**Matplotlib**

Matplotlib เป็น open-source python library ที่ใช้สำหรับแสดงภาพนิ่ง ภาพเคลื่อนไหว หรือภาพแบบinteractive ด้วยภาษา Python ส่วนมากมักใช้ในการพล็อตกราฟ หรือภาพ 3D ที่สามารถเคลื่อนไหว / ขยาย / แก้ไขได้ โดยไฟล์ที่ใช้ matplotlib เขียนสามารถ export ได้หลาย format และสามารถนำไปปรับใช้ได้กับหลากหลายงาน เช่น เว็บไซต์, Python scripts, หรือ toolkits ที่ออกแบบ user interfaces เป็นต้น

Architecture Styles

Architecture ของ Matplotlib ถูกแบ่งออกเป็น 3 layers ซึ่งสามารถถูกมองเป็น Stack ได้เช่นกัน โดย layer ที่อยู่เหนือกว่าอีก layer จะเป็นคนที่รู้ว่าจะสื่อสารกับ layer ข้างล่างได้อย่างไรโดยที่ layer ข้างล่างไม่รู้ว่ามี layers อื่นๆอยู่เหนือตัวเอง โดย Layers ของ matplotlib เรียงจากล่างขึ้นบนได้ดังนี้

* Backend: ประกอบด้วย abstract interface classes

1. FigureCanvas – encapsulates สิ่งที่ผู้วาดจะวาดลง (เปรียบเสมือนกระดาษ)
2. Renderer – Class ที่ใช้ในการวาด (เปรียบเสมือนพู่กัน)
3. Event - ตรวจสอบ input ของผู้ใช้ผ่าน hardware ต่างๆเช่น เม้าส์ คีย์บอร์ด

* Artist: สมมติต่อจาก Backend Layer ว่า FigureCanvas เป็นกระดาษสำหรับวาดเขียน Artist

Layer คือ object ที่รู้ว่าจะต้องนำพู่กัน (Renderer) วาดรูปลงกระดาษอย่างไร

* Scripting (pyplot): เป็น Layer ที่นำตัว classes ออกมาให้เหล่าโปรแกรมเมอร์นำออกมาใช้ใน web application server, UI Application etc. โดยการ import

matplotlib.pyplot as \_\_

แหล่งที่มา: <https://www.aosabook.org/en/matplotlib.html>, <https://pypi.org/project/matplotlib/>

**Selenium**

Selenium คือ open source APIs ที่สามารถทำให้เราติดต่อกับ web browsers สมัยใหม่และเปลี่ยน actions นั้นเป็นแบบอัตโนมัติ โดย Selenium ไม่ใช่ tool แบบเดี่ยวๆแต่ประกอบไปด้วยหลายๆ tools เพื่อประกอบเป็น Selenium suite ซึ่งจะถูกสร้างขึ้นเมื่อ Selenium RC และ Web driver ถูกรวมเข้าด้วยกัน Selenium มีความสามารถรองรับได้หลายภาษาและหลาย Browsers ทำให้ได้รับความนิยมเป็นอย่างมาก

โดย Feature ต่างๆที่ Selenium ทำได้มีหลากหลาย และ feature ที่ได้รับความนิยมมากที่สุดคือ Dynamic Web Application หรือก็คือ Web Application ที่จะปรับเปลี่ยน content ไปตาม action ของผู้ใช้งาน

Architecture Styles

Diagram

Description automatically generated

Architecture ของ Selenium เป็นไปตามรูปข้างต้น โดย Architecture นั้นจะประกอบไปด้วย 4 Layers ซึ่งได้แก่ Selenium Client Library, JSON Wire Protocol, Browser Drivers และ Browsers

1. Selenium Client Library - ประกอบไปด้วยภาษาคอมพิวเตอร์มากมาย เช่น Java, Python, Ruby, C# และอื่นๆ หลังการที่ test cases ถูกส่งมา Selenium code จะถูกแปลงเป็น JSON
2. JSON Wire Protocol - เปรียบเสมือน medium สำหรับการส่งข้อมูลโดยใช้ RESTful API
3. Browser Driver - เนื่องจาก Selenium สามารถใช้งานได้ในหลากหลาย Browser และแต่ละ Browser ก็จะมี implementation เป็นของตัวเอง ซึ่งสิ่งนี้จะถูกเรียกว่า Browser Driver โดย Browser Driver จะรันโค้ดจากไฟล์ JSON ที่ถูกส่งมาจาก Browser จะส่ง Response กลับไปในรูปแบบ HTTP Response
4. Browser – Google Chrome, Mozilla Firefox etc.

แหล่งที่มา: <https://www.toolsqa.com/selenium-webdriver/selenium-webdriver-architecture/>, <https://www.tutorialspoint.com/what-is-the-selenium-web-driver-architecture>

**Jitsi**

Jitsi คือ open source projects ที่สามารถสร้างทำให้ผู้ใช้สร้างการ Video Conferencing ที่มีความปลอดภัย, ใช้งานง่าย และง่ายสำหรับการ host วิดิโอคอลด้วยตนเอง โดย Jitsu ถูกสร้างมาใช้ได้หลาย Platform ได้แก่ Windows, Linux, macOS, iOS และ Android

Architecture Design

Diagram

Description automatically generated

Jitsi มีการออกแบบด้วยวิธี Component ดังรูปข้างต้น ซึ่งประกอบไปด้วย Component ต่างๆดังต่อไปนี้

1. Jitsi Meet – WebRTC ที่สามารถใช้กับ JavaScript Web Application โดยใช้ Jitsi Videobridge สร้าง video conferences ที่คุณภาพของภาพสูงและปรับขนาดได้ ถูกสร้างโดยใช้ React และ React Native
2. Jitsi Videobridge (JVB) – เซิฟเวอร์ที่ถูกออกแบบมาใช้ในการส่ง-รับวิดีโอระหว่างผู้เข้าร่วมประชุม
3. Jitsi Conference Focus – Component ที่โฟกัสไปที่ฝั่งเซิฟเวอร์ ถูกใช้ในการจัดการสื่อ (ภาพ/วิดีโอ) ต่างๆ และทำหน้าที่เป็น Load Balancer ระหว่างผู้เข้าประชุมและระหว่าง Videobridge
4. Jitsi Gateway to SIP – Application ที่สามารถ grant access ให้กับลูกค้ามาเข้าร่วม Jitsi Meet Conference ที่ถูกสร้างขึ้นโดย JVB
5. Jitsi Broadcasting Infrastructure – Tools ที่ใช้สำหรับการ Live, บันทึกวิดีโอในการประชุม

แหล่งที่มา: <https://meetrix.io/blog/general/understanding-the-architecture-and-components-of-jitsi-meet.html>