

Matplotlib

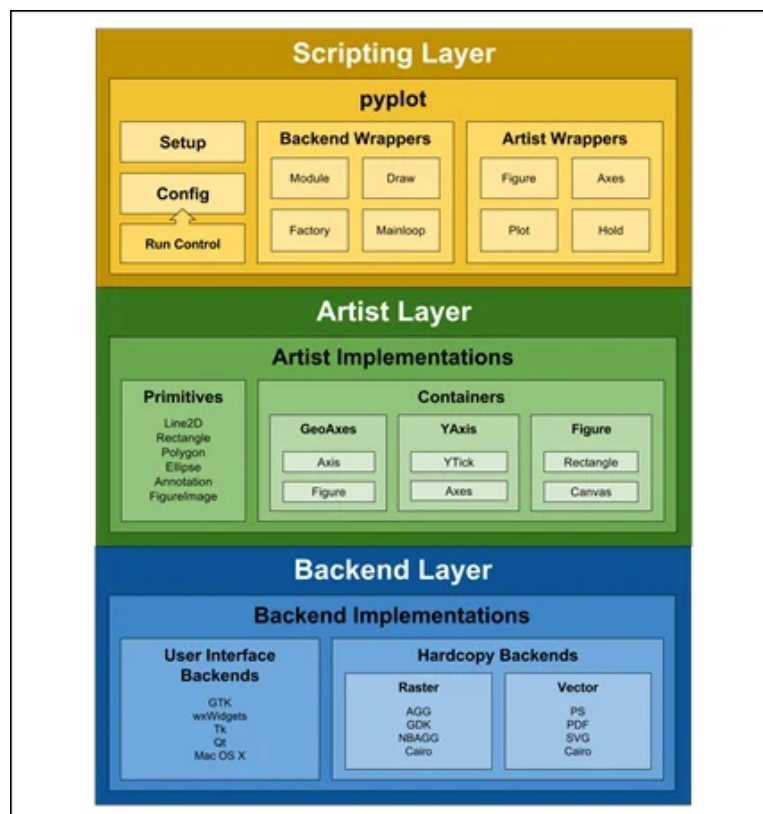
<https://www.aosabook.org/en/matplotlib.html>

<https://articles.adsabs.harvard.edu/pdf/2005ASPC..347...91B>

วัตถุประสงค์ของโปรเจก :

สำหรับการสร้างการแสดงผลภาพแบบคงที่ ภาพเคลื่อนไหว การนำเสนอข้อมูลเชิงกราฟิกใน Python สามารถ plot กราฟได้ง่าย, เอาต์พุต PostScript เพื่อรวมเข้ากับเอกสาร TEX , สามารถฝังกราฟิกในชุดเครื่องมืออินเทอร์เน็ตเพชผู้ใช้ที่คุณเลือก และ ปัจจุบันรองรับกราฟิกแบบโต้ตอบบนระบบปฏิบัติการเดสก์ท็อปหลักทั้งหมดโดยใช้ GTK+, Qt, Tk, FLTK, wxWidgets และชุดเครื่องมือ Cocoa มันสามารถเรียกแบบโต้ตอบจากเซลล์ Python แบบโต้ตอบเพื่อสร้างกราฟิกด้วยคำสั่งขั้นต่อนง่าย ๆ เช่น Mathematica, IDL หรือ MATLAB

Architecture ของ matplotlib :



สถาปัตยกรรมของ matplotlib มีสามชั้น แต่ละชั้นที่อยู่เหนือชั้นอื่นรู้วิธีพูดคุยกับชั้นที่อยู่ด้านล่าง แต่ชั้นล่างไม่ทราบชั้นที่อยู่เหนือชั้นนั้น สามชั้นจากล่างขึ้นบน คือ backend, artist, scripting

- Back-end layer: เตรียมการใช้งานของ 3 abstract interface classes
 - FigureCanvas : สรุปรูปแนวคิดของพื้นผิวที่จะวาดลงบน
 - Renderer: วาดรูปออกมา
 - Event : จัดการอินพุตของผู้ใช้เช่นเหตุการณ์แป้นพิมพ์และเมาส์

- Artist Layer : จะประกอบด้วย object ที่เรียกว่า Artist ที่รู้วิธีเอา Renderer (พู่กัน) และใส่หมึกลงบนผ้าใบ
ทุกสิ่งที่คุณเห็นใน Figure ของ matplotlib เป็นอินสแตนซ์ของ Artist ซึ่ง Artist object มี 2 ประเภท
 - Primitive: Line2D, Rectangle, Circle, text.
 - Composite: Axis, Axes, Tick, and figure
- Scripting Layer (pyplot):
ใช้ API ด้านบนทำงานโดยเฉพาะอย่างยิ่งสำหรับโปรแกรมเมอร์ และมักจะเป็นกระบวนการ
ทัศนการณ์เขียนโปรแกรมที่เหมาะสมเมื่อเขียนเว็บแอปพลิเคชันเซิร์ฟเวอร์ แอปพลิเคชัน
UI หรือบางทีอาจเป็นสคริปต์เพื่อแชร์กับนักพัฒนารายอื่น

Quality attribute scenarios:

- Usability : ใช้งานคำสั่งได้ง่าย
 - ❖ Source of stimulus: end user
 - ❖ stimulus : สามารถเรียนรู้ได้รวดเร็ว, ใช้งานได้ง่าย
 - ❖ Environment: run time, configuration time
 - ❖ Artifact: command line ได้รับการออกแบบตามอินเทอร์เน็ตเฟส MatLab
 - ❖ Respond : สามารถเข้าใจโค้ดได้ง่าย ใช้งานได้อย่างราบรื่น
 - ❖ Respond measure : ตรวจระยะเวลา, ความพึงพอใจของผู้ใช้
- Adaptability : รองรับการทำงานบนระบบปฏิบัติการที่หลากหลาย
 - ❖ Source of stimulus: user, developer
 - ❖ stimulus: สามารถทำงานได้ในระบบปฏิบัติการ
 - ❖ Environment: ระบบ
 - ❖ Artifact: GTK+, Qt, Tk, FLTK, wxWidgets และ ชุดเครื่องมือ Cocoa
 - ❖ Respond : สามารถทำงานได้ในระบบปฏิบัติการต่าง ๆ
 - ❖ Respond measure: ทำงานได้อย่างปกติในระบบปฏิบัติการต่าง ๆ
- Integrity : ป้องกันข้อมูลสูญหาย
 - ❖ Source of stimulus: end user
 - ❖ stimulus: สามารถเรียกดูข้อมูลได้
 - ❖ Environment: มีการบันทึกลงใน hardcopy
 - ❖ Artifact: GUI
 - ❖ Respond : สามารถเรียกคืนประวัติ
 - ❖ Respond measure: ความครบถ้วนสมบูรณ์ของข้อมูลที่เรียกคืน

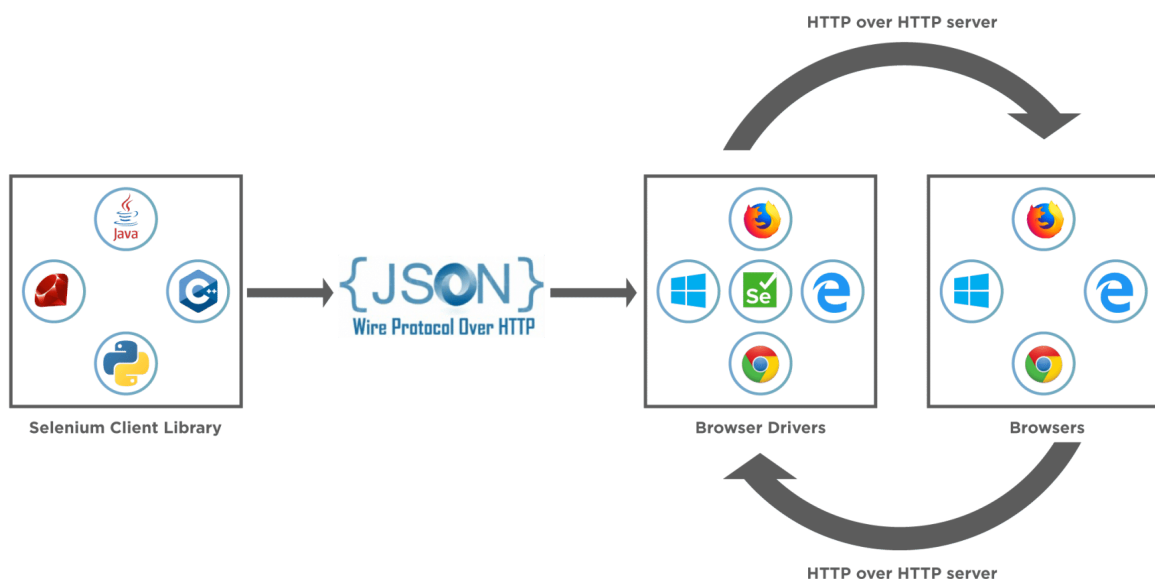
Selenium WebDriver

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

วัตถุประสงค์ของโปรเจก :

เป็นเฟรมเวิร์กการทำงานอัตโนมัติของเบราว์เซอร์ที่ยอมรับคำสั่ง และส่งไปยังเบราว์เซอร์ มีการใช้งานผ่านไคลเอนต์เฉพาะของเบราว์เซอร์ มันสื่อสารโดยตรงกับเบราว์เซอร์และควบคุมมัน จากนั้นจึงทำให้การดำเนินการ automate action ของ user จึงนิยมใช้ในการทำ UI automation Selenium WebDriver รองรับภาษาโปรแกรมต่างๆ เช่น Java, C#, PHP, Python, Perl, Ruby และ Javascript

Architecture ของ Selenium WebDriver :



Selenium WebDriver ไม่ใช่เครื่องมือทดสอบแบบสแตนด์อโลน ประกอบด้วยส่วนประกอบต่างๆ ที่จำเป็นสำหรับการทดสอบประกอบด้วยองค์ประกอบหลัก 4 ส่วน ได้แก่ Selenium Client Library, JSON WIRE PROTOCOL Over HTTP Client, Browser Drivers, Browsers

- Selenium WebDriver Client Libraries / Language Bindings :
รองรับหลายไลบรารีเช่น Java, Ruby, Python เป็นต้น Selenium Developers ได้พัฒนาการเชื่อมโยงภาษาเพื่อให้ Selenium รองรับหลายภาษา หากคุณต้องการทราบข้อมูลเพิ่มเติมเกี่ยวกับห้องสมุด โปรดดูเว็บไซต์อย่างเป็นทางการสำหรับไลบรารี
- JSON WIRE PROTOCOL Over HTTP Client :
JSON ย่อมาจาก JavaScript Object Notation ใช้ในการถ่ายโอนข้อมูลระหว่างเซิร์ฟเวอร์และไคลเอนต์บนเว็บ JSON Wire Protocol คือ REST API ที่ถ่ายโอนข้อมูล

ระหว่างเซิร์ฟเวอร์ HTTP BrowserDriver แต่ละตัว (เช่น FirefoxDriver, ChromeDriver เป็นต้น) มีเซิร์ฟเวอร์ HTTP ของตัวเอง

- Browser Drivers :
แต่ละเบราว์เซอร์มีไดรเวอร์เบราว์เซอร์แยกต่างหากไดรเวอร์เบราว์เซอร์สื่อสารกับเบราว์เซอร์ที่เกี่ยวข้องโดยไม่เปิดเผย logic ภายในของฟังก์ชันการทำงานของเบราว์เซอร์ เมื่อไดรเวอร์เบราว์เซอร์ได้รับคำสั่งใด ๆ คำสั่งนั้นจะถูกดำเนินการบนเบราว์เซอร์นั้น ๆ และการตอบสนองจะกลับไปในรูปแบบของการตอบสนอง HTTP
- Browser :
ซีลีเนียมจะทำการทดสอบบนเบราว์เซอร์ได้ก็ต่อเมื่อติดตั้งไว้ในเครื่อง ไม่ว่าจะในเครื่องlocal หรือ บนเครื่องเซิร์ฟเวอร์ ดังนั้นการติดตั้งเบราว์เซอร์จึงมีความจำเป็น

Quality attribute scenarios:

- flexibility : รองรับภาษาต่างๆ
 - ❖ Source of stimulus: Software Testers, user
 - ❖ stimulus : ต้องการเลือกภาษาที่ถนัด
 - ❖ Environment: ช่วงการ compile, ช่วงการออกแบบ, พัฒนาระบบ
 - ❖ Artifact: Selenium WebDriver Client Libraries
 - ❖ Respond : ผู้ทดสอบทำ automation ในความถนัดของตนมีภาษาต่างๆ เช่น Java, C#, Python, Ruby, PHP เป็นต้น
 - ❖ Respond measure: ความพึงพอใจ,ความสะดวก,ความพยายามใช้ระบบ
- Performance :WebDriverไม่ได้ขึ้นอยู่กับเซิร์ฟเวอร์มิดเดิลแวร์เพื่อสื่อสารกับเบราว์เซอร์
 - ❖ Source of stimulus: Software Testers, user
 - ❖ stimulus : ดำเนินการได้เร็วขึ้น
 - ❖ Environment: การสื่อสารกับเบราว์เซอร์
 - ❖ Artifact: JSON Wire
 - ❖ Respond: run time น้อยลง, ลดการใช้ทรัพยากร JSON มีน้ำหนักเบามาก ดังนั้นปริมาณการถ่ายโอนข้อมูลต่อการโทรจึงน้อยที่สุด
 - ❖ Respond measure: ความพึงพอใจ,การดำเนินการได้เร็วขึ้น

- Adaptability: ทำงานได้หลากหลายเบราว์เซอร์
 - ❖ Source of stimulus: Software Testers, user
 - ❖ stimulus: เรียกใช้งานเบราว์เซอร์ที่ต่างกัน
 - ❖ Environment: ช่วงการพัฒนาระบบ
 - ❖ Artifact: API
 - ❖ Respond :เรียก API โดยตรงโดยไม่ต้องใช้ซอฟต์แวร์ หรือ อุปกรณ์มิดเดิลแวร์
 - ❖ Respond measure: ความพึงพอใจ,การดำเนินการได้เร็วขึ้น,ความพยายามใช้

Joomla

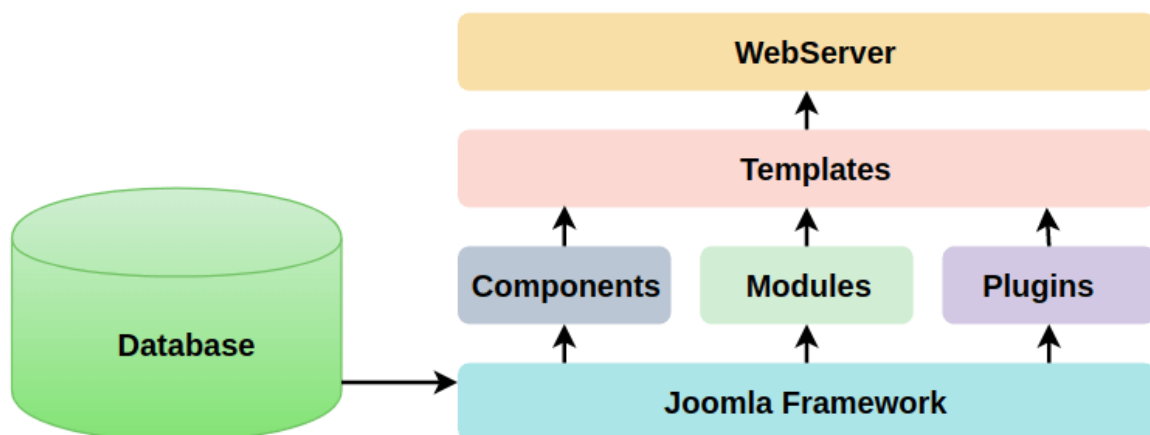
https://www.tutorialspoint.com/joomla/joomla_architecture.htm

<https://catalogimages.wiley.com/images/db/pdf/9781118474914.excerpt.pdf>

วัตถุประสงค์ของโปรเจก :

เฟรมเวิร์ก PHP (ชุดของไลบรารีซอฟต์แวร์/แพ็คเกจ) สำหรับการเขียนเว็บ และแอปพลิเคชัน ใช้ฐานข้อมูล MySQL เพื่อจัดเก็บข้อมูลและใช้เทคนิคการเขียนโปรแกรมเชิงวัตถุ เป็นระบบจัดการเนื้อหา เช่น การแคชหน้า การสนับสนุนหลายภาษา ปลั๊กอินและ ส่วนขยาย

Architecture ของ Joomla :



Joomla Architecture

Joomla เขียนด้วย PHP และ อิงตามรูปแบบการออกแบบ MVC (Model-View-Controller) สถาปัตยกรรมของ Joomla ประกอบด้วยเลเยอร์ต่อไปนี้:

- Database:
ประกอบด้วยข้อมูล ยกเว้นไฟล์รูปภาพและเอกสารที่สามารถจัดเก็บ จัดการ และจัดระเบียบในลักษณะเฉพาะ ประกอบด้วยข้อมูลผู้ใช้ เนื้อหา และข้อมูลที่จำเป็นอื่นๆ ของไซต์
- Joomla Framework:
คือชุดของซอฟต์แวร์โอเพ่นซอร์สที่สร้าง Joomla CMS ได้รับการพัฒนาให้มีความยืดหยุ่นมากขึ้นและแบ่งกรอบงานออกเป็นแพ็คเกจโมดูลาร์เดียว ซึ่งช่วยให้แต่ละแพ็คเกจพัฒนาได้ง่ายขึ้น
- Components:
มี mini-applications ซึ่งประกอบด้วยสองส่วน:
 - Administrator
 - Site
 เมื่อใดก็ตามที่โหลดหน้า ส่วนประกอบจะถูกเรียกเพื่อแสดงเนื้อหาของหน้าหลัก
- Modules:
โมดูลเป็นส่วนเสริมที่ใช้ในการแสดงหน้าใน Joomla นอกจากนี้ยังใช้เพื่อแสดงข้อมูลใหม่จากส่วนประกอบ
- Plugin:
นี่เป็นส่วนขยายของ Joomla ซึ่งมีความยืดหยุ่นและมีประสิทธิภาพมากสำหรับการขยายกรอบงาน ประกอบด้วยรหัสบางส่วนที่ใช้ในการเรียกใช้งานทริกเกอร์เหตุการณ์เฉพาะ
- Templates:
กำหนดรูปลักษณ์ของเว็บไซต์ Joomla เทมเพลตที่ใช้มี 2 ประเภทคือ
 - Front-end : เป็นวิธีการจัดการรูปลักษณ์ของเว็บไซต์
 - Back-end : ใช้ในการจัดการหรือควบคุมการทำงานโดย Administrator
- Web Server:
เป็นเซิร์ฟเวอร์ที่ผู้ใช้โต้ตอบ มันส่งหน้าเว็บให้กับลูกค้า HTTP (Hyper Text Transfer Protocol) ใช้ในการสื่อสารระหว่างไคลเอนต์และเซิร์ฟเวอร์

Quality attribute scenarios:

- Compatibility : รองรับ design ใน หลาย device
 - ❖ Source of stimulus: user, developer
 - ❖ stimulus : Responsive design
 - ❖ Environment: ช่วงการออกแบบ, พัฒนาระบบ
 - ❖ Artifact: ระบบ
 - ❖ Respond : มีอุปกรณ์มือถือและแพลตฟอร์มเบราว์เซอร์ที่หลากหลาย
 - ❖ Respond measure: ความพยายามใช้ระบบ, ลดเวลาการพัฒนา, ลดค่าใช้จ่าย
- Modifiability : ปรับแต่งไซด์เพื่อรวมฟังก์ชันที่คุณต้องการ
 - ❖ Source of stimulus: user, developer
 - ❖ stimulus : ต้องการเพิ่ม / ลบ / แก้ไข / เปลี่ยนฟังก์ชัน
 - ❖ Environment: ช่วงการออกแบบ, compile time, พัฒนาระบบ
 - ❖ Artifact: สภาพแวดล้อมของระบบ
 - ❖ Respond : เปลี่ยนโดยไม่มีผลกระทบกับฟังก์ชันอื่นๆ
 - ❖ Respond measure: ลดเวลาการพัฒนา
- Usability : ใช้เวลาในการเขียน Code น้อยลง
 - ❖ Source of stimulus: user, developer
 - ❖ stimulus : เขียนโค้ดง่ายและมีประสิทธิภาพ
 - ❖ Environment: runtime , configuration tiome
 - ❖ Artifact: สภาพแวดล้อมของระบบ
 - ❖ Respond :เขียน CSS และ jQuery น้อย
 - ❖ Respond measure: ลดเวลาการพัฒนา, ลดค่าใช้จ่าย