

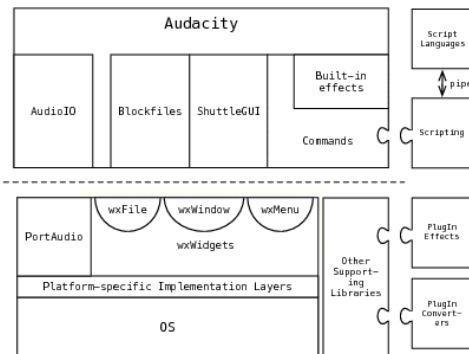
Audacity

Proposal

Audacity เป็นโปรแกรม OpenSource ที่ใช้ตัดต่อและบันทึกเสียงด้วยการใช้ Effect ต่างๆ ซึ่งมีอยู่มากมายและครบครัน เช่น Noise Reduction (ลดเสียง) Amplify (เพิ่มเสียง) ปรับระดับของเดซิเบล ฯลฯ

Architectural Patterns

โดยที่ Architectural Patterns ของ Audacity เป็นสถาปัตยกรรมแบบ Layers โดยใช้หลากหลาย Libraries



จากรูป ภายใน **wxWidget** จะมีส่วนที่สำคัญสามส่วนที่บอกการทำงานของโปรแกรมนี้อยู่ โดยทั้ง **wxFile**, **wxWindow**, **wxMenu** เป็นตัวควบคุมการทำงานของ **Blockfiles ShuttleGUI** และ **Commands** ส่วน **wxWidgets** และ **PortAudio** จะทำการติดต่อกับส่วน Platform-specific Implementation Layers ที่ใช้ติดต่อกับ OS ไว้ใช้สำหรับ Platform ที่แตกต่างกัน ส่วน **Other Supporting Libraries** ไว้สำหรับจัดเก็บ Libraries หรือโมดูลอื่น ๆ ไว้

Quality Attributes

Portability โดยที่ Audacity เป็น Software ที่ใช้กันแพร่หลายและการใช้งานควรจะสามารถทำได้จากทุก Platform, Operating System เพื่อสามารถทดสอบการทำงานบนทุกอุปกรณ์ที่ผู้ใช้งานมีได้

Usability เนื่องจากมีกลุ่มผู้ที่ใช้งานเป็นกลุ่มใหญ่ที่ต้องการใช้ พีเจอร์หลัก ๆ ของโปรแกรมดังนั้นการทำให้

Audacity สามารถใช้งาน และเข้าใจได้ง่ายจะทำให้ผู้ใช้งานเกิดความสะดวก

Performance โดยที่ Audacity มีประสิทธิภาพในการทำ Render ไฟล์เสียงต่าง ๆ ให้ออกมามีคุณภาพและประสิทธิภาพในการทำงานมากที่สุด

Reference :

<http://www.aosabook.org/en/audacity.html>

matplotlib

Proposal

Matplotlib เป็นการรวบรวม Library ของภาษา Python ซึ่งจำเป็นในการสร้างข้อมูลที่มีอยู่สามารถอธิบายออกมาได้ด้วยภาพต่าง ๆ ซึ่งทำให้สามารถบรรยายข้อมูลที่มีอยู่ให้สามารถเข้าใจได้ง่ายขึ้นโดยใช้ภาพเป็นเครื่องบรรยาย ทั้งเป็นแบบ Static, Animated, Visualizations ที่มีปฏิสัมพันธ์ได้อีกด้วย สามารถทำได้ทั้งการ Zoom, Pan, Update ซึ่งก็คือ Graphic ที่ใช้ในการ Publication รวมถึงยังสามารถใช้ในการเปลี่ยนหน้าต่าง รูปแบบของ Visualization ได้ตามที่ต้องการ และการนำข้อมูลไปใช้ต่อที่หลากหลายทั้งนำไป Exports เป็นไฟล์สกุลได้มากมาย และด้วยการที่ matplotlib เป็น Library ของภาษา Python จึงสามารถนำไปฝังกับ User Interface ตามการประยุกต์ใช้ที่ต้องการ รวมถึงประสิทธิภาพในการสร้าง Visualization ที่ใช้ทรัพยากรน้อยกว่าการใช้ User Interface Software อื่น ๆ ที่ใช้ทรัพยากรมากกว่า และยังใช้ร่วมกับ Third-Party Package ที่สามารถทำให้ทำงานร่วมกันได้อย่างมีประสิทธิภาพ อาทิเช่น NumPy ที่ใช้ร่วมกันในการคำนวณและ Plot Graph ต่าง ๆ โดย matplotlib จะช่วยในการ Visualization ข้อมูลที่มาจากการคำนวณให้สามารถอธิบายข้อมูลได้ ประสิทธิภาพจากการคำนวณมากยิ่งขึ้น

Architectural Patterns

โดยที่ Architectural Patterns สถาปัตยกรรมของ matplotlib เป็นสถาปัตยกรรมแบบ Layer โดยที่การทำงานของ Layer ซึ่งชั้นการทำงานของ matplotlib จะแบ่งเป็น 3 Layer ได้แก่ Backend Layer, Artist Layer, Scripting

Layer ซึ่ง Backend Layer จะเป็น Layer ที่มีความซับซ้อนมากที่สุด เป็น Layer ที่ทำหน้าที่คำนวณ และเป็น Algorithm ในการคำนวณทั้งหมดของ matplotlib Library ซึ่งทำหน้าที่ติดต่อสื่อสารกับ Toolkit เหมือนกับ wxPython หรือภาษาที่ใช้วาดภาพเช่น PostScript ภายในเครื่องที่ใช้ในการคำนวณ เป็น Layer ที่รวม Logical Function ของ matplotlib Library ไว้ โดยมี 3 Built-In Abstract Interface หลักไว้คือ

- **FigureCanvas** ทำให้สามารถ Render Canvas ได้
- **Renderer** เป็น Abstract Class ทำหน้าที่เป็น Handler ในการ Operation Render หรือการวาด รับผิดชอบการทำงานภายใน FigureCanvas
- **Event** เป็น Event Handler Input ของผู้ใช้งาน matplotlib Library จาก Keyboard และ Mouse Click เป็นต้น

Artist Layer เป็น Layer ที่อนุญาตให้ผู้ใช้สามารถควบคุมการทำงาน และดึงประสิทธิภาพของ Element ที่อยู่ใน Figure ให้มากที่สุดเท่าที่เป็นไปได้ ช่วยให้ผู้ใช้สามารถปรับเปลี่ยน Element ที่อยู่ใน Renderer บน Canvas ซึ่งมีการสร้าง Artist Instance บน matplotlib Figure ได้แก่ The title, the lines, the tick labels, the images รวมไปถึง instance ที่เฉพาะทางในด้านอื่นที่เฉพาะเจาะจงด้วย และมี Artist Object 2 ประเภท ซึ่งประเภทแรกคือ Primitive Type เช่น Line2D, Rectangle, Circle, และ Text และประเภทที่สองคือ Composite Type เช่น Axis, Tick, Axes และ Figure

Scripting Layer เป็น Layer ถูกออกแบบมาใช้ matplotlib สามารถทำงานได้เหมือน Matlab Script โดยจะเป็นการรวบรวม Command ทำให้สามารถใช้งาน Layer นี้ได้ง่ายเหมือนเป็นการเขียนชุดคำสั่งทั่วไปผ่าน Command ที่ matplotlib Library Built-in ไว้ให้

Quality Attributes

Performance Efficiency โดยที่มุ่งเน้นให้ matplotlib จะมีประสิทธิภาพในการทำ Real-Time Plotting โดยใช้จำนวนจริงใด ๆ ในการ Plot องค์ประกอบต่าง ๆ ของ Visualization

Usability โดยที่ matplotlib จะทำการรวบรวม Function สำหรับการทำให้ Visualization ทั้งหมดไว้ใน Scripting Layer สามารถทำให้ผู้ใช้สามารถเรียนรู้การใช้งานได้ง่าย และยังสามารถใช้เครื่องมือต่าง ๆ ที่อยู่ภายใน Library ได้

Integrability จากการที่ matplotlib สามารถทำงานได้ร่วมกับ Python

Library ได้หลากหลาย Library Third-Party Package อาทิเช่น Pandas, NumPy โดยที่สามารถทำงานร่วมกันได้อย่างมีประสิทธิภาพซึ่ง Function ของแต่ละ Library จะทำงานร่วมกันจาก Built-In Function ที่ได้สร้างไว้แล้วจาก Developer

Reference :

<https://ieeexplore.ieee.org/document/4160265>

<https://medium.datadriveninvestor.com/data-visualization-with-python-matplotlib-architecture-6b05af533569>

<https://matplotlib.org/stable/index.html#>

Jitsi

Proposal

Jitsi เป็นชุดของแอปพลิเคชันเสียงการประชุมทางวิดีโอและการส่งข้อความโต้ตอบแบบหลายแพลตฟอร์มฟรีสำหรับแพลตฟอร์มเว็บ Windows, Linux, macOS, iOS และ Android

Architectural Patterns

โดยที่ Architectural Patterns ของ Jitsi เป็นสถาปัตยกรรมแบบ Client-server โดยจะมี ส่วนที่เป็น client จะทำการ request วิดีโอหรือเสียงด้วยการรีโมทและส่วนของ server ที่ทำหน้าที่ติดต่อกับโปรแกรมภายนอก คือ jitsi video bridge ที่ทำหน้าที่ provide วิดีโอคุณภาพสูง

ส่วนของฝั่ง server จะมี component ต่าง ๆ เช่น **Jitsi conference focus** ที่ใช้จัดการสื่อต่าง ๆ และทำหน้าที่เป็น load balancer ระหว่าง ผู้ใช้งานกับ videobridge **jigasi** เป็น component ที่ทำให้ผู้ใช้งานสามารถเข้าร่วมการประชุมได้

Quality Attributes

Usability โดยที่ Software ต้องสามารถใช้งานได้ง่ายมากเนื่องจากเป็นโปรแกรมสำหรับคนกลุ่มใหญ่ มีการใช้งานที่ไม่ยากจนเกินไป และไม่เฉพาะทางเกินไป

Portability โดยที่ jitsi เป็น Software ที่ใช้กันแพร่หลายและการใช้งานควรจะสามารถทำได้จากทุก Platform, Operating System เพื่อสามารถทดสอบการทำงานบนทุกอุปกรณ์ที่ผู้ใช้งานมีได้

Availability โปรแกรมจะสามารถรองรับการประชุมได้ตลอดเวลาและสามารถแก้ไขข้อผิดพลาดได้ภายในเวลาอันสั้น เพื่อให้ผู้ใช้สามารถใช้งานได้อย่างต่อเนื่อง

Reference :

<https://jitsi.org/about/>

<https://jitsi.github.io/handbook/docs/architecture/>