



# پروژه درسی

درس مبانی اینترنت اشیا

نیم سال دوم ۱۴۰۴-۱۴۰۳

هدف اصلی این پروژه، به کارگیری مفاهیم کلیدی اینترنت اشیا صنعتی (IIoT) در یک سناریوی عملی است. شما یک کارخانه شبیه سازی شده که از طریق پروتکل استاندارد صنعتی **OPC-UA** داده های خود را به اشتراک می گذارد را به پلتفرم اینترنت اشیا **ThingsBoard** متصل خواهید کرد. در این مسیر، شما با استفاده از **ThingsBoard IoT Gateway**، داده های تولید شده ماشین آلات را جمع آوری کرده، وضعیت آن ها را پایش می کنید و با ارسال فرامین از راه دور (RPC)، خط تولید را کنترل می کنید. در نهایت، با ساخت یک داشبورد مدیریتی جامع، یک کارخانه متصل (Connected Factory) را پیاده سازی می کنید که از طریق وب و اپلیکیشن موبایل قابل کنترل و نظارت است. این پروژه به شما کمک می کند تا درک بهتری از چگونگی یکپارچه سازی سیستم های اتوماسیون صنعتی (OT) با پلتفرم های فناوری اطلاعات (IT) پیدا کنید.

## دانش و ابزارهای پیش نیاز

- پیش از شروع پروژه، اطمینان حاصل کنید که بر مفاهیم و ابزارهای زیر مسلط هستید.
  - **مفاهیم اصلی ThingsBoard** (که در تمرین ۳ و ۴ مورد بررسی قرار گرفت)
    - موجودیت ها: **Customer, Device, Asset** و روابط بین آن ها.
    - انواع داده: **Telemetry, Client/Server/Shared Attributes**.
    - کنترل و منطق: **RPC (Remote Procedure Call), Rule Engine, Alarms**.
    - دیداری سازی: **Dashboards** و **Widgets**.
  - **پلتفرم ThingsBoard IoT Gateway**
    - معماری و هدف اصلی Gateway.
    - نحوه عملکرد **Connector** ها، به ویژه **OPC-UA Connector**.
    - فایل پیکربندی **opc-ua.json** و ساختار بخش های **server** و **mapping** (و یا واسط گرافیکی تنظیمات).
  - **پروتکل OPC-UA** (که در درس معرفی شده است)
    - مفاهیم پایه مانند **Server, Client, Node, NodeId** و فضای آدرس (Address Space).
    - توانایی کار با یک کلاینت گرافیکی مانند **Prosys OPC UA Browser** برای مشاهده و پیمایش **Node** ها.
  - **ابزارهای نرم افزاری**
    - **Docker** و **Docker Compose** و **Python venv** برای راه اندازی سرویس ها.
    - مفاهیم پایه **Git** برای دریافت کد شبیه ساز.

## مراحل انجام پروژه

پروژه در چند گام اصلی تعریف شده است. لطفاً مراحل را به ترتیب دنبال کنید.

## گام اول: راه اندازی و بررسی شبیه ساز کارخانه (factory-sim)

در این گام، شما محیط شبیه سازی کارخانه را راه اندازی کرده و از صحت عملکرد سرور OPC-UA آن اطمینان حاصل می کنید.

۱. دریافت کد: پروژه factory-sim را از آدرس آن<sup>۱</sup> دریافت کنید.

۲. راه اندازی با venv: با استفاده از Virtual Environment پایتونی، کتابخانه های وابستگی لازم را نصب و شبیه ساز و کلاینت کنترل کننده آن را طبق دستورالعمل اجرا کنید.

### ۳. بررسی سرور OPC-UA

- یک ابزار کلاینت OPC-UA مانند Prosys OPC UA Browser را نصب و اجرا کنید.
- به سرور شبیه سازی شده که در آدرس سرور کارخانه شبیه سازی شده در دسترس است، متصل شوید. تنظیمات امنیتی را بر روی None قرار دهید.
- درخت آبجکت ها را پیمایش کنید (\* -> MyObject -> Objects). در این بخش، متغیرهای مختلف مانند وضعیت و حسگر نزدیکی و ... را مشاهده خواهید کرد.
- ۴. تست کنترل: با اجرای فرامین از طریق کلاینت کنترل کننده، تغییر مقادیر را در Prosys Browser مشاهده کنید. برای مثال، دستور پخت را تغییر دهید و یا anomaly در مقادیر پیوسته یا گسسته ایجاد کنید.

## گام دوم: راه اندازی ThingsBoard و Gateway

همانند تمرین های گذشته، پلتفرم ThingsBoard و IoT Gateway را با استفاده از Docker Compose راه اندازی کنید. اطمینان حاصل کنید که هر دو سرویس به درستی اجرا شده و از طریق مرورگر به داشبورد ThingsBoard دسترسی دارید.

### گام سوم: پیکربندی OPC-UA Connector در ThingsBoard IoT Gateway

این بخش مهم ترین گام پروژه است. شما باید Gateway را طوری تنظیم کنید که به سرور OPC-UA شبیه ساز متصل شده و داده ها را به ThingsBoard ارسال کند<sup>۲</sup>.

۱. ویرایش پیکربندی: تنظیمات کانکتور OPC-UA در Gateway را پیدا کرده و باز کنید.
۲. تنظیمات اتصال سرور: بخش server را مطابق اطلاعات سرور شبیه سازی شده کارخانه در پایتون پیکربندی کنید. برای اتصال از داخل کانکتور Gateway به سرور OPC-UA که روی هاست شما اجرا می شود، از آدرس host.docker.internal استفاده کنید.

۳. تنظیمات Data mapping: به ازای هر داده متغیر و ثابت خط تولید یک time series و یا attribute به پلتفرم نگاشت دهید. در محیط prosys browser می توانید با کلیک راست مسیر هر متغیر را پیدا کرده و با قالب مشخص شده در مستندات و مثال کانکتور OPC UA تینگبورد در تنظیمات اضافه کنید. مثلاً:

```
{Root\\Objects\\.MyObject\\.ArmX}
```

می توانید به جای محیط گرافیکی با ویرایش فایل تنظیمات کانکتور هم کارهای فوق را انجام دهید.

## گام چهارم: مدل سازی کارخانه در ThingsBoard

۱. ایجاد Asset: یک Asset به نام Bakery Production Line ایجاد کنید.
۲. ایجاد Device ها: برای هر یک از ماشین آلات اصلی یک دستگاه (Device) در ThingsBoard بسازید. نام این دستگاه ها باید دقیقاً با deviceName که در پیکربندی تعریف کرده اید، مطابقت داشته باشد. البته گیت وی در صورت پیکربندی صحیح دستگاه ها را به طور خودکار اضافه می کند.

<sup>1</sup> <https://github.com/exalens/factory-sim>

<sup>2</sup> <https://thingsboard.io/docs/iot-gateway/config/opc-ua/>

۳. ایجاد روابط: دستگاه‌های ساخته شده را به Asset خط تولید (یعنی Bakery Production Line) متصل کنید تا سلسله مراتب منطقی کارخانه شکل گیرد.

### گام پنجم: ساخت داشبورد کنترل و پایش

- یک داشبورد جامع برای مدیریت خط تولید شیرینی‌پزی ایجاد کنید. این داشبورد باید حداقل شامل ویجت‌های زیر باشد.
- ویجت‌های نمایش تله‌متری (time series): برای نمایش پارامترهای گسسته و پیوسته خط تولید
  - ویجت‌های کنترلی (RPC): جهت تغییر دستور پخت و وضعیت فرایند با تنظیم کانکتور مناسب در گیت‌وی (اختیاری).
  - ویجت نمایش ویژگی‌ها: یک کارت (Card) برای نمایش ویژگی‌های ثابت دستگاه‌ها مانند شماره مدل.

### گام ششم (اختیاری/امتیازی): شناسایی ناهنجاری و ایجاد آلام

- در این بخش، یک ناهنجاری (anomaly) در سیستم تزریق کرده و آن را در ThingsBoard شناسایی می‌کنید.
۱. تزریق ناهنجاری: با استفاده از کلاینت کنترل‌کننده، یک ناهنجاری در مقادیر پیوسته ایجاد کنید.
  ۲. ایجاد Rule Chain: در Rule Engine پلتفرم ThingsBoard، یک قانون جدید تعریف کنید.
- این قانون باید داده‌های تله‌متری ورودی از دستگاه را دریافت کند.
  - با یک Script Filter Node، بررسی کنید که آیا تغییرات مقدار گزارش شده از یک آستانه مشخص بالاتر است یا خیر.
  - در صورت صحیح بودن شرط، پیام را به یک Create Alarm Node ارسال کنید.
  - در این نود، یک آلام با شدت Critical ایجاد کنید.
۳. نمایش آلام: یک ویجت آلام (Alarm Widget) به داشبورد خود اضافه کنید و نشان دهید که پس از تزریق ناهنجاری، آلام به‌درستی در داشبورد نمایش داده می‌شود.

### گزارش

- گزارش نهایی که توسط گروه‌ها تحویل داده می‌شود باید شامل موارد زیر باشد.
  - مقدمه و شرح معماری پیاده‌سازی شده.
  - توضیح کامل نگاشت‌های انجام شده در پیکربندی
  - تصاویری (Screenshots) از داشبورد نهایی در حالت‌های مختلف (در حال کار، متوقف، و در صورت انجام بخش امتیازی، در حالت آلام).
  - (اختیاری) تصویری از Rule Chain طراحی شده برای شناسایی آلام.
  - شرح چالش‌های احتمالی و راه‌حل‌های به کار رفته.
- ویدئوی نمایش عملکرد (حداکثر ۵ دقیقه): یک ویدئوی کوتاه که در آن به وضوح نشان داده می‌شود.
  - اتصال موفق با Prosys OPC UA Browser.
  - نمایش داشبورد ThingsBoard و داده‌های تله‌متری زنده.
  - (اختیاری) اجرای موفق یک دستور RPC (مثلاً کلیک روی دکمه "Start" و نمایش شروع به کار خط تولید).
  - (اختیاری) نمایش فرایند تزریق ناهنجاری و فعال شدن آلام در داشبورد.

## تحويل

در روز تحويل اعضاى گروه با به همراه داشتن يك نسخه از گزارش پروژه براى تحويل مراجعه مى كنند (و يا به صورت مجازى با آنها هماهنگ مى شود).

اعضاى گروه در ابتدا يك گزارش شفاهى کوتاه (در حد ۳-۴ دقيقه) در مورد پروژه ارائه مى كنند كه شامل نكات مهم، چالش ها، شيوه انجام كار و انتخاب پارامترها مى باشد.

پس از آن گروه سيستم در حال كار را نمايش خواهند داد و توضيحات لازم را ارائه خواهد نمود.

در مرحله بعد در صورتى كه گروه پياده سازى بخش هاى اختيارى را نيز انجام داده باشد، آن را نمايش مى دهند.

دقت كنيد كه وظيفه تك تك اعضاى گروه است كه كيفيت كار انجام شده و ميزان مشاركت خود را به هنگام تحويل اثبات كنند. در صورت سكوت هر يك از اعضا هنگام جلسه تحويل طبيعى است كه نمره اى به آنها تعلق نخواهد گرفت.

موفق باشيد

عطارزاده