی داخلی`

مانایی JSON 📄

ذخیره داده‌ها در فرمت JSON

`سریال‌سازی خودکار` `دیسریال‌سازی خودکار`

{ ساختار داده‌ها

ساختار پروژه 📁

```
{ "id": int, "name": str, "description": str,
  "created_date": str, "updated_date": str,
  "status": str, "priority": str, "owner": str,
  "tags": List[str] }
```

ساختار وظیفه 📝

```
{ "id": int, "title": str, "description": str,
  "project_id": int, "status": str, "priority":
  str, "assignee": str, "due_date": str,
  "created_date": str, "updated_date": str,
  "tags": List[str] }
```

سیستم مدیریت پروژه خودکار دارای **REST API قدرتمند** با استفاده از چارچوب FastAPI است که امکان یکپارچه‌سازی با سایر سیستم‌ها را فراهم می‌کند

/api/projects

POST

ایجاد پروژه جدید

JSON project object

درخواست:

JSON project

پاسخ:

/api/projects

GET

لیست تمام پروژه‌ها

-

درخواست:

JSON array

پاسخ:

/api/projects/{id}

PUT

به‌روزرسانی پروژه

JSON project object

درخواست:

JSON project

پاسخ:

/api/projects/{id}

GET

دریافت پروژه با شناسه

-

درخواست:

JSON project

پاسخ:

/api/projects/{id}/tasks

GET

لیست وظایف پروژه

-

درخواست:

JSON array

پاسخ:

/api/projects/{id}

DELETE

حذف پروژه

-

درخواست:

JSON confirmation

پاسخ:

/api/projects/{id}/tasks

POST

افزودن وظیفه به پروژه

JSON task object

درخواست:

JSON task

پاسخ:

معماری ذخیره‌سازی داده‌ها

ساختار دایرکتوری

JsonDataBase/
Inputs/
 UserInputs/
 project_config.json
 user_preferences.json
 SystemGeneratorInputs/
 system_defaults.json
OutPuts/
 commit_progress.json
 commit_task_database.json
 progress_report.md
 project_data.json
Backups/
 backup_*.json
metadata/
 backup_*.json

فایل‌های داده

commit_progress.json

پیگیری پیشرفت کامیت‌ها

```
{
  project_id: {
    task_id: status
  }
}
```

commit_task_database.json

پایگاه داده وظایف

```
{
  tasks: [task_objects]
}
```

project_data.json

داده‌های اصلی پروژه

```
{
  projects: [project_objects]
}
```

progress_report.md

گزارش‌های پیشرفت در قالب Markdown

گزارش‌های قابل خواندن توسط انسان

یکپارچه سازی با گیت هاب

سیستم مدیریت پروژه خودکار دارای یکپارچه سازی کامل با گیت هاب است که امکان همگام سازی پروژه ها، وظایف و مستندات را فراهم می کند

تخته های پروژه

ایجاد تخته های پروژه برای مدیریت وظایف

POST /repos/{owner}/{repo}/projects ❖

ایجاد مخازن

ایجاد مخازن گیت هاب برای پروژه ها

POST /user/repos ❖

مدیریت نقاط عطف

ایجاد نقاط عطف برای پروژه ها

POST /repos/{owner}/{repo}/milestones ❖

مدیریت مسائل

ایجاد و به روز رسانی مسائل گیت هاب

POST /repos/{owner}/{repo}/issues ❖

همگام سازی خودکار

همگام سازی خودکار داده ها با گیت هاب

🕒 زمان بندی شده و بر اساس رویداد

مدیریت برچسب ها

ایجاد و مدیریت برچسب های مسائل

POST /repos/{owner}/{repo}/labels ❖

سرویس‌های خودکارسازی

سیستم مدیریت پروژه خودکار دارای **سرویس‌های خودکارسازی قدرتمند** است که کارایی و بهره‌وری را افزایش می‌دهند

کامیت خودکار

انجام خودکار کامیت‌های گیت



فعال‌سازی

زمان‌بندی شده یا بر اساس تغییر

خروجی

کامیت‌های گیت با پیشرفت

مدیریت پشتیبان

ایجاد خودکار نسخه‌های پشتیبان



فعال‌سازی

زمان‌بندی شده

خروجی

فایل‌های فشرده با متادیتا

همگام‌سازی ویکی

همگام‌سازی خودکار مستندات



فعال‌سازی

بر اساس به‌روزرسانی پروژه

خروجی

صفحات ویکی به‌روز شده

نصب و پیکربندی

روش‌های نصب ⬇

پیکربندی سیستم ⚙

مقداردهی اولیه سیستم



راه‌اندازی اولیه سیستم مدیریت پروژه

```
apm init
```

پیکربندی گیت‌هاب



تنظیم توکن API برای یکپارچه‌سازی با گیت‌هاب

```
apm config set  
github.token  
YOUR_GITHUB_TOKEN
```

تنظیم مسیر داده‌ها



مشخص کردن مسیر ذخیره‌سازی داده‌های پروژه

```
apm config set  
data.path  
./my_project_data
```

1 نصب با pip

```
pip install autoprojectmanagement
```

2 نصب از سورس

```
git clone  
https://github.com/autoprojectmanagement/autoprojectmanagement.git  
cd autoprojectmanagement  
pip install -e .
```

3 راه‌اندازی توسعه

```
git clone  
https://github.com/autoprojectmanagement/autoprojectmanagement.git  
cd autoprojectmanagement  
pip install -r requirements-dev.txt  
python setup.py develop
```

نقشه راه آینده

ویژگی‌های برنامه‌ریزی شده

پشتیبانی از پایگاه داده
پشتیبانی از SQLite/PostgreSQL به عنوان پشتیبان
اولویت بالا
Q1 2025

رابط کاربری وب
رابط کاربری مبتنی بر React
اولویت متوسط
Q2 2025

اپلیکیشن موبایل
اپلیکیشن موبایل مبتنی بر React Native
اولویت پایین
Q3 2025

یکپارچه‌سازی هوش مصنوعی
پیشنهاد وظایف هوشمند مبتنی بر هوش مصنوعی
اولویت متوسط
Q4 2025

بدهی فنی

مدیریت خطاها
پیاده‌سازی جامع مدیریت استثناها
تأثیر بالا
اولویت بالا

لاگ‌گیری
پیاده‌سازی لاگ‌گیری ساختاریافته
تأثیر پایین
اولویت متوسط

پوشش تست
دستیابی به پوشش تست 90%+
تأثیر بالا
اولویت بالا

مستندات
تکمیل مستندات API
تأثیر متوسط
اولویت متوسط

خلاصه و نتیجه‌گیری

سیستم مدیریت پروژه خودکار یک راه‌حل جامع و آماده بهره‌برداری برای مدیریت خودکار پروژه‌ها با ویژگی‌های کلیدی زیر است:



یکپارچه‌سازی گیت‌هاب

همگام‌سازی کامل با مخازن و پروژه‌های گیت‌هاب



رابط‌های چندگانه

پشتیبانی از REST API، CLI و افزونه VSCode



معماری ماژولار

جداسازی تمیز نگرانی‌ها با رابط‌های تعریف شده



آماده بهره‌برداری

تست جامع، مدیریت خطا و مستندات



طراحی قابل گسترش

افزودن آسان ویژگی‌ها و یکپارچه‌سازی‌ها



مانایی JSON

راه‌حل ذخیره‌سازی داده‌های ساده و مؤثر

این سند معماری، یک نقشه کامل برای درک، پیاده‌سازی و گسترش سیستم مدیریت پروژه خودکار بر پایه کدبیس و پیاده‌سازی واقعی فراهم می‌کند.

