



การพัฒนาเว็บคอร์ดกีตาร์  
Development of Guitar chord web

นายณภัทร กรดสุวรรณ 664230050

หมู่เรียน 66/46

โครงการนี้เป็นส่วนหนึ่งของการศึกษารายวิชา 7204903

โครงการด้านเทคโนโลยีสารสนเทศ 2

สาขาวิชาเทคโนโลยีสารสนเทศ คณะวิทยาศาสตร์และเทคโนโลยี

มหาวิทยาลัยราชภัฏนครปฐม

ภาคเรียนที่ 1 ปีการศึกษา 2568

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันผู้เล่นกีตาร์มือใหม่ รวมถึงผู้ที่พอเล่นเป็นอยู่แล้ว มักใช้วิธีค้นหาเนื้อเพลงและคอร์ดจากเว็บไซต์หรือหนังสือเพลง เพื่อนำมาฝึกฝนการดีดกีตาร์ให้คล่องขึ้น วิธีการดังกล่าวช่วยให้ผู้เล่นสามารถเรียนรู้การจับคอร์ดและฝึกตามเพลงที่ต้องการได้ แต่ก็มักพบปัญหาว่าผู้ใช้งานต้องใช้หลายขั้นตอน เช่น การค้นหาชื่อเพลงบนเว็บไซต์ แล้วเลือกผลลัพธ์ที่ต้องการ ก่อนที่จะได้เนื้อเพลงและคอร์ดมาใช้งาน

เพื่อแก้ไขปัญหาดังกล่าว เว็บไซต์นี้จึงมุ่งเน้นพัฒนาออกแบบมาให้ใช้งานง่ายมากขึ้น คือ ไม่จำเป็นต้องกรอกชื่อเพลงแล้วกดค้นหาหลายครั้ง แต่สามารถ วางลิงก์ YouTube หรือพิมพ์ชื่อเพลงลงไปโดยตรง ระบบจะดึงข้อมูลเพลงที่เกี่ยวข้อง พร้อมทั้งแสดง เนื้อเพลงและคอร์ดในหน้าเดียวกันโดยอัตโนมัติ

### 1.2 แนวคิดในการแก้ไขปัญหา

จากปัญหาที่กล่าวมาข้างต้น ผู้จัดทำได้ทำเว็บไซต์คอร์ดกีตาร์โดยใช้แนวคิดในการเชื่อมต่อ API ของ YouTube เพื่อดึงข้อมูลเพลงที่ผู้ใช้กรอกเข้ามา ไม่ว่าจะเป็นชื่อเพลงหรือลิงก์ จากนั้นจึงนำข้อมูลไปค้นหาคอร์ดที่เกี่ยวข้องจากแหล่งข้อมูลออนไลน์ด้วย Custom Search หรือเทคนิค “Web Scraping” แล้ว ใช้ Gemini API ทำหน้าที่วิเคราะห์และประมวลผลข้อมูลที่ค้นหาได้

### 1.3 วัตถุประสงค์ของระบบ

- 1.3.1 เพื่อพัฒนาเว็บไซต์ สำหรับค้นหาและแสดงคอร์ดเพลง โดยสามารถรับข้อมูลนำเข้าเป็นชื่อเพลงหรือลิงก์จากYouTube
- 1.3.2 เพื่อใช้ YouTube API, Google Custom Search และ Gemini API ในการดึง ตรวจสอบ และประมวลผลข้อมูลคอร์ดเพลง

## 1.4 ขอบเขตการศึกษา

### 1.4.1 ขอบเขตของระบบ

#### 1.4.1.1 ผู้ใช้งานระบบ

- ก) สามารถกรอก ลิงก์ยูทูปและชื่อเพลงได้
- ข) สามารถดูเนื้อเพลงและคอร์ดเพลงได้ในหน้าเดียวกัน

### 1.4.2 ฮาร์ดแวร์ที่ใช้ในการพัฒนา

#### 1.4.2.1 เครื่องคอมพิวเตอร์

- ก) หน่วยประมวลผลกลาง Intel Core i5 12400F
- ข) หน่วยความจำหลักความจุ 32 กิกะไบต์
- ค) อุปกรณ์จัดเก็บข้อมูลขนาด 512 กิกะไบต์
- ง) หน่วยประมวลผลกราฟฟิก NVIDIA GTX 1660

### 1.4.3 ซอฟต์แวร์ที่ใช้ในการพัฒนา

- 1.4.3.1 ระบบปฏิบัติการ ไมโครซอฟท์ วินโดวส์ เท็น
- 1.4.3.2 วิซวล สตูดิโอ โค้ดใช้เป็นเครื่องมือในการพัฒนา
- 1.4.3.3 เอช ที เอ็ม แอล ไฟฟ์ ซี เอส เอสสำหรับการเขียนหน้าเว็บ
- 1.4.3.4 ไฟร์เบส สตูดิโอ
- 1.4.3.5 เจมีไนน์ เอพีไอ ช่วยวิเคราะห์ข้อมูลที่ได้รับ
- 1.4.3.6 ยูทูป เดต้า เอพีไอ เวอร์ชัน 3 นำเข้ามาเพื่อเก็บชื่อเพลงและชื่อศิลปิน
- 1.4.3.7 คัส ต้อม เสิร์ช เอพีไอ ใช้ในการระบุเว็บแบบเจาะจงที่จะดึงข้อมูล
- 1.4.3.8 ไพ ทอน ใช้สำหรับการเขียนโปรแกรม

### 1.4.4 ข้อกำหนดของระบบ

- 1.4.4.1 ระบบแสดงคอร์ดเพลงได้เฉพาะเพลงที่มีอยู่ในเว็บไซต์ Chordzaa
- 1.4.4.2 ความแม่นยำในการระบุตำแหน่งคอร์ด มีจุดที่เปลี่ยนคอร์ดไม่ตรงกับจังหวะของเพลงจริง

## 1.5 ประโยชน์ที่ได้คาดว่าจะได้รับ

- 1.5.1 ผู้ใช้สามารถค้นหาเนื้อเพลงและคอร์ดเพลงได้ง่ายขึ้น
- 1.5.2 ช่วยลดเวลาในการค้นหาเนื้อเพลงและคอร์ดเพลง

## 1.6 คำนิยาม

1.6.1 ผู้ใช้งาน หมายถึง บุคคลทั่วไปที่ใช้เว็บไซต์เพื่อใช้งานต่างๆ เช่น การส่งข้อมูลลิงก์ชื่อเพลง และดูผลลัพธ์จากหน้าเว็บ

1.6.2 ยูทูบ เดต้า เอพีไอ(Youtube Data API) หมายถึง ส่วนที่ใช้เชื่อมต่อกับข้อมูลของยูทูบ เพื่อให้ระบบได้รับข้อมูลชื่อเพลงและศิลปิน

1.6.3 คัส ต่อม เซิร์ช เอพีไอ (Custom Search API) หมายถึง ส่วนที่ใช้เจาะจงเว็บไซต์ที่จะนำมาดึงข้อมูลให้ระบบ

1.6.4 ยูอาร์แอล (URL) หมายถึง ส่วนที่ได้ข้อมูลมาจาก คัส ต่อม เซิร์ช เอพีไอ

1.6.5 เอช ที เอ็ม แอล (HTML) หมายถึง ส่วนของข้อมูลในหน้า ยูอาร์แอลที่ได้มาจากคัส ต่อม เซิร์ช เอพีไอ

1.6.6 เจมินิ เอพีไอ (Gemini API) หมายถึง ส่วนของ การวิเคราะห์ข้อมูล เช่น คอร์ดและเนื้อเพลง

## บทที่ 2

### หลักการและทฤษฎีที่เกี่ยวข้อง

ในการพัฒนาโครงงาน การพัฒนาเว็บคอร์คิตาร์ ผู้จัดทำได้ทำการศึกษา รวบรวมหลักการ ทฤษฎี รวมถึงสำรวจระบบงานอื่นที่มีลักษณะคล้ายคลึงกัน เพื่อใช้เป็นองค์ความรู้พื้นฐานในการ ออกแบบและพัฒนาระบบให้สามารถทำงานได้ตามวัตถุประสงค์ที่กำหนดไว้ โดยมีรายละเอียดดังนี้

#### 2.1 ระบบงานเดิม

ในปัจจุบัน ผู้ที่ต้องการค้นหาคอร์คิตาร์เพื่อฝึกเล่นกีตาร์มักต้องดำเนินการหลายขั้นตอน กระบวนการโดยทั่วไปเริ่มต้นจากการใช้เครื่องมือค้นหา (Search Engine) เช่น Google เพื่อค้นหา ชื่อเพลงที่ต้องการ จากนั้นผู้ใช้จะต้องเลือกเว็บไซต์จากผลลัพธ์การค้นหา ซึ่งมีอยู่หลากหลาย เมื่อเข้าไปยังเว็บไซต์เป้าหมายแล้ว ผู้ใช้อาจต้องค้นหาเพลงนั้นซ้ำอีกครั้งภายในเว็บจึงจะสามารถเข้าถึงเนื้อ เพลงและคอร์คิตาร์ได้

#### 2.2 ระบบงานอื่นที่เกี่ยวข้อง

สุทิน อุทธรณ์ (2560) งานนิพนธ์นี้นำเสนอผลการศึกษาและประยุกต์ใช้วิธีการสกัดข้อมูล บนเว็บเพจด้วยเว็บครอว์เลอร์ (Web Crawler) จากเว็บไซต์ที่รวบรวมงานวิจัยต่างๆ มาแสดงไว้บน เว็บไซต์เดียวกัน เพื่อให้สามารถค้นหาได้สะดวกและรวดเร็วยิ่งขึ้น โดยใช้ภาษาและเครื่องมือที่ปรากฏ อยู่ในปัจจุบัน ได้แก่ Node.js และ Cheerio ซึ่งเป็นเครื่องมือที่สามารถดึงข้อมูลจากเว็บไซต์ด้วย วิธีการเข้าถึงโครงสร้าง HTML ของเว็บไซต์นั้นๆ เพื่อสกัดข้อมูลที่ต้องการ และจัดเก็บข้อมูลที่ได้ลงใน ฐานข้อมูลสำหรับนำไปสร้างส่วนแสดงผลต่อไป

พงษ์สิน จันทรแก้ว , วชิราวุธ ธรรมวิเศษ (2568) ได้ทำการพัฒนาระบบเปรียบเทียบราคา สินค้าอีคอมเมิร์ซและแบ่งกลุ่มสินค้าที่คล้ายกันโดยใช้เทคนิค Web Scraping ระบบดังกล่าวถูก พัฒนาด้วยภาษาไพทอนเพื่อดึงข้อมูลผลิตภัณฑ์จากเว็บไซต์อีคอมเมิร์ซที่หลากหลาย นอกจากนี้ยังใช้ เทคนิคขั้นสูงอย่าง TF-IDF และ Convolutional Neural Networks (CNN) ในการสกัดคุณลักษณะ ของชื่อและรูปภาพผลิตภัณฑ์เพื่อนำมาจัดกลุ่มสินค้าที่คล้ายกัน ผลที่ได้จากการพัฒนาคือระบบที่ สามารถดึงข้อมูลจากเว็บไซต์เป้าหมายได้สำเร็จ และการจัดกลุ่มสินค้าที่คล้ายกันมีประสิทธิภาพสูงขึ้น เมื่อเทียบกับการใช้เพียงชื่อหรือรูปภาพเพียงอย่างเดียว

## 2.3 องค์ความรู้ที่เกี่ยวข้อง

### 2.3.1 วิซวล สตูดิโอ โค้ด (Visual Studio Code)

เป็นโปรแกรมแก้ไขซอร์สโค้ดที่พัฒนาโดย ไมโครซอฟท์ มีการสนับสนุนสำหรับการดีบั๊ก การควบคุม Git ในตัวและ GitHub การเน้นไวยากรณ์ การเติมโค้ดอัจฉริยะ ตัวอย่าง และ code refactoring มันสามารถปรับแต่งได้หลายอย่าง ให้ผู้ใช้สามารถเปลี่ยนธีม แบนพิมพ์ลัด การตั้งค่า และติดตั้งส่วนขยายที่เพิ่มฟังก์ชันการทำงานเพิ่มเติม ดังภาพที่ 2.1



ภาพที่ 2.1 วิซวล สตูดิโอ โค้ด

ที่มา: <https://logospng.org/logo-visual-studio-code/>

### 2.3.2 เอชทีเอ็มแอล 5 (HTML5)

ภาษา HTML เวอร์ชันที่ 5 ซึ่งเป็นมาตรฐานล่าสุดที่ใช้สร้างและนำเสนอเนื้อหาบนเว็ลต์ไวด์เว็บ มีการพัฒนาและปรับปรุงจาก HTML เวอร์ชันก่อนหน้าอย่างมาก โดยเพิ่มคุณสมบัติใหม่ๆ ที่ช่วยให้นักพัฒนาสามารถสร้างเว็บไซต์ที่ซับซ้อนและรองรับมัลติมีเดียได้ดีขึ้น เช่น การรองรับวิดีโอและเสียงโดยตรง, การวาดกราฟิก, และการทำงานแบบออฟไลน์ ดังภาพที่ 2.2



ภาพที่ 2.2 เอชทีเอ็มแอล 5

ที่มา: <https://en.wikipedia.org/wiki/HTML5>

### 2.3.3 ระบบปฏิบัติการ ไมโครซอฟท์ วินโดวส์ เท็น (Microsoft Windows 10)

Windows ระบบปฏิบัติการที่ผู้ใช้งานใช้เยอะมากที่สุดจากทางไมโครซอฟท์ (Microsoft) ที่ถูกออกแบบให้สามารถใช้งานได้กับทุกอุปกรณ์ที่ใช้ระบบปฏิบัติการ Windows ในทุกขนาดหน้าจอ ทุกรูปแบบการใช้งาน ไม่ว่าจะเป็นหน้าจอแบบระบบสัมผัส แท็บเล็ต หรืออุปกรณ์ควบคุมอย่างเมาส์หรือคีย์บอร์ด รวมไปถึงยังมีระบบแอปพลิเคชันครอบจักรวาล (Universal Application) ที่ให้แอปฯ ต่างๆ นั้นสามารถใช้งานได้บนทุกอุปกรณ์ในระบบ Windows 10 อีกด้วยเช่นกัน ดังภาพที่ 2.3



ภาพที่ 2.3 ไมโครซอฟท์ วินโดวส์ เท็น

ที่มา: [https://en.wikipedia.org/wiki/Windows\\_10](https://en.wikipedia.org/wiki/Windows_10)

### 2.3.4 ซีเอสเอส (CSS)

เป็นภาษาที่ใช้ในการออกแบบและจัดรูปแบบเว็บไซต์ โดยใช้ควบคู่กับ HTML (Hypertext Markup Language) ที่เป็นภาษาหลักในการสร้างโครงสร้างของเว็บไซต์ CSS มีบทบาทสำคัญในการแยกความรับผิดชอบระหว่างโครงสร้างเนื้อหาและการนำเสนอ ทำให้เว็บไซต์มีความสวยงามและใช้งานง่ายขึ้น CSS ย่อมาจากคำว่า “Cascading Style Sheets” ซึ่งแปลเป็นไทยได้ว่า “สไตล์ชีตแบบซ้อนทับ” คือ ภาษาที่ใช้กำหนดรูปแบบการแสดงผลของ HTML โดยใช้หลักการของการกำหนดกฎเกณฑ์หรือ “กฎของ CSS” (CSS Rules) เพื่อควบคุมลักษณะการแสดงผลขององค์ประกอบต่างๆ ในหน้าเว็บ เช่น ขนาดตัวอักษร สีพื้นหลัง ขอบของกล่อง และการจัดวางองค์ประกอบ ดังภาพที่ 2.4



ภาพที่ 2.4 ซีเอสเอส

ที่มา: [https://en.wikipedia.org/wiki/CSS#/media/File:Official\\_CSS\\_Logo.svg](https://en.wikipedia.org/wiki/CSS#/media/File:Official_CSS_Logo.svg)

### 2.3.5 ไฟร์เบส สตูดิโอ (Firebase Studio)

Firebase Studio เป็นเครื่องมือพัฒนาใหม่ที่ขับเคลื่อนด้วย AI ของ Google ซึ่งช่วยให้คุณสร้างแอปพลิเคชันโดยใช้ข้อความแจ้งภาษาธรรมชาติ จริงๆ แล้วเป็นวิวัฒนาการของ Project IDX ซึ่งเป็น IDE บนคลาวด์ที่ Google เปิดตัวในปี 2023 Project IDX มุ่งเน้นไปที่การเป็น IDE บนคลาวด์เอนกประสงค์ที่รองรับ AI แต่การรีแบรนด์เป็น Firebase Studio ทำให้โฟกัสแคบลงเพื่อเพิ่มประสิทธิภาพประสบการณ์สำหรับนักพัฒนา Firebase ใช้ส่วนที่ดีที่สุดของ IDX ดังภาพที่ 2.5



ภาพที่ 2.5 ไฟร์เบส สตูดิโอ

ที่มา: <https://aipill.io/firebase-studio/>

### 2.3.6 เจมินี เอพีไอ (Gemini API)

เจมินี เอพีไอ ชุดเครื่องมือและโมเดลภาษาขนาดใหญ่ (LLM) จาก Google ที่ช่วยให้นักพัฒนาสามารถเข้าถึงความสามารถของ AI ในการสร้างแอปพลิเคชันต่างๆ ได้ โดย Gemini API รองรับการประมวลผลข้อมูลได้หลายรูปแบบ เช่น ข้อความ รูปภาพ เสียง วิดีโอ และโค้ด. นักพัฒนาสามารถใช้ Gemini API เพื่อสร้างสรรค์ผลิตภัณฑ์ที่ใช้ AI ได้อย่างหลากหลาย ตั้งแต่การสร้างเนื้อหาอัตโนมัติไปจนถึงการวิเคราะห์ข้อมูลที่ซับซ้อน ดังภาพที่ 2.6





ภาพที่ 2.6 เจมินน์ เอพีไอ

ที่มา: [https://onlinepartner.se/info/sa-skriver-du-promptar-till-google-gemini-workspace/gemini\\_primary\\_logo\\_full-color\\_rgb2x/](https://onlinepartner.se/info/sa-skriver-du-promptar-till-google-gemini-workspace/gemini_primary_logo_full-color_rgb2x/)

### 2.3.7 ยูทูบ เดต้า เอพีไอ เวอร์ชัน 3 (Youtube Data API v3)

ชุดเครื่องมือที่ให้นักพัฒนาสามารถเข้าถึงและจัดการข้อมูลของ YouTube ได้โดยการส่งคำขอผ่านอินเทอร์เฟซโปรแกรมประยุกต์ (API) เพื่อสร้างแอปพลิเคชันที่ทำงานร่วมกับ YouTube ได้ เช่น การดึงข้อมูลวิดีโอ การค้นหา การจัดการเพลย์ลิสต์ หรือการอัปโหลดวิดีโอ โดย API จะช่วยให้โปรแกรมเมอร์สามารถรวมฟังก์ชันต่างๆ ของ YouTube เข้ากับเว็บไซต์หรือแอปพลิเคชันของตนเองได้ ดังภาพที่ 2.7

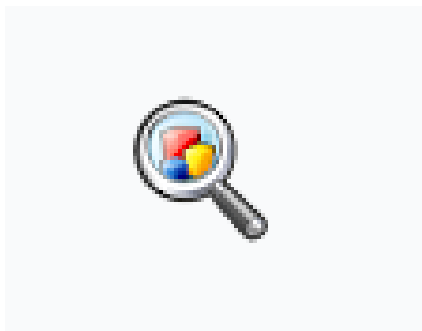


ภาพที่ 2.7 ยูทูบ เดต้า เอพีไอ เวอร์ชัน 3

ที่มา: <https://fr.vecteezy.com/png-gratuit/youtube-logo>

### 2.3.8 คัสต้อม เสิร์ช เอพีไอ (Custom Search API)

คัสต้อม เสิร์ช เอพีไอ (Custom Search API) คือ API ที่ช่วยให้คุณสร้างฟังก์ชันการค้นหาแบบเฉพาะเจาะจงสำหรับเว็บไซต์หรือแอปพลิเคชันของคุณเองได้ โดยสามารถปรับแต่งการตั้งค่าต่างๆ ได้เอง เช่น การเลือกผลลัพธ์ที่จะแสดง, การกำหนดเกณฑ์การค้นหา ดังภาพที่ 2.8



ภาพที่ 2.8 คัสต้อม เซิร์ช เอพีไอ

ที่มา:

<https://console.cloud.google.com/apis/api/customsearch.googleapis.com/metrics?authuser=2&project=ivory-amphora-467909-t4>

### 2.3.9 ไพทอน (Python)

ภาษาโปรแกรม Python คือภาษาโปรแกรมคอมพิวเตอร์ระดับสูง โดยถูกออกแบบมาให้เป็นภาษาสคริปต์ที่อ่านง่าย โดยตัดความซับซ้อนของโครงสร้างและไวยากรณ์ของภาษาออกไป ในส่วนของการแปลงชุดคำสั่งที่เราเขียนให้เป็นภาษาเครื่อง Python มีการทำงานแบบ Interpreter คือเป็นการแปลชุดคำสั่งทีละบรรทัด เพื่อป้อนเข้าสู่หน่วยประมวลผลให้คอมพิวเตอร์ทำงานตามที่เราต้องการ นอกจากนั้นภาษาโปรแกรม Python ยังสามารถนำไปใช้ในการเขียนโปรแกรมได้หลากหลายประเภท โดยไม่ได้จำกัดอยู่ทำงานเฉพาะทางใดทางหนึ่ง (General-purpose language) จึงทำให้มีการนำไปใช้กันแพร่หลายในหลายองค์กรใหญ่ระดับโลก เช่น Google, YouTube, Instagram, Dropbox และ NASA เป็นต้น ดังภาพที่ 2.9



ภาพที่ 2.9 ไพทอน

ที่มา: <https://www.fiverr.com/tplueer/create-a-python-api-for-you>

## บทที่ 3

### วิธีการดำเนินงาน

บทนี้เป็นการอธิบายขั้นตอนและวิธีการดำเนินงานในการพัฒนาระบบ "เว็บคอร์ดกีตาร์" เพื่อให้การทำงานเป็นไปอย่างมีระบบและบรรลุวัตถุประสงค์ที่ตั้งไว้ โดยมีรายละเอียดในแต่ละขั้นตอนดังนี้

#### 3.1 การศึกษาเบื้องต้น

จากการศึกษาข้อมูลเบื้องต้น ระบบงานเดิมที่ผู้ใช้ส่วนใหญ่ใช้ในการค้นหาพบว่า

3.1.1 ระบบงานเดิมที่ผู้ใช้ส่วนใหญ่ใช้ในการค้นหาคอร์ดกีตาร์มีขั้นตอนที่เยอะ

3.1.1.1 ผู้ใช้ต้องเริ่มจากการค้นหาชื่อเพลงผ่านเครื่องมือค้นหาทั่วไป

3.1.1.2 เลือกเว็บไซต์จากผลลัพธ์ที่หลากหลาย

3.1.1.3 อาจต้องค้นหาซ้ำภายในเว็บไซต์นั้นอีกครั้งกว่าจะได้ข้อมูลที่ต้องการ

3.1.2 ระบบงานใหม่

ระบบใหม่ถูกออกแบบมาเพื่อแก้ไขปัญหาดังกล่าว โดยผู้ใช้สามารถกรอกชื่อเพลงหรือวงลึกลงได้โดยตรง ระบบจะทำการดึงข้อมูล, ค้นหา, และประมวลผลคอร์ดเพลงมาแสดงผลในหน้าเดียวโดยอัตโนมัติ ซึ่งช่วยลดขั้นตอนในการใช้งานลง

#### 3.2 การกำหนดความต้องการของระบบ

เพื่อให้การพัฒนาระบบเป็นไปอย่างมีทิศทาง ได้มีการกำหนดความต้องการของระบบในด้านต่างๆ ดังนี้

3.2.1 ขอบเขตของระบบที่สามารถทำได้

3.2.1.1 สามารถกรอกชื่อเพลง หรือ วงลึกลงในช่องค้นหาได้

3.2.1.2 ดูเนื้อเพลงและคอร์ดกีตาร์ที่ผ่านการดึงข้อมูลแล้วได้ในหน้าเดียวกัน

3.2.2 ขอบเขตของระบบที่ไม่สามารถทำได้

3.2.2.1 เพลงใหม่ที่เว็บChordzaaยังไม่ได้นำเข้า

3.2.2.2 จังหวะจุดที่เปลี่ยนคอร์ดไม่ตรงกับจังหวะจริง

3.2.2.3 วาล์วจากแพลตฟอร์มอื่น

### 3.2.3 ฮาร์ดแวร์ที่ใช้กับระบบงาน

#### 3.2.3.1 ฮาร์ดแวร์ฝั่งเซิร์ฟเวอร์

- ก) ระบบปฏิบัติการ วินโดวส์ เท็น
- ข) หน่วยประมวลผลกลาง Intel Core i5 6400
- ค) หน่วยความจำหลักความจุ 4 กิกะไบต์
- ง) อุปกรณ์จัดเก็บข้อมูลขนาด 12 กิกะไบต์

### 3.2.4 ซอฟต์แวร์ที่ใช้กับระบบงาน

#### 3.2.4.1 ระบบปฏิบัติการ: ไมโครซอฟท์ วินโดวส์ เท็น (Microsoft Windows 10)

#### 3.2.4.2 เครื่องมือพัฒนา: วิวส สตูดิโอ โค้ด (Visual Studio Code)

#### 3.2.4.3 ภาษาที่ใช้พัฒนา: ไพทอน (Python), เอชทีเอ็มแอล (HTML5) และ ซีเอสเอส (CSS)

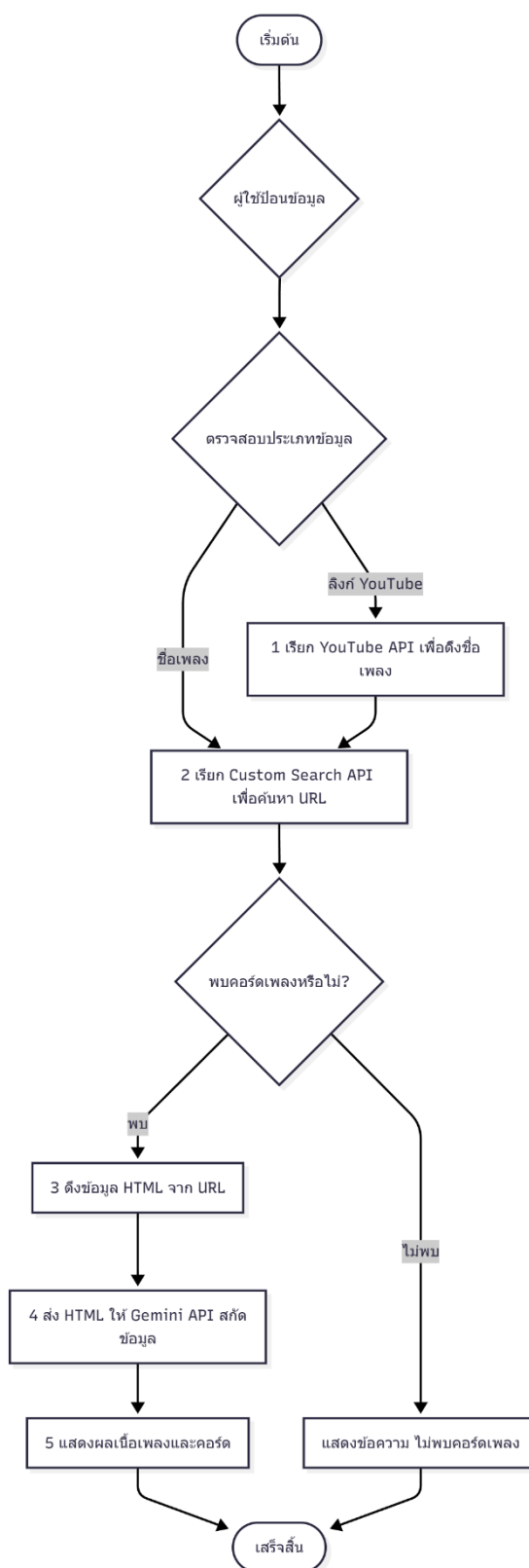
#### 3.2.4.4 API และบริการที่เกี่ยวข้อง

- ก) ยูทูบ เดต้า เอพีไอ (Youtube Data API v3)
- ข) คัสต้อม เสิร์ช เอพีไอ (Custom Search API)
- ค) เจมินไนน์ เอพีไอ (Gemini API)
- ง) ไฟร์เบส (Firebase)

## 3.3 การออกแบบระบบ

การออกแบบระบบเว็บคอร์คิตาร์ ประกอบไปด้วยการออกแบบระบบ

## 3.3.1 การออกแบบระบบ



ภาพที่ 3.10 ภาพรวมการทำงานของระบบ

เมื่อผู้ใช้ป้อนชื่อเพลงหรือลิงก์จาก Youtube ระบบจะทำงานดังนี้

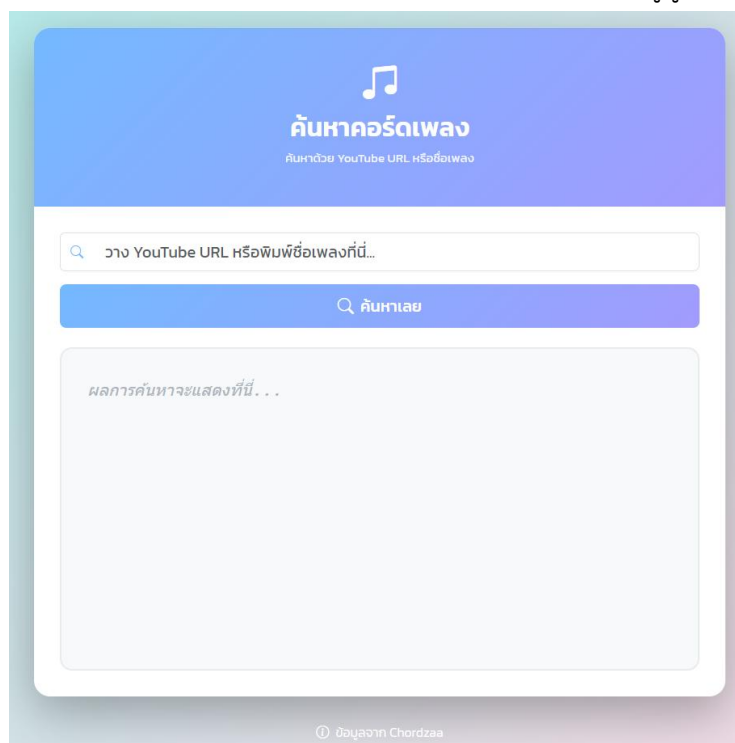
- 1) ถ้าเป็นลิงก์ Youtube Python จะ ขอข้อมูลจาก Youtube Data API เพื่อเก็บชื่อเพลง
- 2) นำชื่อเพลงที่ได้ไปค้นหา URL ด้วย Google Custom Search API
- 3) Python ดึงข้อมูล HTML จาก URL ที่ได้จาก Google Custom Search API
- 4) ส่ง HTML ที่ได้ไปให้ Gemini API เพื่อสกัดข้อมูลเนื้อเพลงและคอร์ด
- 5) หน้าเว็บเพจ แสดงผลลัพธ์ คือเนื้อเพลงและคอร์ดมาให้ผู้ใช้

### 3.4 การออกแบบส่วนติดต่อกับผู้ใช้

การออกแบบส่วนติดต่อกับผู้ใช้สำหรับระบบ เว็บคอร์ดกีตาร์ มุ่งเน้นความเรียบง่าย และ การใช้งานที่สะดวก เพื่อให้ผู้ใช้สามารถค้นหาคอร์ดเพลงที่ต้องการได้อย่างรวดเร็วที่สุด โดยลด ขั้นตอนที่ซ้ำซ้อน การออกแบบจะประกอบด้วยหน้าจอหลักเพียงหน้าจอเดียว ซึ่งรวมส่วนนำเข้าสู่ ข้อมูลและส่วนแสดงผลพร้อมไว้ด้วยกัน เพื่อให้ผู้ใช้ได้รับประสบการณ์การใช้งานที่ต่อเนื่องและราบรื่น

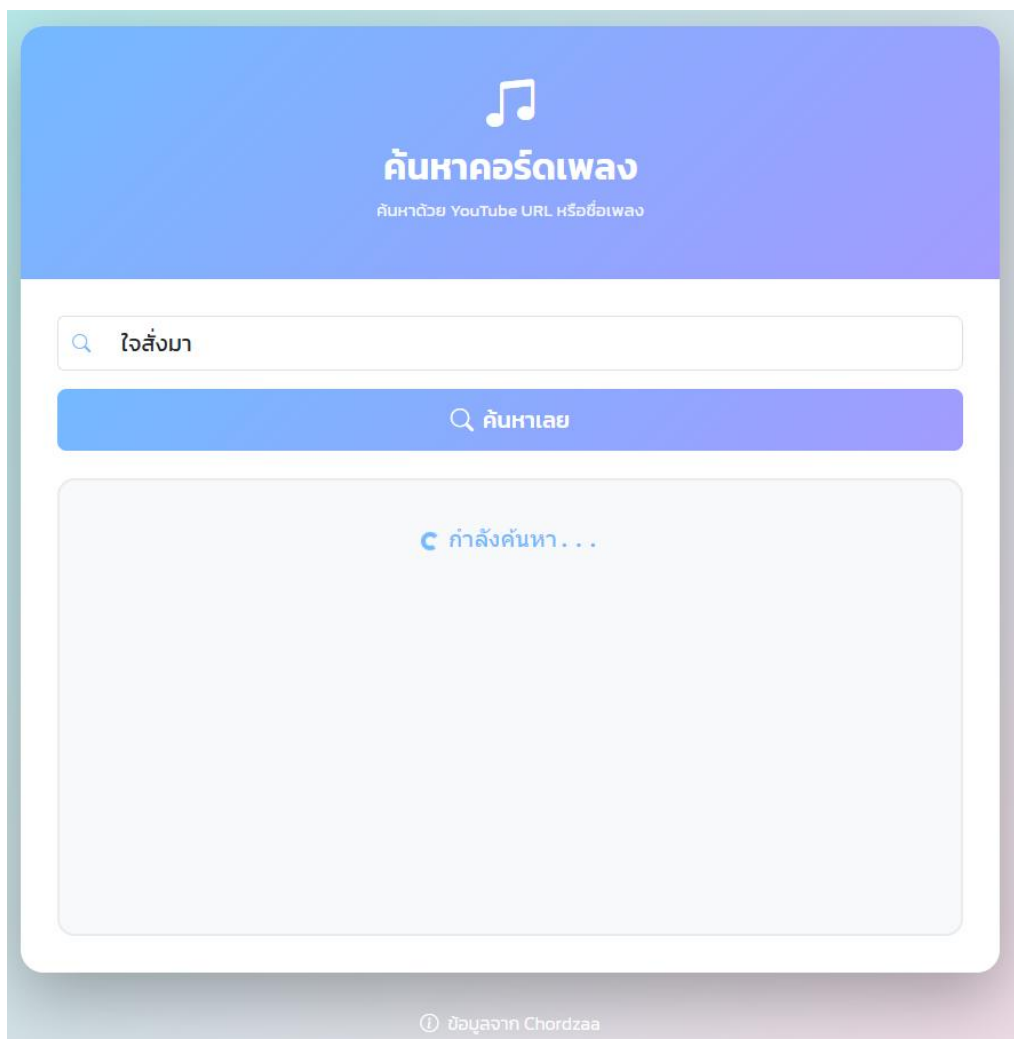
#### 3.4.1 การออกแบบหน้าจอหลัก

- 1) การเริ่มต้นการใช้งานสามารถพิมพ์ชื่อเพลงหรือวางลิงก์ยูทูปได้ ดังภาพที่ 3.11



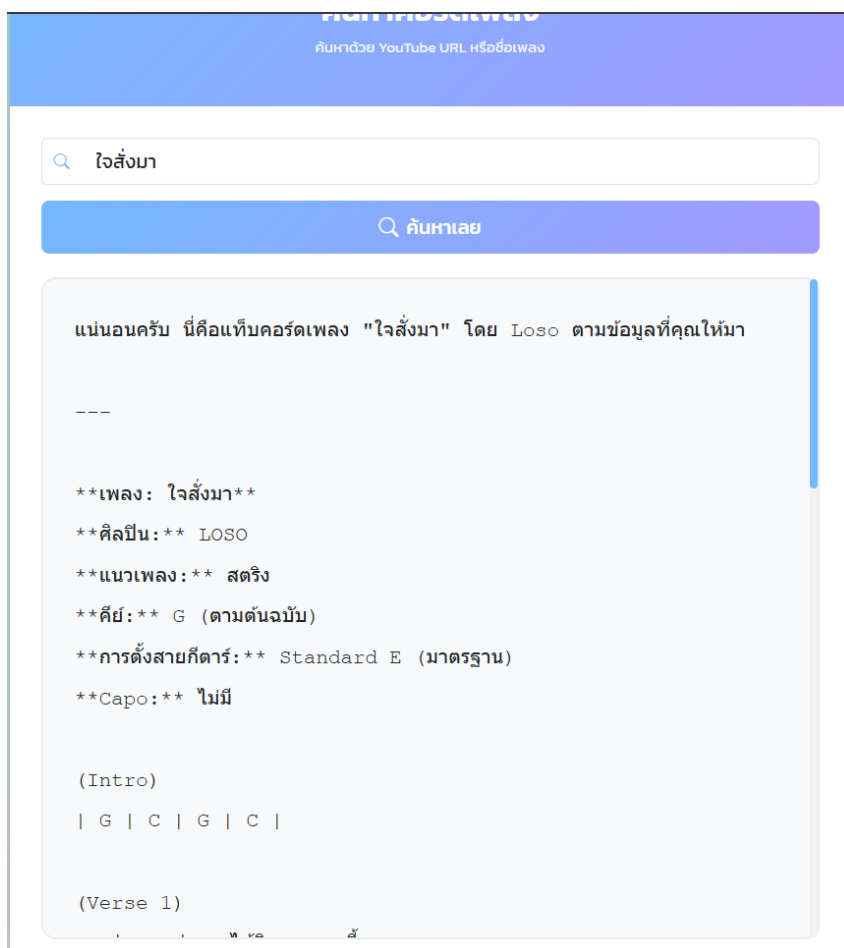
ภาพที่ 3.11 หน้าจอหลัก

2) เมื่อผู้ใช้พิมพ์ชื่อเพลงหรือวางลิงก์ยูทูประบบจะประมวลผล ดังภาพที่ 3.12



ภาพที่ 3.12

3) ระบบจะแสดงผลที่ได้ทางหน้าจอหลัก ดังภาพที่ 3.13



ภาพที่ 3.13



## บรรณานุกรม

สุทิน อุทธบูรณ์. (2560). กรณีศึกษาการสกัดข้อมูลงานวิจัยบนเว็บเพจด้วยเว็บครอว์เลอร์. (งานนิพนธ์ปริญญาวิทยาศาสตรมหาบัณฑิต). มหาวิทยาลัยบูรพา.

พงษ์สิน จันทรแก้ว และ วชิราวุธ ธรรมวิเศษ. (2568). ระบบเปรียบเทียบราคาสินค้าอีคอมเมิร์ซและแบ่งกลุ่มสินค้าที่คล้ายกันโดยใช้เทคนิค Web Scraping. Journal of Applied Informatics and Technology, 7(2), 136–152.