

Metapolib

เป็นไลบรารีที่ครอบคลุมสำหรับการสร้างการแสดงผลแบบสแตติก ภาพเคลื่อนไหว และแบบโต้ตอบใน Python Matplotlib โดยที่ จะทำให้สามารถทำจากเรื่องง่ายเป็นยากได้ และจากหลักการออกแบบด้วยหลักปรัชญาที่ว่าควรจะสามารถสร้างพล็อตต่างๆ ได้ด้วยคำสั่งเพียงไม่กี่คำ หรือเพียงคำสั่งเดียว โดยที่หลักการทำงานหลักๆ จะมาจาก Library การลงในแต่ละจุดพื้นฐานเป็นส่วนๆ และเป็นระบบปฏิบัติการที่เป็นอิสระและอินเทอร์เฟซผู้ใช้แบบกราฟิก (GUI) สามารถใช้โดยไม่มี (GUI) เป็นส่วนหนึ่งของเว็บเซิร์ฟเวอร์เพื่อสร้างพล็อตและรูปภาพเอาต์พุต ที่หลากหลาย หรือสามารถฝังลงในแอปพลิเคชันขนาดใหญ่ได้โดยใช้ GUI ตัวใดตัวหนึ่ง (เช่น GTK, Tk หรือ WXwindows) ที่ทำงานบนหนึ่งในหลาย ๆ OS และจากที่กล่าวมาทำให้รู้ว่าเป้าหมายของ Metapolib มีอะไรบ้างซึ่งจะทำการระบุไว้ดังนี้

1. Plot ควรมีคุณภาพที่ดีโดยเฉพาะข้อขบถต่างๆ
2. ฝังได้ในส่วนต่อประสานผู้ใช้แบบกราฟิกสำหรับการพัฒนาแอปพลิเคชันและโค้ดควรอ่านเข้าใจได้
3. การสร้าง Plot ควรทำได้ไม่ยาก
4. เนื่องจากซอฟต์แวร์เป็นโอเพ่นซอร์ส จึงสามารถดาวน์โหลดใช้และส่งต่อได้อย่างอิสระ

Software Architecture

สถาปัตยกรรมโค้ด matplotlib แบ่งออกเป็น 3 ส่วนตามแนวคิดที่จะระบุไว้ข้างล่างดังนี้

1. อินเทอร์เฟซของ MATLAB คือชุดของฟังก์ชันที่อนุญาตให้ผู้ใช้สร้างพล็อตจากบรรทัดคำสั่ง
2. Frontend (matplotlib API) คือชุดของคลาสที่ทำงานเป็นหลักโดยจะทำการสร้างและจัดการ figures, text, lines, plots ฯลฯ นี่คือนินเทอร์เฟซแบบนามธรรมที่ไม่ทราบอะไรเกี่ยวกับเอาต์พุตหมายความว่า จะทำงานโดยที่ไม่รู้ผลลัพธ์นั่นเอง
3. Backend เป็นอุปกรณ์ที่ขึ้นอยู่กับอุปกรณ์หรือตัวแสดงผลที่เปลี่ยนการแสดงผลส่วนหน้าที่เป็นเอกสารหรืออุปกรณ์แสดงผล

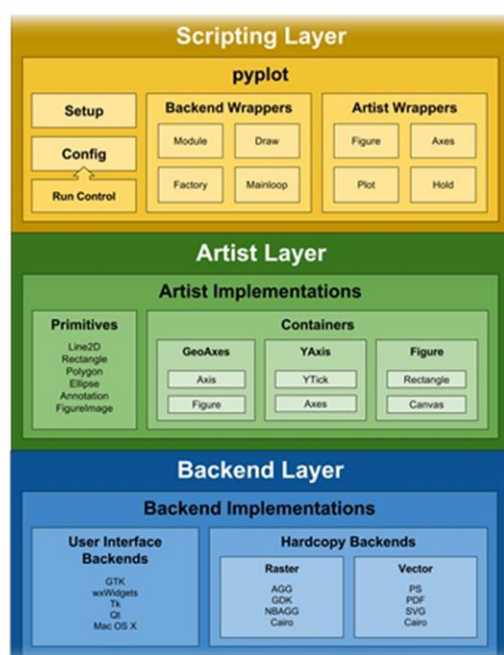
Agg คือไลบรารี Anti-Grain Graphics ที่ช่วยให้สามารถเขียนกราฟิกเวกเตอร์ไปยังผู้ซื้อ ซึ่งสามารถบล็อกการถ่ายโอน (หรือ BLT) ไปยัง display device ซึ่งหมายความว่า การปรับใช้เชิงโต้ตอบทั้งหมดตาม Agg หลักเลี้ยง matplotlib – A Portable Python Plotting Package 93 ตามรูปด้านล่างจะเห็น พล็อตทางการเงินที่ใช้ค่าสูง ต่ำ และข้อจำกัด astockprice.graphical ของ GUI จากนั้นทำการแสดงผลกราฟิกที่เหมือนกันโดยไม่คำนึงถึงอินเทอร์เฟซ GUI



Figure 2. A financial plot that uses the daily high, low, and closing values of a stock price.

Design Architecture

-เป็นรูปแบบ Model-View-Controller(MVC)



-จากภาพจะเห็นว่ามีมีการมีการเรียกใช้ Plot ในที่นี้ หมายถึง Library matplotlib โดยที่ผู้ใช้จะทำการ interface เปรียบเทียบซึ่งมีหลักการทำงานคล้ายกับ Controller แล้วทำการส่งคำสั่งไปยัง Frontend (คล้าย model) แล้วทำการ update คำสั่งไปที่ Backend (คล้าย View) แล้วทำการแสดงผลใน render ให้ผู้ใช้ได้เห็น

Quality Attribute Scenarios

1. Usability

-หมายความว่า matplotlib มีการเรียนรู้ที่รวดเร็ว มีประสิทธิภาพในการทำงาน ปรับเปลี่ยนความต้องการตามที่ใช้ต้องการได้

2. Modifiability

-matplotlib สามารถทำการปรับเปลี่ยนได้เพราะว่าเป็นโอเพ่นซอร์สนั่นเอง

3. Integrability

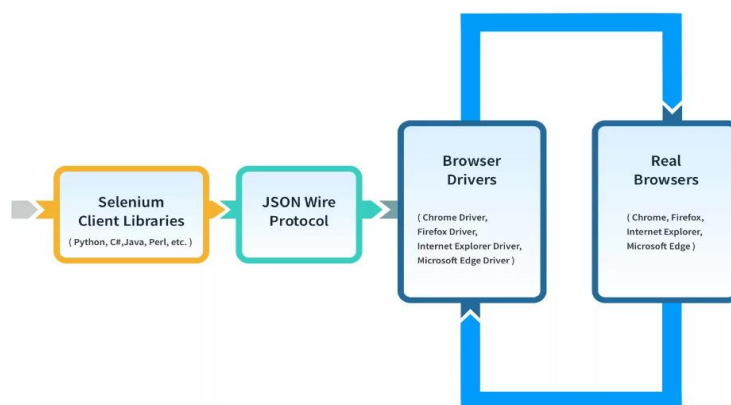
- matplotlib มีการทำงานร่วมกับไฟล์ \LaTeX

Reference: https://www.researchgate.net/publication/234238535_matplotlib_--_A_Portable_Python_Plotting_Package

Selenium Webdriver

- Selenium WebDriver เป็นเฟรมเวิร์กของเว็บที่อนุญาตให้คุณดำเนินการทดสอบข้ามเบราว์เซอร์ เครื่องมือนี้ใช้สำหรับการทดสอบแอปพลิเคชันบนเว็บโดยอัตโนมัติเพื่อตรวจสอบว่าทำงานได้ตามที่คาดหวัง Selenium WebDriver ให้คุณเลือกภาษาการเขียนโปรแกรมเพื่อสร้างสคริปต์ทดสอบ ดังที่กล่าวไว้ก่อนหน้านี้ เป็นความก้าวหน้าเหนือ Selenium RC เพื่อเอาชนะข้อจำกัดบางประการ Selenium WebDriver ไม่สามารถจัดการส่วนประกอบของหน้าต่างได้ แต่ข้อเสียเปรียบนี้สามารถเอาชนะได้โดยใช้เครื่องมือเช่น Sikuli, Auto IT เป็นต้น

Design Architecture



จะเป็นรูปแบบ Plug-in (Microkernel) ที่เป็น design นี้เพราะว่ามี Browser Drivers , Real Browsers เป็นการทำงานหลักแล้วมี Selenium Client Libraries และ JSON Wire Protocol มาเสริมการทำงานให้ นั่นเองซึ่งที่เพิ่มเข้ามาอาจจะเป็นตัวแปรหรือฟังก์ชันเสริมก็ได้

Quality Attribute Scenarios

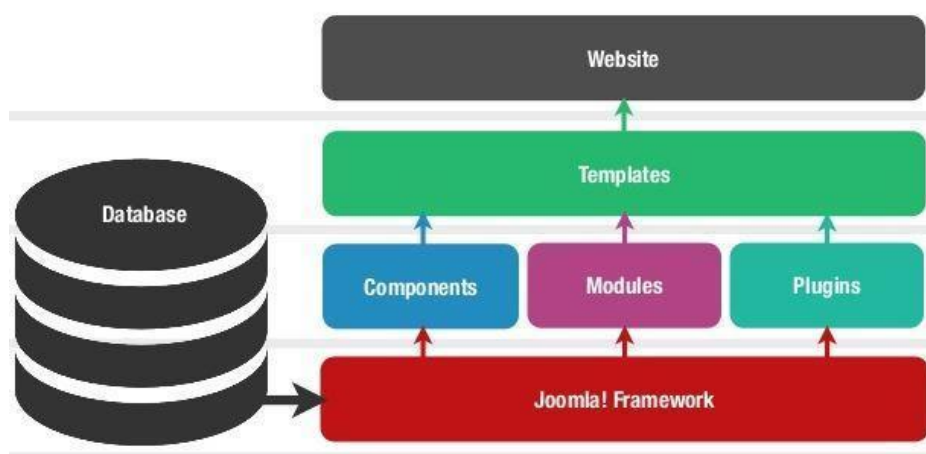
1. Performance- Selenium Webdriver มีการทำงานที่ประหยัดเวลาได้ค่อนข้างมาก
2. Testability- สามารถตรวจจับข้อผิดพลาดได้ หรือจะวัดประสิทธิภาพก็ยังสามารถทำได้
3. Usability- เป็นหนึ่งในเครื่องมือโอเพ่นซอร์สที่ที่ง่ายต่อการเริ่มต้นสำหรับการทดสอบแอปพลิเคชันบนเว็บ นอกจากนี้ยังช่วยให้สามารถทำการทดสอบความเข้ากันได้ข้ามเบราว์เซอร์

Reference: <https://www.browserstack.com/guide/selenium-webdriver-tutorial#:~:text=Selenium%20WebDriver%20is%20a%20web,language%20to%20create%20test%20scrupts.>

Joomla

Joomla เป็นระบบจัดการเนื้อหาโอเพ่นซอร์ส (CMS) ช่วยให้คุณสร้างเว็บไซต์และแอปพลิเคชันแบบไดนามิกที่ทรงพลัง มีอินเทอร์เฟซที่ใช้งานง่ายซึ่งช่วยให้คุณใช้คุณลักษณะและฟังก์ชันการทำงานได้อย่างเต็มที่ Joomla เขียนด้วย PHP และใช้ฐานข้อมูล MySQL เพื่อเก็บข้อมูลในขณะที่ใช้เทคนิคการเขียนโปรแกรมเชิงวัตถุ สามารถตั้งค่าได้ด้วยการติดตั้งเพียงคลิกเดียวผ่านแผงควบคุมเว็บไซต์ Joomla มีบทความมากมายที่ช่วยคุณในการติดตั้ง Joomla มีบริการโฮสติ้ง Joomla ที่หลากหลาย

Design Architecture



- Joomla! ใช้สถาปัตยกรรมการออกแบบ Model-View-Controller (MVC) ตามรูปแบบ MVC เมื่อ Joomla! ประมวลผลคำขอ อันดับแรกจะวิเคราะห์ URL เพื่อประเมินว่าองค์ประกอบใดจะประมวลผลคำขอโมเดลประกอบด้วยข้อมูลที่ส่วนประกอบใช้ นอกจากนี้ยังเป็นความรับผิดชอบของ Model ในการอัปเดตฐานข้อมูลเมื่อจำเป็นและจำเป็น มุมมองรับผิดชอบในการผลิตผลงาน สามารถติดต่อกับฐานเพื่อรับข้อมูลที่ต้องการได้ หลังจากที่มีมุมมองสร้างผลลัพธ์ออกมาแล้ว ส่วนประกอบจะคืนการควบคุมให้กับกรอบงาน Joomla! ซึ่งจะดำเนินการกับแม่แบบ

Quality Attribute Scenarios

1. Usability

- Joomla เป็นโอเพ่นซอร์สและใช้งานได้ฟรีทั้งหมด แต่ฟรีไม่ได้หมายความว่ามันขาดคุณสมบัติ อันที่จริง อินเทอร์เฟซที่ใช้งานง่ายจะทำให้ประหลาดใจด้วยคุณสมบัติ WYSIWYG ที่ให้ผลลัพธ์เหมือนกันทุกประการ บั๊กจี้ที่ทำให้รู้สึกดีอีกประการหนึ่งก็คือการอัปเดตบ่อยๆ Joomla นำเสนอการอัปเดตใหม่ในรูปแบบของคุณสมบัติและฟังก์ชันใหม่ ทุกครั้งที่อัปเดตใหม่ การทำงานจะง่ายขึ้น

2. Security

-ความปลอดภัยเป็นปัจจัยสำคัญที่ควรพิจารณาเมื่อคุณสร้างเว็บไซต์ Joomla ให้การรับรองความถูกต้องสองปัจจัยเพื่อหลีกเลี่ยงโอกาสในการถูกแฮ็คเพื่อให้ไซต์ของคุณจะไม่ถูกแฮ็ค แต่ในกรณีที่คุณทิ้งชื่อผู้ใช้และรหัสผ่านทั่วไปไว้ และมีคนเข้ามาจะสามารถกู้คืนไซต์ Joomla ที่ถูกแฮ็คได้อย่างง่ายดาย

3. Availability

- Joomla เป็นหลายภาษา รองรับ 75 ภาษาสำหรับผู้ไม่เข้าใจภาษาอังกฤษก็สามารถใช้ได้นั่นเอง

Reference: <https://blog.templatetoaster.com/what-is-joomla/>