

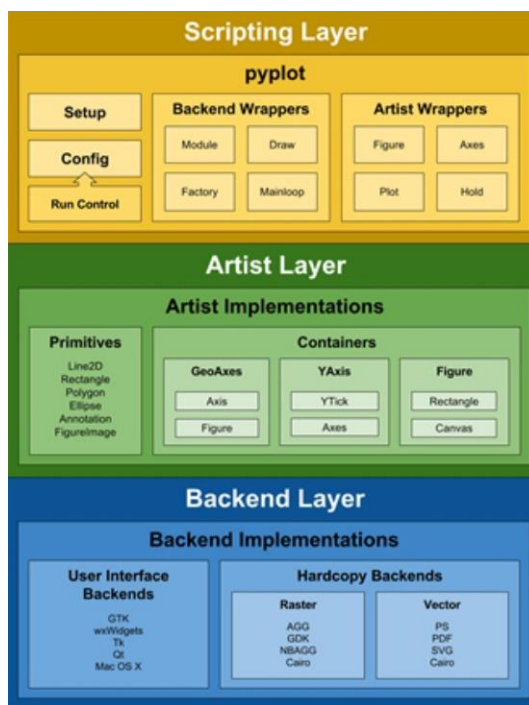
Architectural Patterns/Styles

Matplotlib

Purpose of the project

Matplotlib เป็น library สำหรับสร้าง static, animated, and interactive visualizations ด้วยภาษา Python โดยช่วยทำให้หลายอย่างทำได้ง่ายขึ้นในงานด้าน visualizations

Architectural patterns/styles



โดย Matplotlib ใช้ Architecture แบบ 3 Layer โดย

1. Backend Layer เป็นส่วนที่ใช้ทำงานที่ต้องใช้ทรัพยากรมากในการติดต่อ toolkits wxPython หรือภาษาที่ใช้วาดอย่าง PostScript ภายในเครื่อง ทำให้ใน Layer นี้ถือได้ว่ามีความซับซ้อนที่สุดใน Matplotlib Library โดยภายในประกอบไปด้วย
 1. FigureCanvas คือกรอบที่ให้ Render
 2. Renderer เป็น abstract base class สำหรับการวาดและ render
 3. Event เป็นการรับ input จากตัวผู้ใช้งาน ไม่ว่าจะเป็นการเลื่อนเมาส์หรือการคลิก
2. Artist Layer เป็นส่วนที่ใช้ render ลงบน Canvas ผู้ใช้สามารถปรับแต่งได้มากกว่า Scripting Layer

3. Scripting Layer เป็นชั้นบนสุดและเป็นชั้นที่ถูกใช้งานบ่อยที่สุด คำสั่งต่าง ๆ ใน Layer นี้ จะเป็นคำสั่งที่ถูกใช้งานง่าย

Quality attribute

- modifiability: เพราะ matplotlib มีการสร้าง GUI เพื่อให้ง่ายต่อการใช้งานและเป็น Opensource
 - Source of Stimulus: end user
 - Stimulus: ต้องการเพิ่ม, ลบ, แก้ไข, เปลี่ยนฟังก์ชัน
 - Artifacts: UI, Platform (windows, Unix), สภาพแวดล้อมของระบบ
 - Environment: เวลาทำงานปกติ (run time), compile time, ขณะการออกแบบหรือเขียนโปรแกรม
 - Responds: กำหนดจุดที่เปลี่ยนใน Architecture, เปลี่ยนโดยไม่มีผลกระทบกับฟังก์ชันอื่นๆ, ตรวจสอบ การเปลี่ยนแปลง, การนำไปใช้งาน
 - Respond Measures: ค่าใช้จ่าย, เวลา, ความซับซ้อนของแอปพลิเคชัน
- Performance: เกี่ยวกับเวลาในการตอบสนองการประมวลผลข้อมูลจำนวนมาก ๆ
 - Source of Stimulus: User
 - Stimulus: User command
 - Artifacts: System, ขั้นตอน generate grap
 - Environment: Normal mode, overload mode
 - Responds: เวลาในการทำงานให้เสร็จเมื่อเทียบกับ library อื่นๆ
 - Respond Measures: วัด Latency deadline, ปริมาณ หรือปริมาณที่ผ่านเข้าระบบได้
- Usability ผู้ใช้สามารถเลือก plot ได้หลากหลาย ซึ่งมีตัวเลือกเช่น scatter plot, Linear, histogram
 - Source of Stimulus: User
 - Stimulus: User ต้องการสร้างการ plot
 - Artifacts: Code
 - Environment: System configuration time
 - Responds: สร้างกราฟ plot ได้
 - Respond Measures: ความง่ายและความพึงพอใจของ User ที่ใช้เครื่องมือต่าง ๆ ในการ plot ข้อมูล

Ref

- <https://medium.datadriveninvestor.com/data-visualization-with-python-matplotlib-architecture-6b05af533569>
- <https://matplotlib.org/>
- [The Architecture of Open Source Applications \(Volume 2\): matplotlib \(aosabook.org\)](#)
- [\(PDF\) matplotlib -- A Portable Python Plotting Package \(researchgate.net\)](#)

Selenium WebDriver

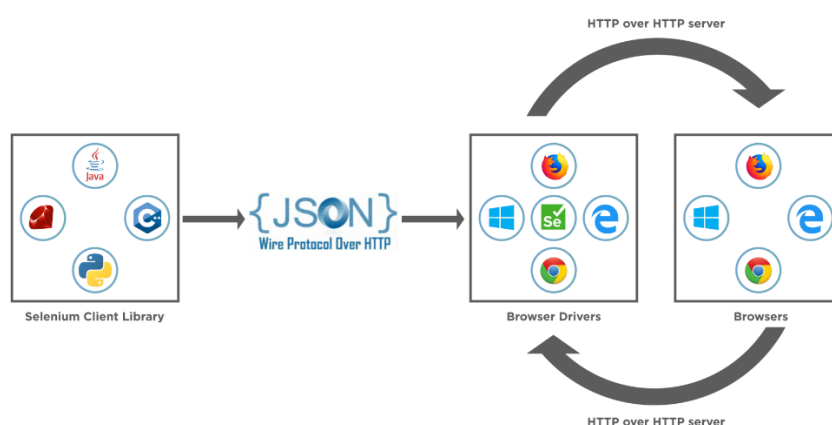
Purpose of the project

เป็นเครื่องมือที่ช่วยให้เราสามารถสร้างโปรแกรมในการทดสอบเว็บแอปพลิเคชันกับ Web browser ได้หลายตัว ซึ่งถือเป็นคุณสมบัติเด่นที่ดีกว่า Selenium IDE ซึ่งใช้งานได้เฉพาะ firefox เท่านั้น โดยจะมี Web Driver เป็นตัวกลางที่มีไลบรารีที่ช่วยในเราติดต่อกับ Web browser

โดยใช้การเขียนโปรแกรมในภาษาต่างๆ ที่เรารู้จักกัน ไม่ว่าจะเป็น JAVA, .Net (VB/C#), Ruby ติดต่อกับไลบรารีของ WebDriver เพื่อเข้าถึงคอนโทรลที่แสดงผ่าน Web browser ได้ ทำให้เราสามารถสร้างโปรแกรมการทดสอบได้หลากหลายมากขึ้น

Architectural patterns/styles

Selenium WebDriver มี Architecture แบบ REST



1. Selenium WebDriver Client Libraries / Language Bindings Software Testers สามารถเลือกภาษาที่ถนัด ซึ่ง WebDriver Architecture สามารถรองรับได้หลายภาษาไม่ว่าจะเป็น Java C#, Python, Ruby, PHP ทำให้คนที่มีความรู้ภาษาคอมพิวเตอร์ที่แตกต่างกันสามารถใช้งานได้

2. JSON WIRE PROTOCOL ใช้ในการติดต่อระหว่าง Browser และ Code ที่ใช้ Test เป็นเสมือน RESTful API
3. Browser Drivers ด้วยความที่ Selenium ต้องทำงานกับหลาย Browser ซึ่งมีการ implement ที่แตกต่างกันในแต่ละ Browser ทำให้การใช้ JSONWire ต้องทำงานร่วมกับ Browser Drivers
4. Browsers Selenium สามารถ run test ได้บน Browser ที่ติดตั้งบนอุปกรณ์แบบ Local เท่านั้น

Quality attribute

1. Compatibility เพราะสามารถทำงานได้แบบ cross-browser ได้โดยใช้ code ตัวเดิม ทำให้สามารถทดสอบได้ทุก Browser ตามที่ผู้ใช้จะเลือกใช้งานโดยไม่ต้องอาศัย middleware
 - Source of Stimulus: Tester, QA, Developer
 - Stimulus: ใช้ code ในการทดสอบ Web บนหลาย Browser
 - Artifact: Website ที่ต้องการทดสอบ
 - Environment: Browser ที่แตกต่างกันไม่ว่าจะเป็น Chrome Opera Firefox
 - Response: Test results
 - Response Measure: ความสามารถในการใช้ Test code เดิมกับ Browser ที่แตกต่าง
2. Performance เพราะสามารถทำงานโดยไม่ต้องใช้ middleware ในการติดต่อระหว่าง Server ไป Browser ซึ่ง WebDriver จะติดต่อกับ Browser โดยตรงจากการใช้ JSON Wire ทำให้สามารถติดต่อได้เร็วกว่า
 - Source of Stimulus: Tester, QA
 - Stimulus: ทดสอบระบบ
 - Artifact: ระบบ Web
 - Environment: Normal, Peak, Overload operation
 - Response: Test results
 - Response Measure: ระยะเวลาในการ response Test result
3. Fuctional Suitability เพราะ Selenium WebDriver สามารถทดสอบได้ถึงระดับ Element ต่าง ๆ บนเว็บ สามารถทำ user action ได้ ไม่ว่าจะเป็น Click, Type, Drag, Drop
 - Source of Stimulus: Tester, QA
 - Stimulus: ทดสอบ HTML element บนหน้าเว็บ
 - Artifact: Web Element
 - Environment: Normal, Load, Peak operation

- Response: Test results
- Response Measure: สามารถทดสอบ Element ได้ตามที่ต้องการและสามารถทดสอบได้เหมือนกับ User จริง ๆ

Ref

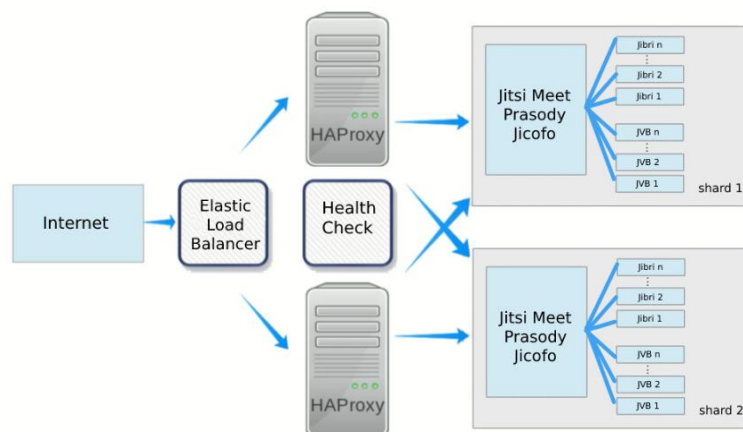
- <https://www.toolsqa.com/selenium-webdriver/selenium-webdriver-architecture/>

Jitsi

Purpose of the project

เป็นแอปพลิเคชันในการประชุมทางไกล ซึ่งสามารถเข้าใช้งานได้ทั้งบนหน้าบราวเซอร์เว็บไซต์ และดาวน์โหลดแอปพลิเคชันไปใช้บนมือถือได้ แอปพลิเคชันการประชุมนี้ทำงานด้วยระบบ End-to-End Encryption ทั้งนี้แอปพลิเคชันดังกล่าวสามารถแบ่งปันการจดโน้ตและแบ่งปันหน้าจอคอมพิวเตอร์ (Screen Sharing) ได้ด้วย

Architectural patterns/styles



- JVB คือ relay server และ scalable video router ที่ทำหน้าที่เป็น video conferencing
- Jibri เป็นการให้บริการในการบันทึกและ Streaming
- Jicofo เป็นตัวที่ทำหน้าที่ถ่ายทอดสัญญาณโดยผู้พูดเข้ามาในห้องและประสานการส่งคำเชิญ เปิด channel และทำ videobridge

Quality attribute

- Scalability โดย Jitsi สามารถเพิ่มหรือลดขนาดการประชุมได้ตามต้องการ
 - Source of Stimulus: End-User
 - Stimulus: ขยายขนาดการประชุม
 - Artifact: Jitsi conference
 - Environment: ก่อนเริ่มหรือระหว่างการประชุม
 - Response: จำนวน user ที่สามารถเข้าประชุม
 - Response Measure: Latency time, Video quality

- Compatibility โดย Jitsi สามารถใช้ได้หลายระบบปฏิบัติการไม่ว่าจะเป็น Windows, Linux, macOS, iOS และ Android.
 - Source of Stimulus: User
 - Stimulus: ใช้ Jitsi บน OS ที่แตกต่าง
 - Artifact: Jitsi conference
 - Environment: บน OS ที่ทำงานปกติ
 - Response: Video and Voice
 - Response Measure: สามารถใช้งาน Jitsi ได้ปกติ
- Performance โดย jitsi ต้องรองรับการใช้งานของ user ที่มาก ทำให้ต้องมีการทำ load balance และการแยก component เช่น JVB, Prosody, Jicofo และ Jigasy อีกทั้งยังมีการใช้ SCTP telegraf ที่ใช้ในการเชื่อมต่อเครือข่ายขนาดใหญ่ซึ่งช่วยลด Latency และ Packet Loss
 - Source of Stimulus: End-User
 - Stimulus: heavy user load
 - Artifact: Jitsi conference
 - Environment: Running Conference
 - Response: Video and voice quality
 - Response Measure: Resolution, Latency

Ref

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