**3) Website architecture** ประกอบด้าน 4 ส่วน ดังภาพ

A picture containing diagram

Description automatically generated

เมื่อมี ผู้ใช้งานจำนวนมาก เข้ามาที่เว็บไซต์ ปรากฎว่าเว็บไซต์ทำงานได้ช้าลงอย่างมาก (แต่ละส่วนอาจเป็นสาเหตุที่เว็บไซช้า)

**ㆍทำ root cause analysis เพื่อหาส่วนที่เว็บไซต์ช้าลง**

3.1) ต้องดำเนินการติดตั้ง หรือทำอะไรที่ส่วนใดของ Website architecture บ้าง เพื่อตรวจหาส่วนที่ทำให้เว็บไซต์ทำงานช้า

ตอบ: ต้องดำเนินการติดตั้งในส่วนของ Tool “Kibana” เพื่อเพื่อตรวจหาส่วนที่ทำให้เว็บไซต์ทำงาน  
  
 3.2) หากดำเนินการตามคำตอบในข้อ 3.1 แล้วจะทราบได้อย่างไรว่าส่วนนั้นเป็นต้นเหตุที่ทำให้เว็บไซต์ช้าลง

ตอบ: โดยเราสามารถที่จะทราบได้จากการดู logs observe จาก Tool “Kibana”

3.3) การดำเนินการตามคำตอบในข้อ 3.1 เลือกใช้ tool ชื่ออะไร เพราะอะไร

ตอบ: Tool ชื่อ Kibana เลือกใช้ tool ชื่อ Grafana เพราะเทียบจาก tools ทั้งหมดในข้อ 3.1 คือ Kibana จะมีข้อดีกว่าคือ tag log observe ได้ง่ายกว่าตัวอื่น

3.4) tool ตามคำตอบในข้อ 3.3 มีข้อดีข้อเสีย ต่างจาก tool ตัวอื่นที่ใช้แทนกันได้อย่างไร

ตอบ: เป้าหมายหลักของระบบนี้ เพื่อสร้างระบบ Visualization เพื่ออธิบายการใช้งานข้อมูล  
ดังนั้นเครื่องมือที่เลือกใช้ต้องสามารถนำข้อมูลไปแสดงในรูปแบบของ graphic ได้ง่าย  
โดยมี 3 ตัวมาเปรียบเทียบกันคือ 1.Kibana, 2. Redash และ 3. Grafana ซึ่งแต่ละตัว ต่างมีข้อดีและข้อเสีย แต่สุดท้ายก็เลือก **Kibana** มาใช้งานด้วยเหตุผลต่าง ๆ ดังนี้

* มีกราฟหลากหลายชนิดให้ใช้
* ยืดหยุ่นต่อการใช้งาน
* รองรับการ filter ข้อมูล
* สามารถใช้ query เพื่อดึงข้อมูลแบบง่าย ๆ

ส่วน Redash และ Grafana นั้นสามารถทำงานได้ดีมาก  
เพียงแต่ไม่ตรงกับความต้องการ หรือ use case นี้เท่านั้นเอง

และอีกเหตุผลหนึ่งก็คือ  
Kibana มันทำงานร่วมกับ Elasticsearch ได้ ซึ่งระบบของ Yelp นั้นใช้ Elasticsearch ในการเก็บข้อมูล feed อยู่แล้ว

**ㆍสรุปสาเหตุที่เว็บไซต์ช้าลง**

3.5) ส่วนใดของ Website architecture (ตามภาพด้านบน ที่อาจทำให้เว็บไซต์ช้าได้บ้างเพราะอะไร)

ตอบ: gateway, server, database

3.6) ปัญหาอะไรในแต่ละส่วนที่กล่าวมา ที่ทำให้เว็บไซต์ช้า

ตอบ:   
1. Gateway อาจจะ manage traffic ไม่ดี load ไปลงอยู่เครื่องเดียว  
2. Server อาจจะไม่พอรองรับ load หรือ logic ใน program มีปัญหาทำให้ response ช้า

3. Database อาจจะไม่ได้มีการทำ index ใน table ที่มีการ query data บ่อยๆ

3. 7) แนวทางการแก้ไขปัญหาในแต่ละส่วนที่กล่าวมา

ตอบ:   
1. Gateway เอาเทคโนโลยีเข้ามาช่วย เช่น kafka

2. Server ออกแบบให้ดี และคำนวนอุปกรณ์ให้เพียงพอ และใช้ performance test เพื่อตรวจสอบ

3. Database design ต้องออกแบบการเขียนให้ดี เช่น ตรวจสอบ query ของ logic program ในการ query บางจุดที่ช้า ให้เร็วขึ้น หรือการเพิ่ม index ที่ table of database เพื่อทำให้ process ในการทำงานเร็วขึ้น ถ้าหลังปรับแล้วยังช้าอยู่ เราสามารถเพิ่ม Resource ของเครื่อง Database ให้มากขึ้น เช่น CPU และ Memory ให้มากขึ้น เพื่อรองรับปริมาณ request มากๆ ได้

**4) จากแผนภาพ ระบบการเติมเงิน ควรจะทดสอบอะไรบ้างและ มีจุดไหนที่ต้อง Concern**

ตัวอย่างแผนภาพ [All level]

Diagram

Description automatically generated

1. ทำการทดสอบในส่วนของการเติมเงินโดย input amount (in case เงินพอและเงินไม่พอ) โดยต้อสามารถ check การ update เงินใน wallet ได้ (Update amount realtime)
2. ทำการทดสอบในส่วนของการเติมเงินโดย สามารถเติมเงินได้จาก ทุกช่องทาง และสามารถ check เงินใน wallet (Update amount realtime)
3. ทำการทดสอบในส่วนของการเติมงานทั้งแบบ Limit per transaction และ limit per day
4. ทำการทดสอบในส่วนของการ payment service response ในช่วงเวลาที่ระบบกำหนด และรองรับ Transaction ที่ถูกต้องเหมาะสม
5. ทำการทดสอบในส่วนของการ payment service in case connecting กับ ธนาคาร ไม่ได้
6. มี Concern **ในส่วนของสถานการณ์ ที่** payment verification in case connecting กับ ธนาคาร ไม่ได้