

گزارش کار پروژه پایانی معماری نرم افزار

**موضوع Microservice extraction based on knowledge graph from monolithic applications**

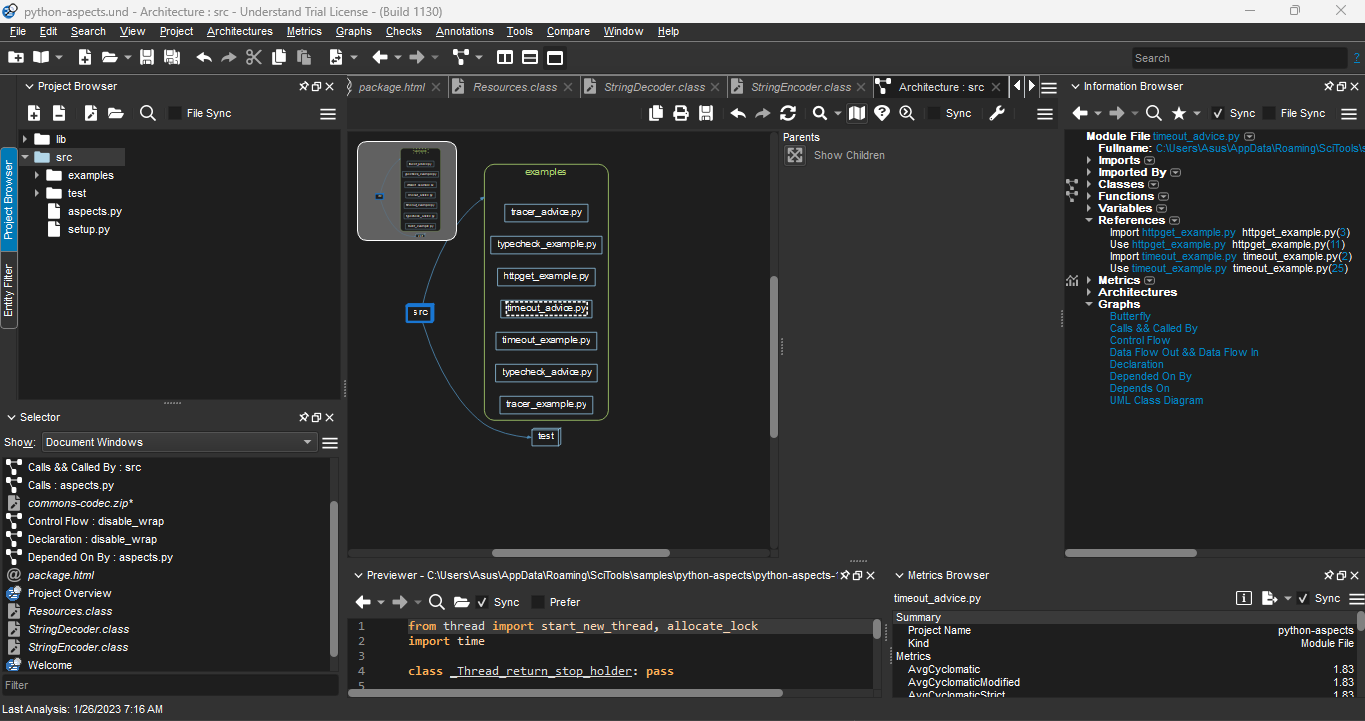
استاد دکتر اشتیانی

گرداورندگان

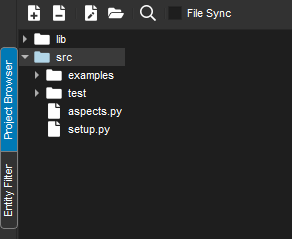
حاتم ابوالقاسمی ,حسین اله وردی, امیر اقابیکلو

مقدمه:

پروژه ای که در این مقاله با آن رو برو هستیم، برای کلاسبندی برنامه های مونولوتیک بر اساس موجودیت های داده، موجودیت فانکشن، موجودیت ماژول و موجودیت ریسورس است. این پروژه بر اساس استاندارد AKF است.

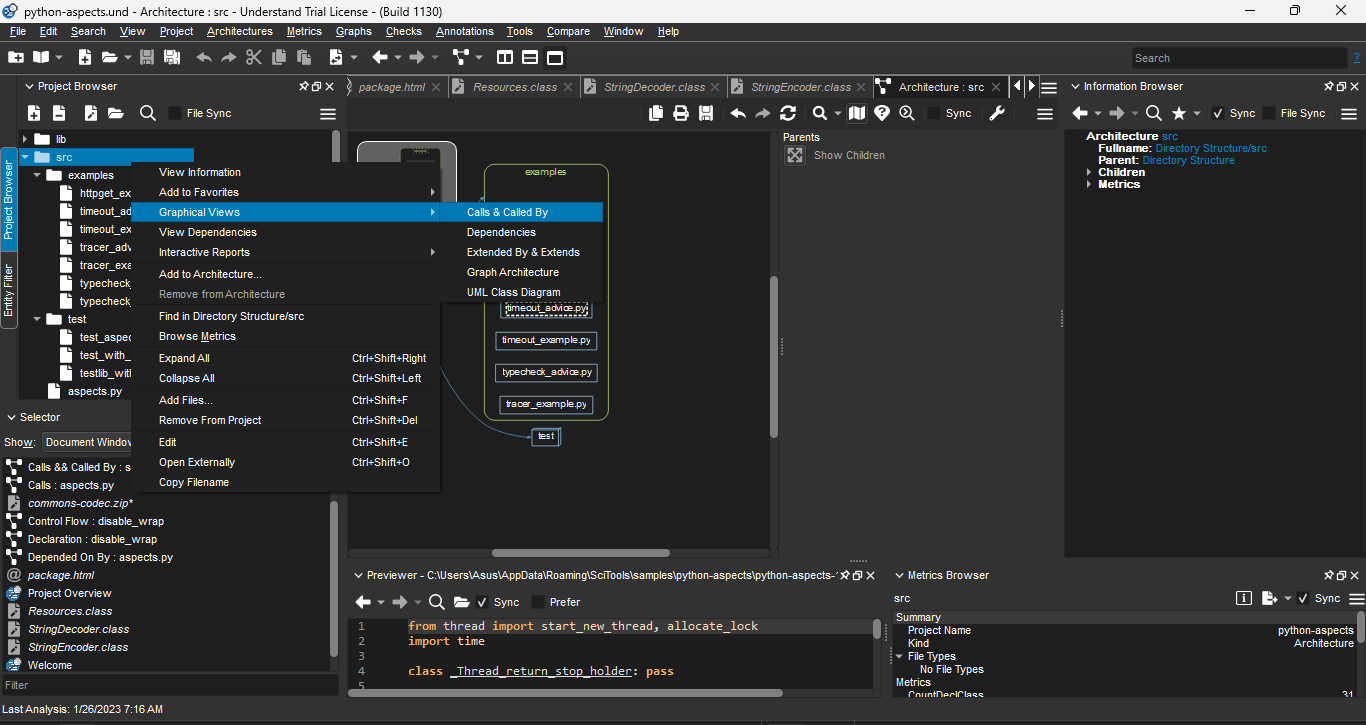


نرم افزاری است که به شما اجازه می دهد تا کنترل کدهای خودتان را در دست بگیرید و با visualization کدها و ابزارهایی که برای بررسی کردن آن ها ارائه می کند و برای تحلیل انواع زبان های برنامه نویسی بکار گرفته میشود

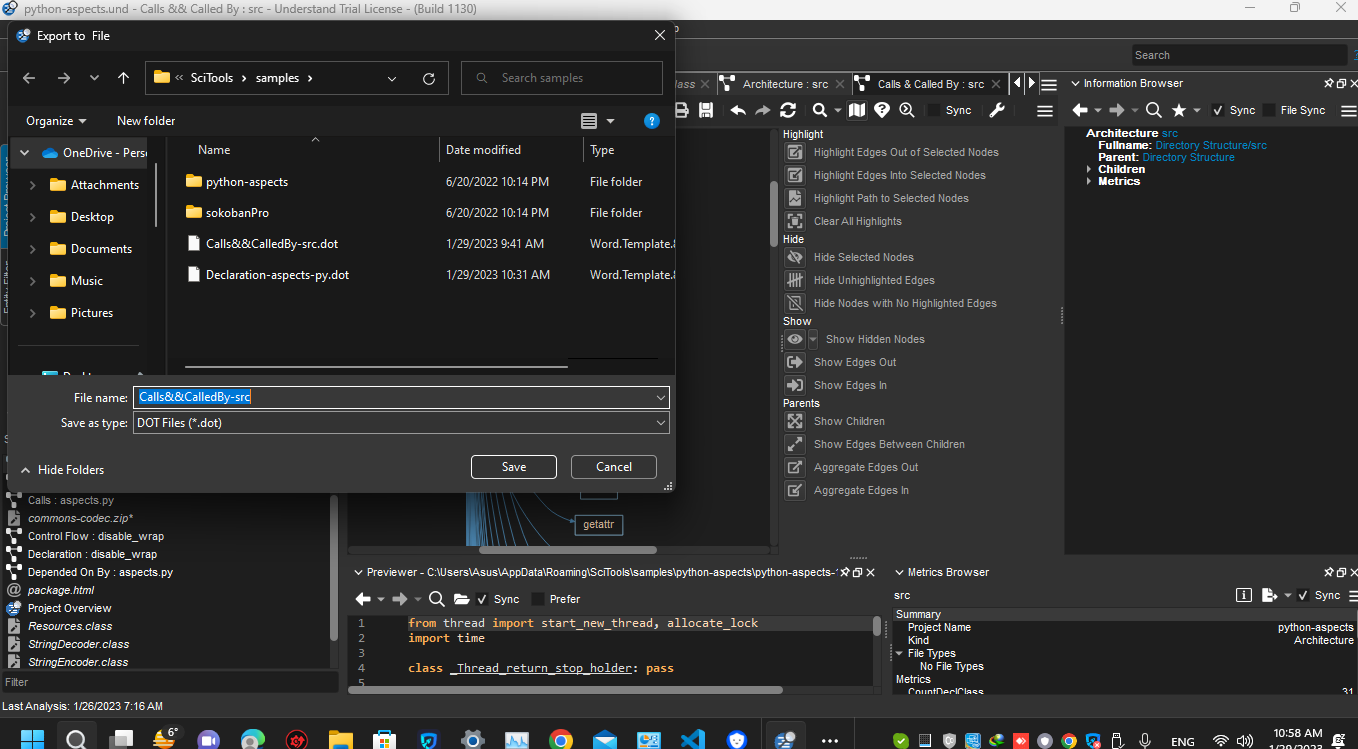


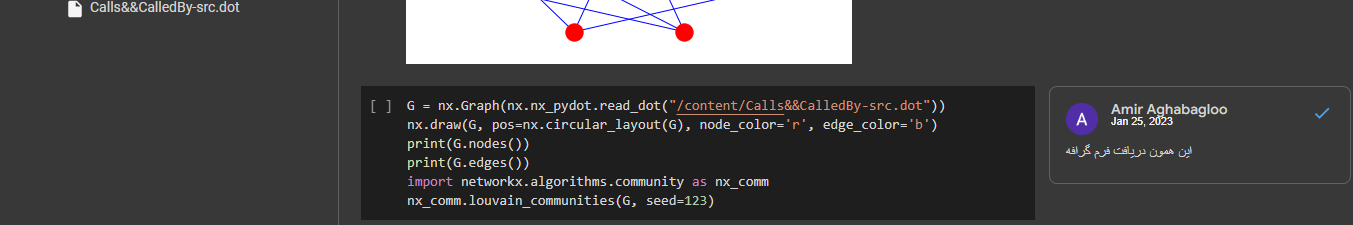
پس از بارگذاری پروژه خود در این نرم افزار و اجرای آن انواع تحلیل ها انجام میشود

اما ما نیاز به تحلیل گرافی calls and called by دارد



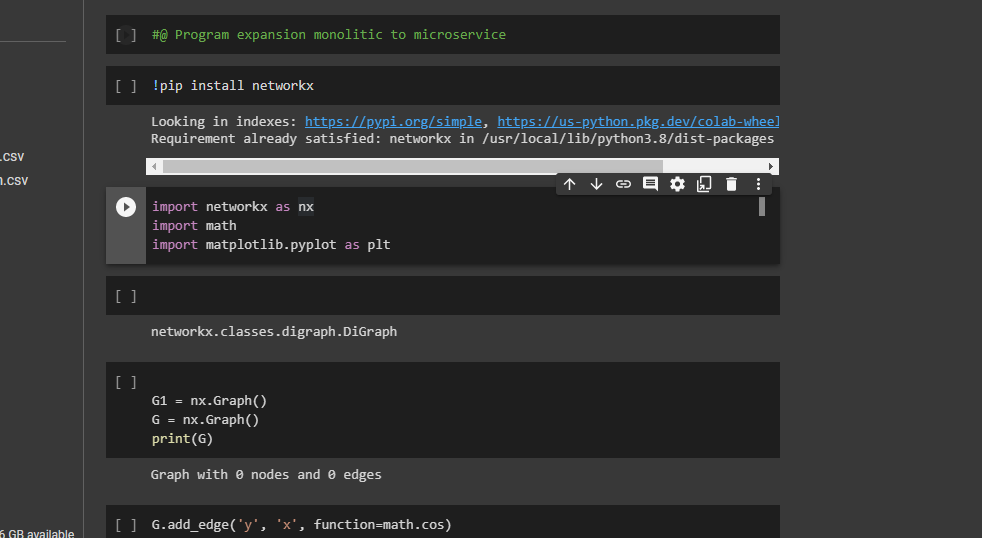
که اگر همه پروژه رو انتخاب کنیم تمام نود ها و یال ها را از انواع داده ها، فانکشن ها، ماژول ها و ریسورس ها به ما میدهد



این فایل گرافی را باید با فرمت dot ذخیره کنیم

ما با پایتون برنامه را نوشتیم

در پایتون کتابخانه networkx برای کار با گراف وجود دارد



و الگوریتم لواین برای کلاس بندی در این کتابخانه وجود دارد

طبق برنامه زیر کتابخانه های مورد نیاز برای نمایش گراف و دریافت فایل dot اقدام میکنیم

سپس فایل دریافتی را به شی ساخته شده از گراف بدون جهت با وزن میدهیم

و این شئ را به الگوریتم لواین میدهیم تا کلاس بندی ها را انجام دهد

این کار را در حلقه ای می اندازیم تا به استاندارد AKF برسد

الگوریتم AKF چیست؟

استراتژی اصلی AKF برای تجزیه­ ی مایکروسرویس تو این مقاله به این صورت است:

1. از منظر دیتا، متد سنتی ماژولار سازی نرم افزار، اجازه­ ی جدایی یه entity دیتا رو توی یه سیستم نرم افزاری مونولیتیک نمیدهد. ­با تکامل سیستم و افزایش حجم داده، عملیات­ های داده ­ای تبدیل به یه گلوگاه و باریک راه اجرایی برای کل سیستم میشوند

2. از جنبه­ ی کارکردی، ماژول­ ها رو بر اساس داده­ ای که بر روی آن است در حال انجام عملیات به مایکروسرویس­ های جداگونه تقسیم می­کنیم که این کار منتهی به تقسیم ماژول میشود. بنابراین موقع تقسیم مایکروسرویس­ ها، ماژول­ های سیستمی و توابع باید انقدری تقسیم شوند که به سطح کاربردی دیگه ای که از ماژول­ ها تو تقسیم بندی دانه بندی، باز تعریف شده تربرسند.

3. از نظر ریسورسی، بیزینیس ­های سیستمی میان بعد از تجزیه­ ی business function سیستمی مایکروسرویس ­ها رو فرم میدهند و بازسازی می­کنند. برعکس اپلیکیشن­ های مونولیتیک، وابستگی مایکروسرویس ­ها به سخت ­افزار خیلی پیچیده­ تره. تخصیص منابع سخت افزاری، تاثیر حیاتی ­ای روی اجرای مایکروسرویس ها دارد. بر همین اساس نیازه که وابستگی business function سیستم موقع استخراج مایکروسرویس به کار گرفته شود که موقع مقداردهی اولیه ی منابع مایکروسرویس بتواند باعث کمک به مناسب ­ترین تخصیص منابع سخت ­افزاری به مایکروسرویس ­های مشخصی شود. این متد میتوان در بهبود مقیاس پذیری افقی سیستم نرم افزاری کمک کند.

کتابخانه مربوط به گراف

import networkx as nx

کتابخانه مربوط به کشیدن گراف به صورت بصری

import math

import matplotlib.pyplot as plt

دستور خواندن فایل calls and called by با فرمت dot و ارسال آن به شی ساخته شده از کلاس بدون جهت وزن دار

G = nx.Graph(nx.nx\_pydot.read\_dot("/content/Calls&&CalledBy-src.dot"))

nx.draw(G, pos=nx.circular\_layout(G), node\_color='r', edge\_color='b')

print(G.nodes())

print(G.edges())

کتاب خانه مربوط به الگوریتم ها

import networkx.algorithms.community as nx\_comm

الگوریتم لواین و فراخوانی آن و انتقال شی مربوطه به این الگوریتم

nx\_comm.louvain\_communities(G, seed=123)