

Matplotlib

สามารถใช้คำสั่งไม่กี่คำสั่งในการสร้างพล็อตหรือกราฟ โดยข้อดีของ matplotlib คือไลบรารีการลงจุดพื้นฐาน ซึ่งเป็นระบบปฏิบัติการที่ไม่ขึ้นกับระบบปฏิบัติการและส่วนติดต่อผู้ใช้แบบกราฟิก (GUI) สามารถใช้โดยไม่มี GUI เป็นส่วนหนึ่งของเว็บเซิร์ฟเวอร์เพื่อสร้างพล็อตและรูปภาพในเอาต์พุตสำเนาที่หลากหลาย หรือฝังลงในแอปพลิเคชันขนาดใหญ่ได้โดยใช้ GUI ตัวใดตัวหนึ่งที่ทำงานบนหนึ่งในระบบปฏิบัติการหลายตัว (เช่น Windows, OS X, Solaris และ Linux) และยังเป็น Open-Source สามารถโหลด ใช้งาน แจกจ่ายได้ฟรี

Software Architecture

ประกอบด้วย 3 ส่วนคือ

- 1) Interface สำหรับผู้ใช้งานใช้พล็อตผ่าน Command Line
- 2) Front-End หรือ matplotlib API ที่เป็นกลุ่มของ class ที่ใช้สร้างและจัดการ figure, ตัวหนังสือ, เส้น, กราฟ
- 3) Back-End ขึ้นอยู่กับแต่ละเครื่องที่ใช้ โดยใช้เรนเดอร์ผลลัพธ์ที่ได้จาก Front-End มาเป็นไฟล์ชนิดต่างๆ หรือแสดงผล โค้ดการเรนเดอร์ที่สำคัญส่วนใหญ่ เขียนด้วยภาษา C/C++ จึงให้ประสิทธิภาพที่ดีมาก

Design Architecture

ที่ใช้ Layer โดยแบ่งเป็น 3 ลำดับ Interface Front-End และ Back-End

-Interface จะทำการติดต่อกับผู้ใช้ผ่าน Command Line Interface แล้วส่งข้อมูลไปหา

Front-End

- Front-End จะทำการจัดการกับ figure, ตัวหนังสือ, กราฟ แล้วส่งให้ Back-End
- Back-End จะทำการเรนเดอร์ข้อมูลที่ได้รับมาจาก Front-End ออกมาให้เป็นรูปภาพ

Quality Attribute Scenarios

1. Usability โดย matplotlib นั้นสามารถเรียนรู้ได้รวดเร็ว, ใช้ได้อย่างมีประสิทธิภาพ, สามารถปรับเปลี่ยนตามความต้องการของผู้ใช้ได้, สามารถทำงานได้อย่างง่ายเพราะ code ที่ใช้งานง่าย
2. Modifiability โดย matplotlib นั้นสามารถแก้ไขได้เนื่องจากเป็น software ที่เป็นในรูปแบบ open-source
3. Integrability โดย matplotlib นั้นใช้งานร่วมกับไฟล์ชนิด TEX, การเซฟรูป figure เป็นไฟล์รูปชนิดมาตรฐาน

ที่มา : [matplotlib -- A Portable Python Plotting Package \(researchgate.net\)](https://www.researchgate.net/publication/312111111/matplotlib--A-Portable-Python-Plotting-Package)

Selenium WebDriver

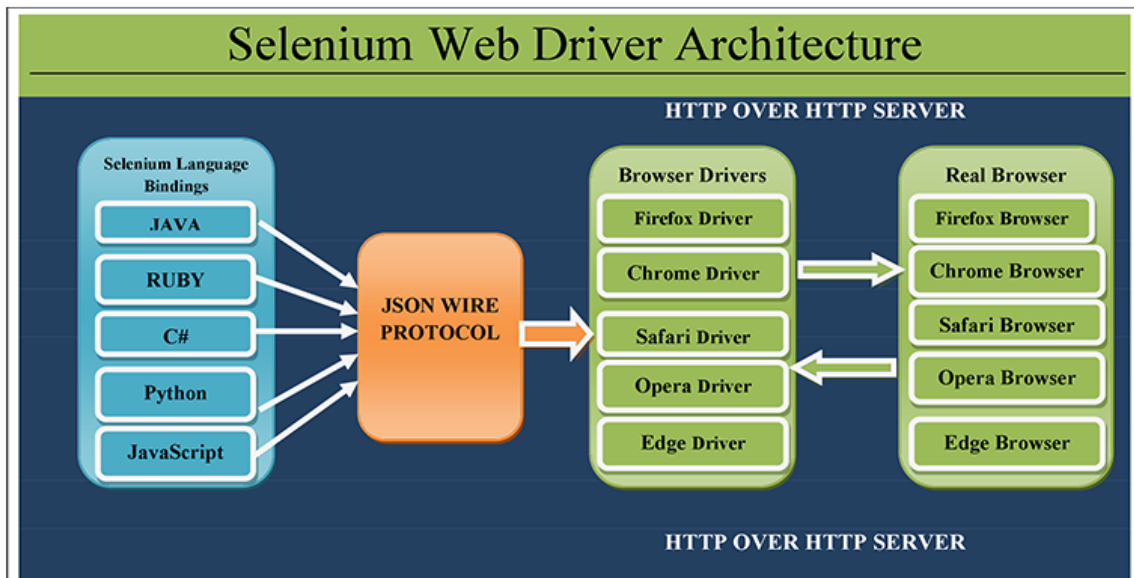
Selenium webdriver เรียกอีกอย่างว่า selenium 2.0 Selenium webdriver สื่อสารโดยตรงกับเบราว์เซอร์

ดังนั้น selenium webdriver จึงเร็วกว่า selenium RC Selenium webdriver รองรับเว็บเบราว์เซอร์หลายตัวและยังรองรับแอปพลิเคชัน Ajax

เป้าหมายหลักของซีลีเนียม webdriver คือการปรับปรุงการสนับสนุนสำหรับปัญหาการทดสอบเว็บแอปพลิเคชันที่ทันสมัย Selenium webdriver รองรับหลายภาษาในการเขียนสคริปต์ทดสอบ API ของ Selenium webdriver นั้นง่ายกว่า selenium RC's .

อย่างไรก็ตาม แม้จะมีข้อดีทั้งหมดของไดรเวอร์เว็บซีลีเนียม แต่ก็ยังมีข้อจำกัดบางประการในการทดสอบเว็บแอปพลิเคชัน Selenium webdriver ไม่มีฟังก์ชันการทำงานในการสร้างภาพหน้าจอสำหรับกรณีทดสอบความล้มเหลว

Selenium webdriver ไม่มีความสามารถในการสร้างผลการทดสอบ ขึ้นอยู่กับเครื่องมือของบุคคลที่สามในการสร้างรายงานการทดสอบ Design Architecture



จะเป็นรูปแบบ design ของ Plug-in (Microkernel) โดยจะเห็นได้ว่ามีสอง ตัวประกอบหลักๆ โดยจะมีชุดการทำงานหลักและตัวแปรเฉพาะที่เพิ่มฟังก์ชันการทำงานให้กับแกน

องค์ประกอบหลักของ Selenium WebDriver มีอยู่ 4 ตัวได้แก่

- 1) Selenium Client Library or Language Bindings
- 2) Browser Driver
- 3) Browsers.
- 4) JSON WIRE PROTOCOL Over HTTP Client

Quality Attribute Scenarios

1. Testability สามารถเสนอ/สาธิตได้ว่ามีข้อผิดพลาด / ตรวจจับข้อผิดพลาดได้ในการวัด จะวัดประสิทธิภาพ คือ หาข้อผิดพลาดได้ทุก ครั้ง, วัดระยะเวลานานแค่ไหนในการทดสอบ, การทดสอบจะครอบคลุมหรือไม่

2. Modifiability โดย Selenium webdriver รองรับหลายภาษาในการเขียนสคริปต์ทดสอบ API ของ Selenium webdriver และมีการ ปรับปรุงเรื่อยๆ

3. Availability เกี่ยวกับความสนใจที่ระบบไม่พร้อมใช้งาน หรือ ระบบล้ม (Failure) และผลที่เกิดจากระบบล้ม และอาจมี System failure เกิดจากการที่ระบบไม่สามารถให้บริการได้ในระยะเวลาหนึ่ง

ที่มา : Analysis and Design of Selenium WebDriver

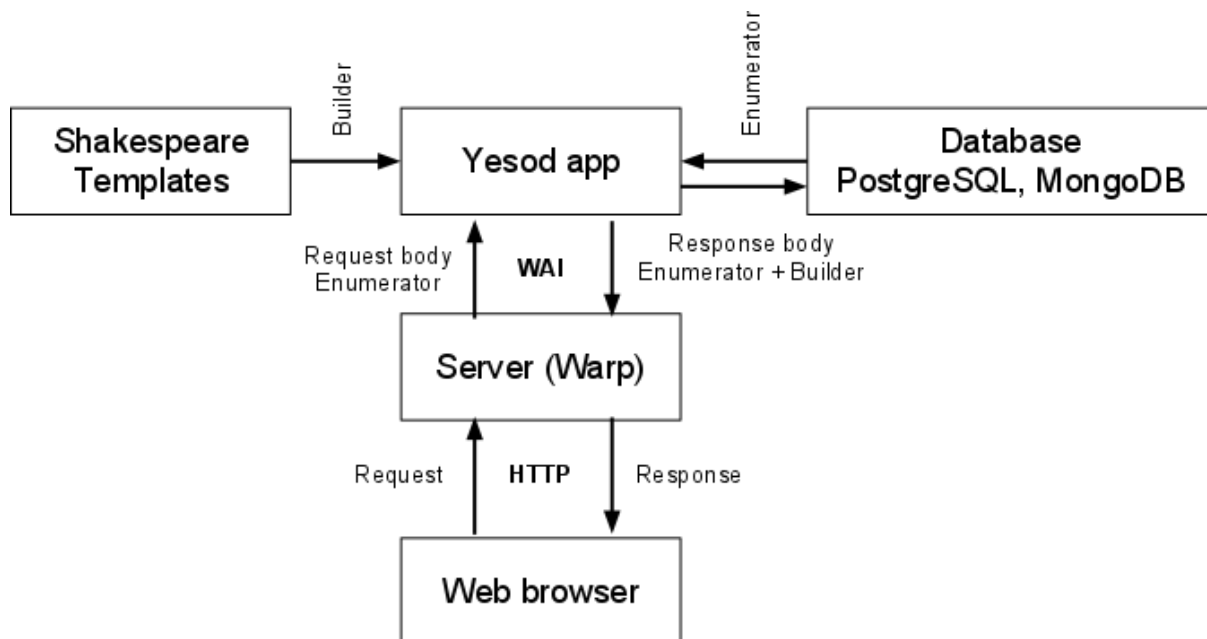
Yesod

Yesod เป็น web framework ที่เขียนขึ้นในภาษา Haskell ซึ่งเป็นภาษาที่คงที่ (Static) จึงทำให้ Yesod ทำงานได้รวดเร็วกว่า และปลอดภัยกว่า

เป้าหมายของ Yesod คือการใช้ข้อได้เปรียบจากภาษา Haskell ในการพัฒนา web Yesod ตั้งใจที่จะทำให้โค้ดมีความรวบรัดมากที่สุดเท่าที่ทำได้ ทุกบรรทัดจะถูกเช็คความถูกต้องตอนที่ compile แทนที่จะใช้ library ภายนอกเพื่อตรวจสอบ Yesod จะใช้ Compiler ตรวจสอบแทน โครงสร้างของ Yesod

Yesod ใช้โครงสร้างสถาปัตยกรรมตามรูปแบบ Model-View-Controller (MVC) ทำให้มีแม่แบบของระบบของส่วน View ที่แยก ออกจากส่วนตรรกะ ให้ระบบ Object-Relational Mapping (ORM) และมี Controller ที่ใช้ในการ routing

การเชื่อมต่อกับ server จะใช้ Web Application Interface (WAI) ซึ่งมีจุดมุ่งหมายในความเป็นส่วนใหญ่ โดย WAI จะทำการ สับสนุน Back-End ในทุกๆด้าน และ ประสิทธิภาพที่เกี่ยวข้องกับการใช้งานพีเจียร์ของภาษา Haskell



Design Architecture ที่ใช้

Model View Controller (MVC) โดยให้ Model = Database, View = Shakespeare Templates, Yesod app = Controller

Quality Attribute Scenarios

Security จากการใช้ภาษา Haskell ซึ่งเป็นภาษาที่มีช่องโหว่น้อยกว่า

- Source of Stimulus : ผู้ไม่ประสงค์ดีที่ต้องการโจมตี
- Stimulus : การแสดงผลข้อมูลที่ไม่พึงประสงค์จากการโจมตี
- Artifacts : ข้อมูลภายในระบบ - Environment : ระบบ
- Responds : ข้อมูลถูกป้องกันไม่ให้แสดงผลข้อมูลที่ไม่พึงประสงค์
- Respond Measures : จำนวนการโจมตีที่ป้องกันได้, ข้อมูลใดที่ถูกป้องกัน

จากการโจมตี

Performance จากการใช้ภาษา Haskell ซึ่งเป็นภาษาชนิดคงที่ทำให้มีประสิทธิภาพที่สูงกว่า

- Source of Stimulus : ส่วนประกอบต้องการใช้งานอีกส่วน
- Stimulus : เหตุการณ์ที่จำเป็นต้องใช้งานส่วนประกอบอีกส่วน - Artifacts : บางส่วนของระบบที่เกี่ยวข้อง
- Environment : การทำงานแบบปกติของแอปพลิเคชัน
- Responds : การใช้งานทรัพยากรของระบบ
- Respond Measures : ใช้เวลาดลดลง, ใช้ทรัพยากรลดลง

Testability จากการใช้Compiler ในการตรวจเช็คความถูกต้อง

- Source of Stimulus : System Tester
- Stimulus : ทดสอบฟังก์ชัน, ทดสอบคุณภาพ, ทดสอบความอันตราย
- Artifacts : ระบบ
- Environment : ทั้งระบบ
- Responds : ผลการตรวจสอบจาก compiler
- Respond Measures : ความอดสาหะในการทดสอบระบบ

ที่มา : [The Architecture of Open Source Applications \(Volume 2\): Yesod \(aosabook.org\)](http://The Architecture of Open Source Applications (Volume 2): Yesod (aosabook.org))