**Matplotlib**

matplotlib เป็นการพล็อตและการสร้างภาพ 2 มิติแบบพกพาที่มุ่งสร้างภาพข้อมูลทางวิทยาศาสตร์ วิศวกรรม และﬁnancial data เป็นหลัก อย่างไรก็ตาม พลังที่แท้จริงของ matplotlib คือไลบรารีการลงจุดพื้นฐาน ซึ่งเป็นระบบปฏิบัติการที่ไม่ขึ้นกับระบบปฏิบัติการและส่วนติดต่อผู้ใช้แบบกราฟิก (GUI) สามารถใช้โดยไม่มี GUI เป็นส่วนหนึ่งของเว็บเซิร์ฟเวอร์เพื่อสร้างพล็อตและรูปภาพในเอาต์พุตสำเนาที่หลากหลาย หรือฝังลงในแอปพลิเคชันขนาดใหญ่ได้โดยใช้ GUI ตัวใดตัวหนึ่ง (เช่น GTK, Tk หรือ WXwindows) ที่ทำงานบนหนึ่งในระบบปฏิบัติการหลายตัว (เช่น Windows, OS X, Solaris และLinux)

**Architecture**

matplotlib code แบ่งออกเป็นสามส่วนตามแนวคิดได้ดังนี้

* อินเทอร์เฟซของ MATLAB คือชุดของฟังก์ชันที่อนุญาตให้ผู้ใช้สร้างพล็อตจากบรรทัดคำสั่ง
* frontend หรือ matplotlib API คือชุดของคลาสที่ทำงานหนักยกโดยการสร้างและจัดการตัวเลข

ข้อความ เส้น โครงเรื่อง ฯลฯ นี่คือส่วนต่อประสานนามธรรมที่ไม่รู้อะไรเลยเกี่ยวกับผลลัพธ์

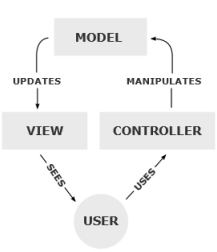
* Backend เป็นอุปกรณ์วาดภาพหรือเรนเดอร์ที่ขึ้นอยู่กับอุปกรณ์ที่เปลี่ยนการแสดงส่วนหน้าเป็นเอกสาร

(JPEG, PNG, PDF, PS, SVG, Paint, GD) หรืออุปกรณ์แสดงผล (Agg, GTK/GTKAgg, TkAgg,WX/WXAgg). โค้ดการเรนเดอร์ที่สำคัญส่วนใหญ่เขียนด้วยภาษา C/C++จึงให้ประสิทธิภาพที่

ดีมาก

**Design Architecture**

จะเป็นรูปแบบ design ของ Model-View-Controller (MVC)โดยจาก software architecture นั้น จะเห็นได้ว่าเมื่อมีการเรียกใช้ไลบลารี matplot โดยตัว user นั้นจะใช้ interface ในการสั่งการจึงเปรียบ interface ได้กับ controller จากนั้นจะส่งคำสั่งไปยัง front-end ซึ่งก็คือ model จากนั้นจึง จึงจะ update ไปยัง view หรือก็คือ back-end ซึ่งจะทำหน้าที่ในการแสดงผล/renderผลลัพธ์ส่งไปยัง user



**Quality Attributes Scenario**

**1.Usability**

โดย matplotlib นั้นสามารถเรียนรู้ได้รวดเร็ว, ใช้ได้อย่างมีประสิทธิภาพ, สามารถปรับเปลี่ยนตามความ

ต้องการของผู้ใช้ได้, สามารถทำงานได้อย่างง่ายเพราะ code ที่ใช้งานง่าย

**2.Modifiability**

โดย matplotlib นั้นสามารถแก้ไขดัดแปลงได้เนื่องจากเป็นsoftwareที่เป็นในรูปแบบ open-source

**3. Integrability**

โดย matplotlib นั้นใช้งานร่วมกับไฟล์ชนิด TEX, การเซฟรูป figure เป็นไฟล์รูปชนิดมาตรฐาน

อ้างอิง (PDF) matplotlib -- A Portable Python Plotting Package (researchgate.net)

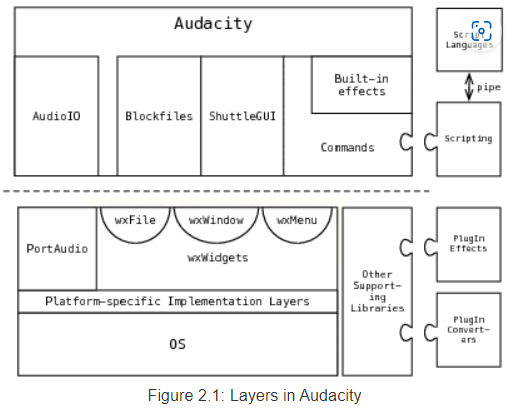
**Audacity**

Audacity (ออแดซิตี) เป็นซอฟต์แวร์เสรี ใช้สำหรับตัดต่อเสียง สนับสนุนแพลตฟอร์ม แม็ค วินโดวส์ และลินุกซ์ สามารถอัดเสียง อิมพอร์ต/เอกซ์พอร์ต แปลงไฟล์ไป-มา ได้หลายฟอร์แม็ต แก้ไข ตัตแต่ง วิเคราะห์ สนับสนุนไฟล์ในหลายรูปแบบรวมถึง WAV MP3 Ogg Vorbis และไฟล์รูปแบบอื่นๆ

<https://th.wikipedia.org/wiki/%E0%B8%AD%E0%B8%AD%E0%B9%81%E0%B8%94%E0%B8%8B%E0%B8%B4%E0%B8%95%E0%B8%B5>

**Architecture**

Audacity ทำงานเป็นชั้นๆอยู่บนlibraries หลายๆอัน เมื่อทำการส่องดูcode ของโปรแกรม audicity จะเห็นได้ว่าcode บางส่วนเท่านั้นที่เป็นส่วนสำคัญของโปรแกรม ส่วนใหญ่ดารทำงานจะขั้นอยู่กับส่วนของ libraries นอกจาก audicity จะมีส่วนของ function built-in ของ audio effectแล้วยังรองรับ LADSPA (Linux Audio Developer'sSimple Plugin API) ซึ่ง เป็น plugin ส่วนเสริมสำหรับการโหลด audio effect อื่นๆ มาใช้งานร่วมในโปรแกรม เช่นVAMP API ใน audacity ทำ หน้าที่ในการวิเคราะห์ และแยกแยะเสียงเพื่อให้ง่ายต่อการ editในการเพิ่ม คำสั่ง การทำงานให้โปรแกรม code ที่มี script ในการทำงานร่วมกับส่วนของการทำงานหลักไม่จำ เป็นที่จะต้องมีcode อยู่ในโปรแกรมโดยตรง



แสดงบางเลเยอร์และโมดูลใน Audacity ไดอะแกรมเน้นสามคลาสที่สำคัญภายใน wxWidgets <https://www.aosabook.org/en/audacity.html>

**Quality Attributes :**

**1.Usability**

จากจุดมุ่งหมายของตัว open-source ที่ต้องการให้application ใช้งานง่าย

**2. Security**

Audacity เป็นซอฟต์แวร์ open-source ซึ่งหมายความว่าซอร์สโค้ดของโปรแกรมนั้นเปิดเผยต่อสาธารณะและสามารถตรวจสอบได้เพื่อยืนยันข้อมูลที่ส่งไปอย่างจำกัด คุณยังสามารถตรวจสอบไฟล์ปฏิบัติการของได้

มีฟีเจอร์ในการรักษาความปลอดภัยดังนี้

**1)update checking** ผู้ใช้จะได้รับแจ้งเกี่ยวกับการตรวจสอบการอัปเดตเมื่อมีการติดตั้ง Audacity เป็นครั้งแรกและมีลิงก์ที่ชัดเจนเพื่อปิดใช้งานในการตั้งค่า

**2)error reporting.**

**3. Performance**

จากการใช้ BlockFile ให้ ไม่จำเป็นต้องแก้ไขไฟล์ทั้งไฟล์

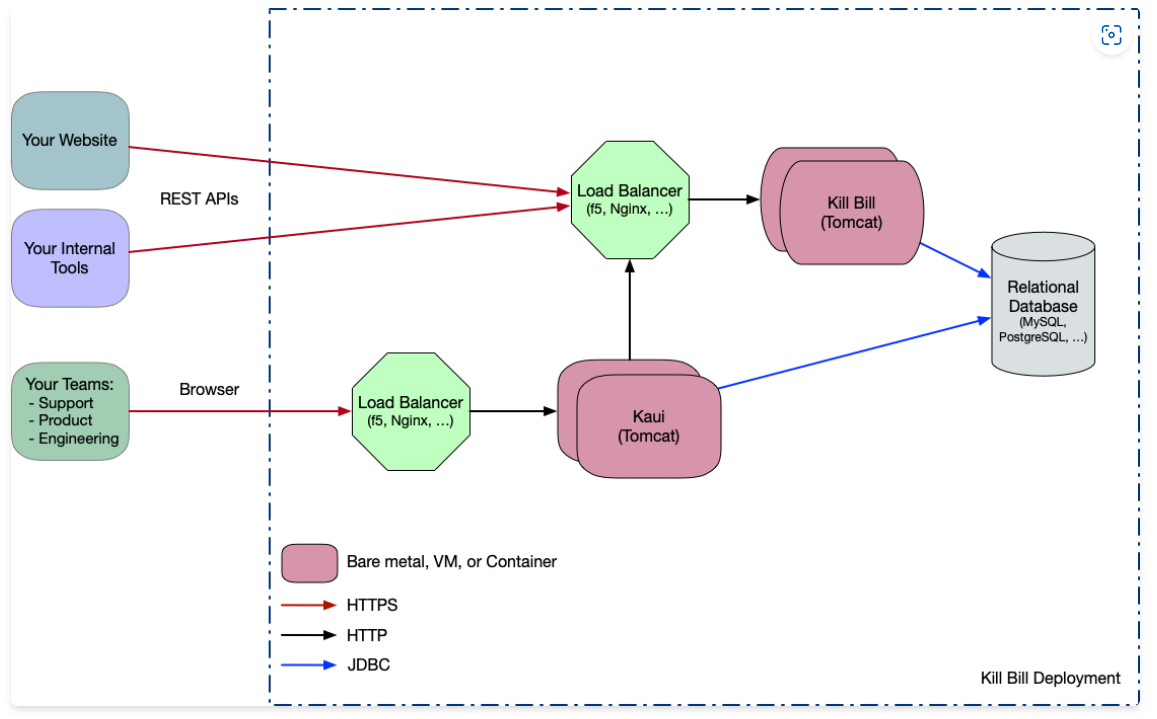
<https://www.audacityteam.org/about/desktop-privacy-notice/>

**Kill Bill**

Kill Bill เป็น open-source แพลตฟอร์มเรียกเก็บเงินและการชำระเงินแบบสมาชิก โดยสามารถที่จะ

สร้าง logic ของธุรกิจตัวเองเพิ่มได้ขึ้นได้ และสามารถที่จะ custom การชำระเงินและเรียกเก็บเงินเพิ่มเติมเองได้

**architectural patterns/stylesKill Bill deployment architecture**

****

ในระดับสูง เรามีซอฟต์แวร์สองชิ้นที่คุณจะต้องปรับใช้และจัดการในศูนย์ข้อมูลของคุณเอง ในคลาวด์ส่วนตัวของคุณ หรือในคลาวด์สาธารณะใดๆ

Kill Bill หรือเซิร์ฟเวอร์ Kill Bill เป็นเซิร์ฟเวอร์ Java แบบสแตนด์อโลนที่ทำงานใน back-end. ลูกค้าของคุณจะไม่มีวันโต้ตอบกับมันโดยตรง แต่เว็บไซต์ของคุณ (อาจเป็นเว็บไซต์ที่กำหนดเอง การปรับใช้ Drupal หรือ WordPress เป็นต้น) และเครื่องมือภายในของคุณ (CRM, CPQ เป็นต้น) จะทริกเกอร์การเรียก REST API (มากกว่า HTTP) เพื่อสร้างการสมัครรับใบแจ้งหนี้ ให้เครดิต ฯลฯ

Kaui อินเทอร์เฟซผู้ใช้ในส่วนหลังของเรา ช่วยให้ทีมของคุณ (การสนับสนุนลูกค้า ผลิตภัณฑ์ การเงิน ฯลฯ) โต้ตอบกับ Kill Bill เพื่อจัดการการคืนเงิน การปรับใบแจ้งหนี้ การรายงาน และอื่นๆ

**Quality Attributes Scenario**

1. Usability

Kill Bill มีฟังก์ชันและความยืดหยุ่นที่ให้คุณสามารถที่จะทดสอบได้ และมีความง่ายในการที่จะกำหนด

ฟังก์ชันของราคาและการเรียกเก็บเงินจำนวนมากที่ง่าย

- ไม่มีสัญญาณการล็อคอิน

- ไม่มีค่าสมัคร

- ไม่มีข้อจำกัดในการกำหนดคุณสมบัติ

- ไม่มีข้อจำกกัดของผู้ให้บริการบุคคลที่สาม

- ไม่มีข้อจำกัดในการสืบค้นข้อมูล

2. Modifiability

คุณสามารถเขียนปลั๊กอินเพื่อปรับเปลี่ยนการทำงานของระบบหรือผสานรวมกับผู้ขายที่เป็นบุคคลที่สาม

ได้ เช่น การแก้ปัญหาการตรวจจับการฉ้อโกง (Accertify, Feedzai, etc.) หรือ ผู้ให้บริการด้านภาษี (Avalara

AvaTax, Vertex, etc.)

3. Testability

มีการทดสอบจำนวนมาก (1,100) :

- Unit tests

- System tests

- Performance tests

- Integration tests

และมีการติดตั้งง่าย (การ run และ/หรือ การเพิ่ม tests) ในอุปกรณ์(เช่น labtop) หรือในระบบคลาวด์

ส่วนตัวหรือสาธารณะของคุณ [**https://docs.killbill.io/latest/what\_is\_kill\_bill.html**](https://docs.killbill.io/latest/what_is_kill_bill.html)