



## پروژه پایانی

درس برنامه نویسی تجهیزات اینترنت اشیا

استفاده از ESP32-CAM برای پایش و تحلیل وضعیت محیط با  
استفاده از بینایی ماشین و مدل‌های زبانی

استادان درس: علی بهلولی و دانیال فارسی

بهار ۱۴۰۴

این پروژه یک سامانه‌ی هوشمند برای پایش و تحلیل تغییرات محیطی با استفاده از بینایی ماشین و هوش مصنوعی است. این سیستم بر پایه‌ی ماژول ESP32-CAM طراحی شده و قادر است دو تصویر متوالی از محیط را با فاصله زمانی دلخواه ثبت کرده و به کمک مدل‌های زبانی بزرگ تحلیل کند. هدف، تشخیص تغییرات احتمالی در وضعیت وسایل، نور، اشیاء یا حضور افراد است، آن هم بدون نیاز به سنسورها یا تحلیل سخت‌افزاری پایه‌ها.

این پروژه با بهره‌گیری از ترکیب ماژول ESP32-CAM و ارتباط Wi-Fi و پردازش تصویری و تحلیل متنی با GPT، یک نمونه کامل از کاربردهای مدرن اینترنت اشیا و مدل‌های زبانی در محیط‌های هوشمند محسوب می‌شود.

### اهداف اصلی:

- ثبت خودکار دو تصویر از محیط با فاصله زمانی مشخص شده توسط کاربر
- ارسال تصاویر به مدل زبانی GPT برای تحلیل تصویری و متنی
- تشخیص تغییرات در وضعیت اشیاء، روشنایی، حضور افراد یا حرکات
- ارائه گزارش واضح، متنی و قابل درک شامل تحلیل، هشدارها و پیشنهادات
- فراهم‌سازی یک بستر آموزشی-عملی برای تمرین مفاهیم HTTP Server، ارتباط با API، و مهندسی پرامپت (Prompt Engineering)

### مراحل دریافت فرمان از مرورگر توسط ESP32-CAM

کاربر از طریق مرورگر یک آدرس مانند زیر را فراخوانی می‌کند تا فرمان ثبت دو تصویر متوالی را با فاصله زمانی مشخص ارسال کند:

```
http://<IP_ADDRESS>/capture?delay=15
```

### ثبت تصویر اول از محیط

پس از دریافت فرمان، ESP32-CAM بلافاصله تصویر اول را از محیط ثبت می‌کند. این تصویر وضعیت فعلی وسایل، روشنایی، یا موقعیت اشیاء را ثبت می‌کند.

### ایجاد تأخیر براساس پارامتر delay

سیستم به مدت مشخص شده در پارامتر (مثلاً ۱۵ ثانیه) صبر می‌کند.

### ثبت تصویر دوم از محیط

پس از اتمام زمان تأخیر، ESP32-CAM تصویر دوم را ثبت می‌کند تا تغییرات احتمالی نسبت به تصویر اول مشخص شوند.

### ارسال دو تصویر به همراه یک پرامپت متنی به مدل GPT از طریق API سرویس AvalAI

تصاویر به فرمت Base64 رمزگذاری می‌شوند و همراه با یک پرامپت هدفمند (Prompt) از طریق درخواست HTTP به مدل GPT ارسال می‌شوند.

### دریافت پاسخ تحلیل شده از GPT شامل تشخیص تغییرات و توصیه‌ها

پاسخ شامل مواردی مانند روشن یا خاموش شدن وسایل، جابجایی اشیاء، تغییر نور، یا حضور افراد است.

### نمایش پاسخ نهایی در Serial Monitor برای کاربر

تحلیل دریافتی به صورت گزارش متنی در Serial Monitor نمایش داده می‌شود و کاربر می‌تواند آن را بررسی یا ذخیره کند.

### ابزارها و ماژول‌های استفاده شده

#### • ESP32-CAM

ماژول اصلی برای ثبت دو تصویر متوالی از محیط. این تصاویر به عنوان ورودی برای تحلیل محیط استفاده می‌شوند.

#### • ESP32-CAM

ارتباط اینترنتی برای ارسال درخواست به API مدل GPT و دریافت پاسخ تحلیلی.

#### • HTTP Server

دریافت دستورات از کاربر از طریق آدرس دهی مرورگر، مثلاً:

`http://<IP_ADDRESS>/capture?delay=15`

#### • Serial Monitor

- نمایش وضعیت سیستم، پیام‌های سیستمی و پاسخ نهایی از GPT.
- ابزار مناسب برای تست، اشکال زدایی، و مشاهده عملکرد زنده سیستم.

- **Arduino IDE**

محیط توسعه و برنامه‌نویسی برای نوشتن، تست و آپلود کد روی ESP32-CAM.

- **AvalAI API**

سرویس واسطه ایرانی برای اتصال به مدل GPT به منظور ارسال تصاویر و دریافت تحلیل متنی. این API به صورت ساده و با قیمت مناسب برای پروژه‌های دانشگاهی قابل استفاده است.

## نکات اجرایی مهم

- **ESP32-CAM به عنوان Web Server عمل می‌کند:**

برخلاف نسخه‌های ساده‌تر که ESP32 فقط نقش Web Client را داشت، در این پروژه ESP32-CAM به صورت Web Server پیاده‌سازی شده و منتظر درخواست از کاربر می‌ماند. کاربر می‌تواند از طریق مرورگر، با ارسال آدرس شامل پارامتر delay، زمان فاصله بین دو عکس را مشخص کند.

- **تصاویر در RAM ذخیره و بلافاصله ارسال می‌شوند:**

تصاویر گرفته‌شده در حافظه موقت (RAM) ذخیره شده و سپس بدون نیاز به کارت حافظه، به فرمت Base64 تبدیل و از طریق HTTP POST به API مدل GPT ارسال می‌شوند. نیازی به ذخیره‌سازی دائمی نیست.

- **پرامپت قابل تنظیم و قابل توسعه است:**

دانشجو می‌تواند متن پرامپت را داخل برنامه به دلخواه تغییر دهد تا سبک پاسخ‌دهی GPT را کنترل کند (مثلاً رسمی، خلاصه، هشدارمحور، یا تحلیل کامل). این انعطاف‌پذیری به دانشجو اجازه می‌دهد مهارت خود را در طراحی پرامپت (Prompt Engineering) تقویت کرده و پاسخ‌های دقیق‌تر و کاربردی‌تری دریافت کند.

## نکته:

- برای اجرای پروژه فقط یک بار فراخوانی کامل GPT کافی است.
- با شارژ اندک در پلتفرم AvalAI، دانشجو می‌تواند پروژه را کامل اجرا کند.
- پیشنهاد می‌شود برای کاهش هزینه، یک پرامپت دقیق و از پیش آماده شده توسط دانشجو طراحی شود تا نیازی به فراخوانی‌های مکرر نباشد.
- به طرح‌هایی که نوآورانه‌تر، هزینه پیاده‌سازی کمتر و کارایی بالاتری داشته باشند نمره اضافه تری تخصیص می‌یابد.
- در صورت طراحی برنامه روی لپ‌تاپ برای ارتباط با پورت سریال (جایگزین ترمینال آردوینو) نمره اضافه تعلق می‌گیرد.
- در صورت استفاده از روش‌های جایگزین، به جای ارسال نتایج روی ترمینال به این شرط که مزایا و امکانات بیشتری را در اختیار کاربر قرار دهد، نمره اضافه تری تعلق خواهد گرفت.
- گزارش کامل از مراحل انجام کار و قابلیت‌های پروژه را تکمیل و در سامانه ال ام اس آپلود نمایید. تحویل به صورت شفاهی نیز انجام خواهد شد.

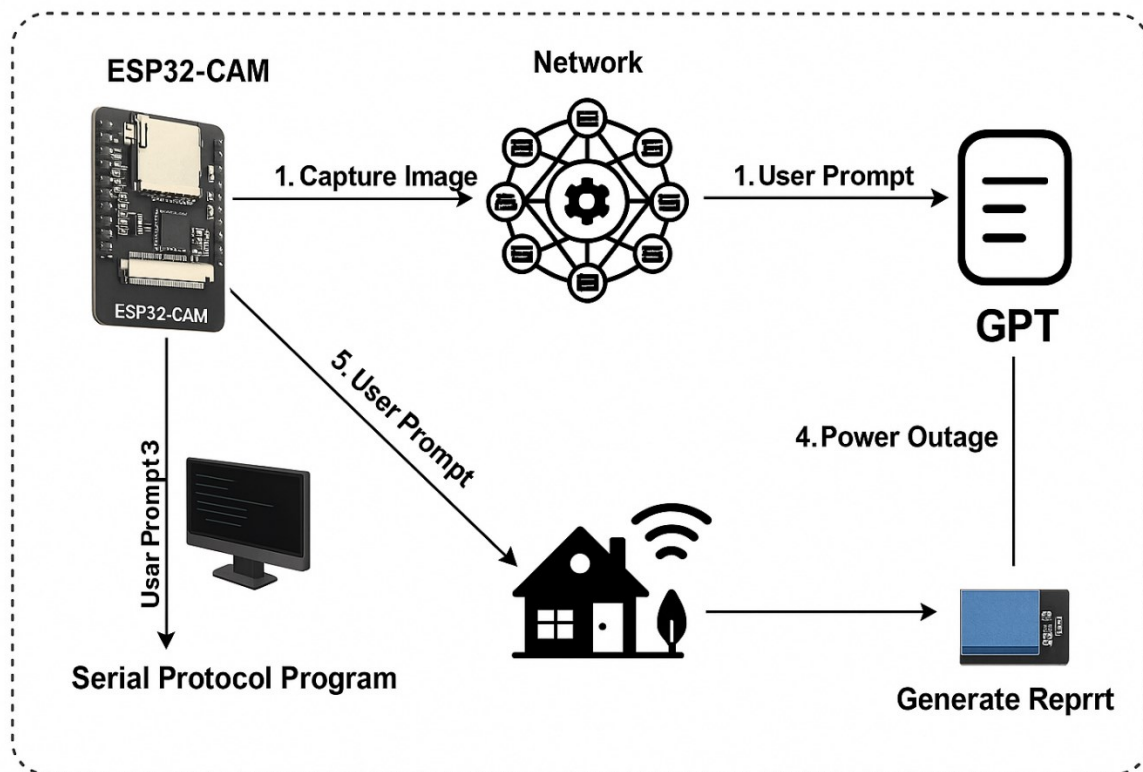
شرح و راهنمای انجام پروژه

## تعریف مسئله

در بسیاری از کاربردهای خانه هوشمند و نظارت محیطی، نیاز است که از محیط با فواصل زمانی مشخص عکس برداری شود و تغییرات به صورت هوشمند تحلیل شود. این کار می تواند در سناریوهایی مثل بررسی تغییرات وسایل، حضور انسان یا حیوان، تغییر روشنایی یا وضعیت امنیتی مورد استفاده قرار گیرد.

در این پروژه، کاربر می تواند از طریق مرورگر و به کمک آدرس دهی HTTP، فاصله زمانی بین دو عکس را مشخص کند (مثلاً ۵ ثانیه یا ۲ دقیقه)، و سیستم پس از دریافت این فرمان:

۱. عکس اول را ثبت می کند.
۲. بعد از گذشت زمان تعیین شده، عکس دوم را ثبت می کند.
۳. هر دو عکس را به همراه پرامپت مناسب برای تحلیل تفاوت ها به GPT ارسال می کند.
۴. پاسخ GPT را به صورت گزارش متنی نمایش می دهد.



## اهداف پروژه (Project Goals)

هدف این پروژه طراحی یک سیستم سبک و هوشمند برای پایش محیط از طریق ثبت دو تصویر با فاصله زمانی دلخواه و تحلیل تفاوت‌ها به کمک GPT است. این سیستم با استفاده از ESP32-CAM و دستورات HTTP از سمت کاربر، دو تصویر از محیط را می‌گیرد و آنها را به مدل هوش مصنوعی ارسال می‌کند تا تفاوت‌ها تحلیل شود.

### اهداف جزئی:

- ارائه یک رابط کاربری ساده با استفاده از آدرس دهی مرورگر (مثلاً `/capture?delay=10`)
- ثبت خودکار دو عکس با فاصله مشخص شده توسط کاربر
- ارسال تصاویر به GPT و دریافت تحلیل متنی
- نمایش گزارش کامل در سریال مانیتور

## نمای کلی سیستم (System Overview)

این سامانه یک سامانه‌ی هوشمند برای پایش تصویری محیط است که با بهره‌گیری از ترکیب سخت‌افزار (ESP32-CAM) و GPT، وظیفه‌ی ثبت، تحلیل و گزارش تغییرات محیط را برعهده دارد. این سیستم برای کاربر امکان‌پذیر می‌سازد که تنها با ارسال یک دستور از طریق مرورگر، دو تصویر متوالی از محیط با فاصله‌ی زمانی دلخواه گرفته شود و تحلیل هوشمند تفاوت‌های آن‌ها به‌صورت گزارشی ارائه گردد.

در این پروژه از یک ماژول اصلی استفاده شده است:

- **ماژول ESP32-CAM** برای گرفتن دو تصویر متوالی از محیط با فاصله زمانی مشخص شده توسط کاربر، و ارسال آن‌ها برای تحلیل.

روند عملکرد سیستم به‌صورت مرحله‌ای به شرح زیر است:

۱. کاربر از طریق مرورگر، آدرس HTTP مانند زیر را فراخوانی می‌کند:

`http://<IP_ADDRESS>/capture?delay=15`

این دستور به سیستم می‌گوید که باید دو عکس با فاصله‌ی ۱۵ ثانیه گرفته شوند.

۲. ماژول ESP32-CAM پس از دریافت این درخواست:

- بلافاصله تصویر اول را ثبت می‌کند.

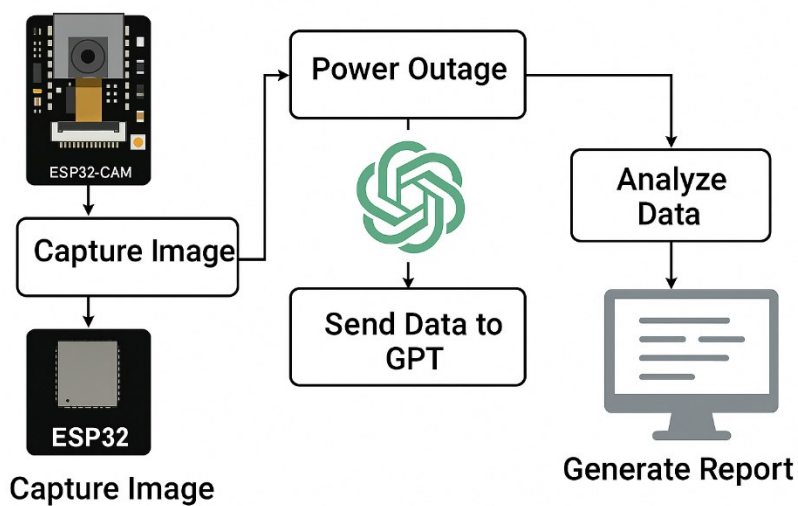
- پس از سپری شدن ۱۵ ثانیه، تصویر دوم را می‌گیرد.

۳. هر دو تصویر، همراه با یک پرامپت متنی مشخص، به مدل GPT ارسال می‌شوند.

۴. GPT تفاوت میان دو تصویر را تحلیل کرده و گزارشی متنی شامل تشخیص تغییرات، هشدارها و توصیه‌ها ارائه می‌دهد.

۵. پاسخ دریافتی به‌صورت شفاف در Serial Monitor نمایش داده می‌شود و قابل استفاده برای کاربر یا تحلیل‌های بعدی است.





## ابزارها و ماژول‌های مورد استفاده

ابزار / ماژول	کاربرد در پروژه
ESP32	ماژول اصلی برای کنترل، پردازش و ارسال اطلاعات به GPT
ESP32-CAM	گرفتن عکس از محیط
Wi-Fi	ارتباط اینترنتی برای ارسال داده به GPT API
سریال مانیتور (Serial Monitor)	نمایش گزارش نهایی برای کاربر
OpenAI GPT API	تحلیل متن و تصویر
تایمر نرم‌افزاری / زمان سیستم	تعیین لحظه عکس‌برداری
IDE Arduino	محیط برنامه‌نویسی و آپلود کدها
کابل USB و پروگرامر FTDI	جهت آپلود برنامه روی ESP32-CAM و برقراری ارتباط سریال

## شرح ماژول‌ها و اجزای پروژه

### ۱. ESP32-CAM

- **وظیفه اصلی:** ثبت دو تصویر متوالی از محیط با فاصله زمانی مشخص شده توسط کاربر از طریق درخواست HTTP.
- **زمان عملکرد:**
  - تصویر اول فوراً پس از دریافت درخواست `capture?delay=N` ثبت می‌شود.
  - تصویر دوم پس از سپری شدن N ثانیه گرفته می‌شود.
- **نقش کلیدی:** منبع اصلی داده‌ی تصویری برای تحلیل تغییرات محیط، مانند تغییر وضعیت وسایل، حضور افراد، تغییر روشنایی و غیره.

### ۲. Wi-Fi Module

- **وظیفه اصلی:** اتصال به اینترنت برای ارسال تصاویر به مدل GPT.
- **زمان عملکرد:** بلافاصله پس از بوت، ماژول ESP32-CAM به شبکه Wi-Fi از پیش تنظیم شده متصل می‌شود و تا پایان عملیات ارسال و دریافت پاسخ، ارتباط برقرار می‌ماند.

### ۳. Serial Monitor

- **وظیفه اصلی:** نمایش پیام‌های سیستمی، مراحل پیشرفت کار، و پاسخ تحلیل شده‌ی دریافت شده از GPT.
- **نقش آموزشی:** به کاربر امکان می‌دهد فرآیند کامل را گام به گام دنبال کند، از ثبت عکس‌ها تا ارسال و تحلیل توسط مدل هوش مصنوعی.

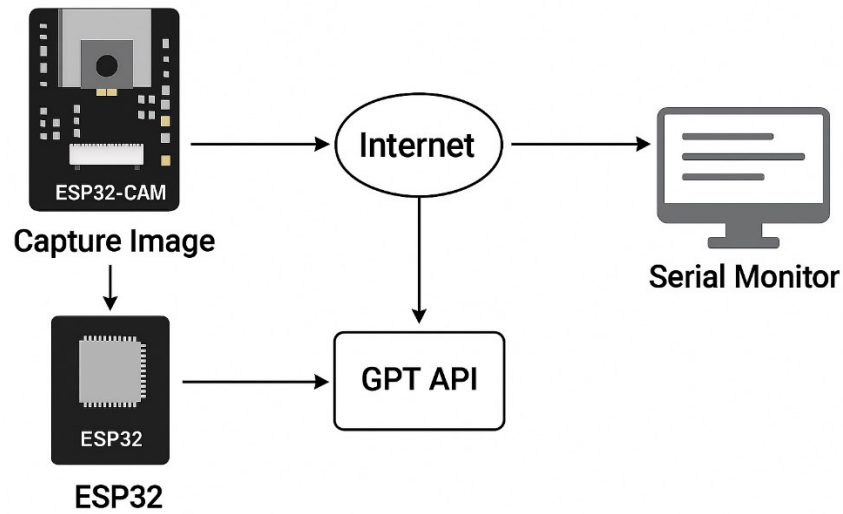
### ۴. HTTP Web Server

- **وظیفه:** دریافت دستورات از طریق مرورگر کاربر، مثلاً:

`http://<IP_ADDRESS>/capture?delay=15`

این مسیر باعث می‌شود سیستم با تأخیر ۱۵ ثانیه، دو عکس ثبت کرده و تحلیل آن‌ها را از GPT دریافت کند.

## نحوه عملکرد گام به گام سیستم



### مرحله ۱: دریافت فرمان از کاربر

کاربر از طریق مرورگر خود، آدرس مشخصی را فراخوانی می‌کند. مثلاً:

```
http://<IP_ADDRESS>/capture?delay=15
```

در این درخواست، کاربر مشخص می‌کند که فاصله بین دو عکس چند ثانیه باشد

### مرحله ۲: ثبت تصویر اول

بلافاصله پس از دریافت درخواست، ماژول ESP32-CAM یک تصویر از محیط ثبت می‌کند. این تصویر ممکن است شامل وضعیت فعلی وسایل، نور، حضور افراد یا تغییرات محیطی باشد.

### مرحله ۳: انتظار برای گذشت زمان مشخص شده

سیستم به مدت مشخص شده در پارامتر delay (مثلاً ۱۵ ثانیه) صبر می‌کند.

### مرحله ۴: ثبت تصویر دوم

پس از گذشت زمان تعیین شده، ماژول ESP32-CAM تصویر دوم را از همان زاویه محیط ثبت می‌کند.

## مرحله ۵: ارسال تصاویر به GPT

هر دو تصویر، به همراه یک پرامپت متنی مشخص، برای مدل GPT ارسال می شوند. این پرامپت از مدل می خواهد تفاوت های میان دو تصویر را تحلیل کرده و گزارشی از تغییرات ارائه دهد.

## مرحله ۶: دریافت پاسخ تحلیلی از GPT

پاسخ GPT ممکن است شامل مواردی مانند تغییر وضعیت وسایل، روشن یا خاموش شدن منابع نوری، جابجایی اشیاء یا حضور/عدم حضور افراد در محیط باشد.

## مرحله ۷: نمایش گزارش در Serial Monitor

پاسخ نهایی دریافت شده از GPT در **Serial Monitor** نمایش داده می شود. گزارش شامل موارد زیر خواهد بود:

- تحلیل تغییرات میان دو تصویر
- هشدارها یا نکات امنیتی/کاربردی
- توصیه های GPT بر اساس تحلیل تصویری

## ۷. نمونه پرامپت و خروجی (GPT (Prompt & Response)

در این پروژه، یکی از مهم‌ترین نقاط قوت و بخش‌های خلاقانه، استفاده از GPT به‌عنوان یک «تحلیل‌گر هوشمند تصویری و زبانی» است. به‌جای تفسیر دستی داده‌ها، سیستم با استفاده از مدل زبان طبیعی GPT، گزارشی روشن و انسانی از تغییرات محیطی بین دو تصویر ارائه می‌دهد.

برای این منظور، باید پرامپتی (Prompt) دقیق و هدفمند طراحی شود تا از GPT بخواهیم تغییرات قابل توجه بین دو عکس را شناسایی و تفسیر کند. طراحی صحیح پرامپت نقش تعیین‌کننده‌ای در کیفیت و کاربردی‌پذیری پاسخ نهایی دارد.

۷,۱ مثال ساده از یک پرامپت (Prompt)

The following two images were captured in the same room with a 15-second delay.

Please analyze and describe:

- Any noticeable changes in objects or lighting.
- If any devices appear to have turned ON or OFF.
- Any signs of human or animal presence or movement.
- Any suspicious or unexpected changes.

Please respond with a clear and concise summary.

۷,۲ نمونه پاسخ تولیدشده توسط (GPT (Response)

Analysis Report:

- The desk lamp was OFF in the first image but is ON in the second image.
- A chair has been slightly moved, indicating possible human interaction.
- Brightness level increased slightly, suggesting a new light source or sunlight change.
- No unexpected or suspicious objects detected.

Recommended Action: If the lamp activation was not scheduled, verify manually.

## بخش امتیازی: مهارت در طراحی پرامپت (Prompt Engineering)

برای کسب امتیاز ویژه، هر دانشجو باید حداقل سه پرامپت مختلف طراحی کرده و بررسی کند که کدام پرامپت خروجی دقیق‌تر، واضح‌تر و کاربردی‌تری از GPT دریافت می‌کند. این مهارت تحت عنوان "Prompt Engineering" شناخته می‌شود و یکی از توانمندی‌های مهم در تعامل با مدل‌های زبانی هوشمند است.

### ایده تمرینی:

فرض کنید می‌خواهید به جای دریافت یک گزارش طولانی، فقط یک هشدار کوتاه و فوری مانند پیام موبایل دریافت کنید. پرامپتی طراحی کنید که تنها این جمله را از GPT دریافت کند:

"Warning: Printer failed to restart after outage. Manual check required."

### وظایف دانشجو:

- طراحی پرامپت کوتاه برای هشدار فوری
- طراحی پرامپت رسمی برای گزارش کامل تحلیلی
- طراحی پرامپت دوستانه برای کاربران خانگی غیرتخصصی

دانشجو باید خروجی‌های هر پرامپت را ذخیره کرده، مقایسه نماید، و در گزارش نهایی پروژه خود ثبت کند. این بخش به صورت مجزا ارزیابی می‌شود و در صورت انجام کامل، امتیاز مثبت در بخش خلاقیت و مهارت هوش مصنوعی برای دانشجو در نظر گرفته خواهد شد. باید در این بخش سعی شود پرامپتی طراحی شود که بتواند تکمیل‌ترین گزارش و مقایسه را به کاربر نشان دهد.

### ۷,۳ وارد کردن پرامپت از طریق سریال مانیتور (روش اصلی و پیشنهادی)

در نسخه‌ی پایه‌ی پروژه PowerWitness ، به‌منظور ساده‌سازی فرآیند اجرای پروژه، ارتباط دانشجو با ماژول ESP32 تنها از طریق Serial Monitor انجام می‌شود. در این روش، نیازی به راه‌اندازی وب‌سرور، طراحی فرم HTML یا استفاده از مرورگر نیست.

#### نحوه کار:

پس از راه‌اندازی سیستم، ESP32 یک پیام اولیه در سریال مانیتور چاپ می‌کند که از دانشجو می‌خواهد پرامپت موردنظر خود را وارد کند. این پرامپت می‌تواند هرگونه پرسشی در رابطه با تحلیل تصاویر گرفته‌شده باشد.

#### نمونه مراحل:

#### شروع پروژه:

ماژول ESP32 بوت می‌شود و در Serial Monitor پیام زیر نمایش داده می‌شود:

Please enter your GPT prompt:

### ورود پرامپت:

کاربر پرامپت خود را تایپ می کند و کلید Enter را می زند. برای مثال:

Please compare the two images and tell me which devices were ON before the outage and are now OFF.

### ارسال به GPT :

ماژول ESP32 این متن را همراه با تصویر گرفته شده (که به فرمت base64 تبدیل شده است) به API سرور GPT ارسال می کند.

### نمایش پاسخ:

پس از دریافت پاسخ از GPT ، ماژول پاسخ را به صورت مستقیم در Serial Monitor نمایش می دهد. برای مثال:

GPT Response:

The printer and fan were ON before the outage but failed to restart.

در این روش، تمام تعامل با سیستم از طریق Serial Monitor انجام می شود و ماژول ESP32 در نقش یک Web Client ظاهر می شود که مستقیماً به GPT متصل شده و پاسخ را دریافت می کند. نیازی به پیاده سازی Web Server یا فرم HTML وجود ندارد.

### ۷,۴ اتصال به مدل های زبانی از طریق سرویس AvalAI

برای پیاده سازی بخش ارتباط با مدل GPT در این پروژه، از سرویس ایرانی AvalAI استفاده شده است. این پلتفرم با هزینه ای مناسب و ساختار ساده، امکان دسترسی به مدل های زبانی قدرتمند مانند GPT-3.5 را فراهم می کند و به ویژه برای پروژه های آموزشی و دانشگاهی گزینه ای مناسب به شمار می آید.



## ۱. ثبت نام و دریافت کلید API از AvalAI

مراحل راه اندازی حساب در این پلتفرم به صورت زیر است:

۱. ورود به سایت [ir.avalai](https://ir.avalai.ir)

۲. ثبت نام از طریق بخش <<لتفرم توسعه دهندگان>>

۳. شارژ حساب در بخش مالی.

۴. ساخت کلید API در بخش <<کلیدهای API با انتخاب نام دلخواه>>

۵. یادداشت و نگهداری کلید در محل امن (کلید فقط یکبار نمایش داده می شود).

## ۲. اتصال به API در زبان Python

برای ارسال درخواست به مدل GPT از طریق AvalAI، می توان از کتابخانه های آماده در پایتون استفاده کرد. مراحل زیر را دنبال کنید:

نصب کتابخانه های لازم:

```
pip install -U openai  
pip install -U langchain  
pip install -U langchain_openai
```

کد نمونه برای ارسال درخواست به GPT :

```
from langchain_openai import ChatOpenAI  
  
llm = ChatOpenAI(  
    model="gpt-3.5-turbo",  
    base_url="https://api.avalai.ir/v1",  
    api_key="your_api_key_here"  
)  
result = llm.invoke("your prompt")  
print(result.content)
```

در این کد، کلید API خود را در قسمت `api_key` قرار دهید و پیام دلخواه را به تابع `invoke()` ارسال کنید. پاسخ مدل به صورت متنی در متغیر `result.content` دریافت خواهد شد.

نمونه خروجی در سریال مانیتور:

System ready. Waiting for HTTP request...

Client connected: 192.168.1.7  
Capture request received.  
Delay parameter: 12 seconds

[16:43:10] Capturing first image...  
Image #1 captured successfully.

Waiting for 12 seconds...  
[16:43:22] Capturing second image...  
Image #2 captured successfully.

Converting images to Base64...  
Encoding complete.

Preparing prompt...  
Sending data to GPT via AwaAI API...

Waiting for GPT response...

--- GPT Response Received ---

GPT Response:

-----  
Analysis Summary:

- The desk lamp was OFF in Image 1 and turned ON in Image 2.
- A shadow near the chair has shifted, possibly indicating movement.
- Light intensity increased, suggesting either a window opened or another light source activated.

Recommended Actions:

- If the lamp activation was unintentional, verify automation settings.
- No anomalies detected beyond expected environmental change.