



COMPATH: เว็บแอปพลิเคชันสำหรับนักศึกษาวิศวกรรมคอมพิวเตอร์เพื่อปรับปรุงการพัฒนาตนเองและเรซูเม่
COMPATH: WEB APPLICATION FOR CPE STUDENTS TO IMPROVE THEIR SELF-DEVELOPMENT AND RESUME

63070501025 NIWATCHAI WANGTRAKULDEE

63070501038 NAPAT WAREEDEE

63070501039 NARITH THANOMSUP

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR
THE DEGREE OF BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)
FACULTY OF ENGINEERING
KING MONGKUT'S UNIVERSITY OF TECHNOLOGY THONBURI
2023

Compath: เว็บแอปพลิเคชันสำหรับนักศึกษาวิศวกรรมคอมพิวเตอร์เพื่อปรับปรุงการพัฒนาตนเองและเรซูเม่
Compath: Web application for CPE students to improve their self-development and resume

63070501025 Niwatchai Wangtrakuldee

63070501038 Napat Wareedee

63070501039 Narith Thanomsup

A Project Submitted in Partial Fulfillment
of the Requirements for
the Degree of Bachelor of Engineering (Computer Engineering)
Faculty of Engineering
King Mongkut's University of Technology Thonburi
2023

Project Committee

..... (Asst.Prof.Dr. Khajonpong Akkarajitsakul, Ph.D.)	Project Advisor
..... (Asst.Prof.Dr. Phond Phunchongham, Ph.D.)	Committee Member
..... (Asst.Prof.Dr. Priyakorn Pusawiro, Ph.D.)	Committee Member
..... (Assoc.Prof.Dr Thumrongrat Amornraksa, Ph.D.)	Committee Member

Copyright reserved

Project Title	Compath: เว็บแอปพลิเคชันสำหรับนักศึกษาวิศวกรรมคอมพิวเตอร์เพื่อปรับปรุงการพัฒนาดตนเองและเรซูเม่ Compath: Web application for CPE students to improve their self-development and resume
Credits	3
Member(s)	63070501025 Niwatchai Wangtrakuldee 63070501038 Napat Wareedee 63070501039 Narith Thanomsup
Project Advisor	Asst.Prof.Dr. Khajonpong Akkarajitsakul, Ph.D.
Program	Bachelor of Engineering
Field of Study	Computer Engineering
Department	Computer Engineering
Faculty	Engineering
Academic Year	2023

Abstract

Wait for information...

Keywords: Test

หัวข้อปริญญานิพนธ์	Compath: เว็บแอปพลิเคชันสำหรับนักศึกษาวิศวกรรมคอมพิวเตอร์เพื่อปรับปรุงการพัฒนาดตนเองและเรซูเม่ Compath: Web application for CPE students to improve their self-development and resume
หน่วยกิต	3
ผู้เขียน	63070501025 นาณิวัฒน์ชัย หวังตระกูลดี 63070501038 นายณภัทร วาริตี 63070501039 นายณริศ ธนอมทรัพย์
อาจารย์ที่ปรึกษา	ผศ. ดร.ขจรพงษ์ อัครจิตสกุล
หลักสูตร	วิศวกรรมศาสตรบัณฑิต
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมคอมพิวเตอร์
คณะ	วิศวกรรมศาสตร์
ปีการศึกษา	2566

บทคัดย่อ

รอข้อมูลในอนาคต...

คำสำคัญ: ทดสอบ

กิตติกรรมประกาศ

ขอบคุณอาจารย์ที่ปรึกษา กรรมการ พ่อแม่พี่น้อง และเพื่อนๆ คนที่ช่วยให้งานสำเร็จ ตามต้องการ

สารบัญ

หน้า

ABSTRACT	ii
บทคัดย่อ	iii
กิตติกรรมประกาศ	iv
สารบัญ	v
สารบัญตาราง	viii
สารบัญรูปภาพ	ix
สารบัญสัญลักษณ์	x
สารบัญคำศัพท์ทางเทคนิคและคำย่อ	xi
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของโครงการ	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
1.5 ตารางการดำเนินงาน	2
บทที่ 2 ทฤษฎีความรู้และงานที่เกี่ยวข้อง	4
2.1 ทฤษฎีที่เกี่ยวข้อง	4
2.1.1 Human Computer Interaction (HCI)	4
2.1.2 Design Thinking	4
2.1.3 MVC Structure	5
2.1.4 The 5 Users design process	5
2.1.5 Usability Heuristics Principles	6
2.1.6 Artificial Neural Network (ANN)	7
2.1.7 Recurrent Neural Network (RNN)	7
2.1.8 Long Short Term Memory (LSTM)	8
2.1.9 SQL และ NoSQL	9
2.2 อัลกอริทึมในการแปลผลภาษา	9
2.2.1 Term Frequency Inverse Document Frequency (TF-IDF)	9
2.3 อัลกอริทึมในการแยกประเภทเรซูเม่	10
2.3.1 อัลกอริทึม I K-Nearest Neighbors (KNN)	10
2.3.2 อัลกอริทึม II Naive Bayes Classifier	10
2.4 ภาษาคอมพิวเตอร์และเทคโนโลยี	11
2.4.1 เครื่องมือในการทำโครงการ	11
2.4.1.1 Figma	11
2.4.1.2 Jira	11
2.4.2 ภาษาโปรแกรมที่ใช้	11
2.4.2.1 Typescript	11
2.4.2.2 Python	11
2.4.3 เทคโนโลยีจัดการระบบหน้าบ้าน (Front-End)	12
2.4.3.1 NextJS	12
2.4.4 เทคโนโลยีจัดการระบบหลังบ้าน (Back-End)	12
2.4.4.1 NestJS	12
2.4.5 บริการคลาวด์ (Cloud Service)	12

2.4.5.1	Google Cloud Platform (GCP)	12
2.4.6	ระบบฐานข้อมูล	12
2.4.6.1	MongoDB	12
2.4.7	เครื่องมือช่วยเหลือการพัฒนา (CI/CD Management)	12
2.4.7.1	Docker	12
2.4.7.2	Github	12
2.5	การศึกษาข้อมูลผลิตภัณฑ์ที่เกี่ยวข้อง	13
2.5.1	หนังสือหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ มจร.	13
2.5.2	LinkedIn	13
2.5.3	JobDB	13
2.5.4	Padlet	14
2.5.5	JobThai	14
2.5.6	Workday	14
2.5.7	Camphub	15
2.5.8	Fuel50	15
2.5.9	Super Resume	15
2.5.10	JobHack (Resume Checker)	16
2.5.11	Competitor Analysis	16
บทที่ 3	การออกแบบและวิธีการดำเนินงาน	17
3.1	การสำรวจความต้องการกับผู้ใช้	17
3.1.1	การสัมภาษณ์เชิงปริมาณผ่านแบบสำรวจ	17
3.1.2	การสัมภาษณ์เชิงคุณภาพผ่านการสัมภาษณ์ตัวต่อตัว	17
3.2	ความสามารถของระบบ	18
3.2.1	Use Case Diagram	18
3.2.2	Use Case Narrative	18
3.2.2.1	Resume Career Prediction	18
3.2.2.2	See career path node tree	19
3.2.2.3	View Community	19
3.2.2.4	View Topic/Forum	19
3.2.2.5	Create Topic/Forum	19
3.2.2.6	Reply Topic/Forum	20
3.2.2.7	Upload Resume	20
3.2.2.8	Analyzed Resume	20
3.3	สถาปัตยกรรมของระบบ	21
3.3.1	ส่วนผู้ใช้ (Client-Side)	21
3.3.2	ส่วนเซิร์ฟเวอร์ (Server-Side)	21
3.4	Sequence Diagram	22
3.4.1	Register	22
3.4.2	Login	23
3.4.3	Resume Prediction	23
3.4.4	Analyze Resume	24
3.4.5	View Forum	25
3.4.6	Reply Forum	25
3.4.7	Create Forum	26
3.4.8	See Career Node Tree	26
3.5	Site Map	27
3.6	Navigation Map	28

3.7	Wireframe	29
3.7.1	Resume Checker	29
3.7.2	Match Career	31
3.7.3	Evaluate Result	31
3.7.4	Career Node	32
3.7.5	Community	33
3.8	ขั้นตอนการพัฒนาโมเดล	35
3.8.1	การรวบรวมข้อมูล (Get Data)	35
3.8.2	การเตรียมข้อมูล (Clean, Prepare and Manipulate Data)	35
3.8.3	การเทรนโมเดล (Train Model)	36
3.8.4	การทดสอบข้อมูล (Test Data)	37
3.8.5	การพัฒนาโมเดล (Improve)	37
บทที่ 4	ผลการดำเนินงาน	38
บทที่ 5	บทสรุป	39

สารบัญตาราง

ตารางที่

หน้า

2.1 ตารางเปรียบเทียบคุณสมบัติที่สนใจ

16

สารบัญรูป

รูปที่	หน้า
1.1 ตารางการดำเนินงาน	3
2.1 ขั้นตอนกระบวนการคิดเชิงออกแบบ	4
2.2 กราฟแสดงความสัมพันธ์ของสมการ	5
2.3 สมการการคำนวณ Term-Frequency (TF)	9
2.4 สมการการคำนวณ Inverse Document Frequency (IDF)	10
2.5 ลักษณะการทำงานของ K-Nearest Neighbors	10
2.6 สมการความน่าจะเป็นของ Bayes หรือ Bayesian	11
3.1 ข้อมูลจากแบบสำรวจเชิงปริมาณ	17
3.2 Use Case Diagram	18
3.3 System Architecture	21
3.4 Register Sequence Diagram	22
3.5 Login Sequence Diagram	23
3.6 Resume Prediction Sequence Diagram	23
3.7 Analyze Resume Sequence Diagram	24
3.8 View Forum Sequence Diagram	25
3.9 Reply Forum Sequence Diagram	25
3.10 Create Forum Sequence Diagram	26
3.11 See Career Node Tree Sequence Diagram	26
3.12 SiteMap	27
3.13 Navigation Map	28
3.14 หน้าหลักสำหรับกดเริ่มวิเคราะห์ Resume	29
3.15 หน้ากรอกข้อมูลของเพื่อนำไปวิเคราะห์	29
3.16 ช่องสำหรับกรอกข้อมูลของเพื่อนำไปวิเคราะห์	30
3.17 หน้าแสดงอาชีพที่วิเคราะห์ได้	31
3.18 หน้าแสดงข้อมูลเชิงลึกของอาชีพ	32
3.19 หน้าแสดงทักษะที่ใช้ในสายอาชีพ	33
3.20 หน้าเลือกหมวดหมู่ของกระทุ้	33
3.21 หน้าแสดงรายการกระทุ้	34
3.22 หน้าแสดงรายละเอียดของกระทุ้	35
3.23 จำนวนของเรซูเม่ในแต่ละหมวดอาชีพ	36
3.24 ลักษณะข้อมูลของชุดข้อมูล	36

สารบัญสัญลักษณ์

SYMBOL

α	Test variable
λ	Interarival rate
μ	Service rate

UNIT

m^2
jobs/ second
jobs/ second

สารบัญคำศัพท์ทางเทคนิคและคำย่อ

Test	=	Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam non condimentum purus. Pellentesque sed augue sapien. In volutpat quis diam laoreet suscipit. Curabitur fringilla sem nisi, at condimentum lectus consequat vitae.
MANET	=	Mobile Ad Hoc Network

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญ

ในช่วงเวลาปัจจุบันนี้ เป็นช่วงที่เกิดปัญหานักศึกษาจบใหม่ว่างงานเพิ่มขึ้นทุกปี [2] และมีแนวโน้มจะเพิ่มขึ้นอีกเรื่อย ๆ ซึ่งอาจส่งผลให้เกิดความเครียดในหมู่นักศึกษาจบใหม่ กระบวนการวางแผนชีวิตในอนาคต อาจต้องมีการย้ายสายงาน ทำงานไม่ตรงสายการเรียน เป็นต้น โดยปัญหานี้ เกิดมาจากปัจจัยหลายอย่างทั้งในแง่ระบบการปกครอง ความต้องการของสายอาชีพต่าง ๆ ในตลาดแรงงานที่เปลี่ยนแปลงไป ระบบการศึกษา หรืออื่น ๆ อีกมากมายเกินที่เราจะควบคุมได้ อย่างไรก็ตาม หากเป็นการส่งเสริมด้านการศึกษาเพิ่มเติมด้วยตัวเองและชี้แนะแนวทางนั้น สามารถเป็นไปได้ ทางคณะผู้จัดทำจึงได้เริ่มมองหาจุดที่สามารถเข้าช่วยเหลือและบรรเทาปัญหานี้ โดยเริ่มตั้งเป้าหมายไว้ที่นักศึกษามหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี คณะวิศวกรรมศาสตร์ สาขาวิศวกรรมคอมพิวเตอร์เป็นกลุ่มแรกก่อน เพื่อนำมาพิสูจน์ผลลัพธ์ของวิธีแก้ปัญหามาที่เราออกแบบ

จากการสำรวจกับกลุ่มเป้าหมาย ทางคณะผู้จัดทำได้สังเกตเห็นถึงปัญหาร่วมบางอย่าง ซึ่งทางคณะผู้จัดทำเองก็ได้ประสบพบเจอด้วยตัวเองตั้งแต่ช่วงชั้นปีที่ 1 และยังคงพบเจออยู่จนถึงปัจจุบันเช่นกัน นั่นคือปัญหาในการค้นหาและรับรู้สายอาชีพที่เหมาะสมกับตนเอง แนวทางการเรียนรู้และพัฒนาความสามารถ ไม่ทราบศาสตร์ความรู้ที่จำเป็นต่อการไปสู่สายงานนั้น ๆ เช่น วิชาเรียนที่ควรเลือกในช่วงมหาวิทยาลัย ค่าจ้างงานแข่งพัฒนาทักษะต่าง ๆ ที่ควรเข้าร่วมเพื่อเสริมประสบการณ์ แม้กระทั่งรายละเอียดหรือหน้าที่ความรับผิดชอบของสายอาชีพต่าง ๆ ก็ยังไม่เป็นที่เข้าใจอย่างถูกต้องในหมู่นักศึกษาหลายคน โดยพบว่าผู้มีที่ประสบปัญหานี้ยังคงมีอยู่ทุกชั้นปี

จากสิ่งที่กล่าวไป ส่งผลให้นักศึกษาหลายคนไม่สามารถตอบได้ว่าตนเองควรพัฒนาตนเองอย่างไร สายอาชีพใดคือสิ่งที่ใช่สำหรับตนเอง แม้จะรู้ว่าต้องการไปในสายงานใดก็ไม่อาจทราบได้ว่าต้องพัฒนาตนเองอย่างไรต่อไปจึงจะตอบโจทย์ตลาดแรงงาน หรือรู้ตัวช้าเกินไปจนพัฒนาได้ไม่ทันการ ซึ่งสะท้อนปัญหาที่ทางคณะผู้จัดทำก็ได้พบเจอด้วย

หากสามารถแก้ไขปัญหานี้ได้ จะช่วยพัฒนาศักยภาพของนักศึกษาได้อย่างรวดเร็วและทำให้พวกเขาไปถึงจุดที่ตนเองพอใจได้ง่ายขึ้น อีกทั้งยังเป็นโอกาสที่ดีในการสร้างชื่อให้แก่มหาวิทยาลัย และทำให้นักศึกษาที่สำเร็จการศึกษาสามารถใช้ช่องทางนี้ในการกลับมาเสาะหาบุคลากรไปช่วยในตลาดแรงงานได้ในอนาคต ทำให้มีชุมชนที่แข็งแกร่งมากขึ้นตามกาลเวลา และยังช่วยให้เห็นภาพการพัฒนาตนเองที่ดียิ่งขึ้นด้วย

ทางคณะผู้จัดทำจึงมีความสนใจที่จะพัฒนาแพลตฟอร์มสำหรับใช้ช่วยเหลือในการค้นหาสายอาชีพที่เหมาะสมกับนักศึกษากลุ่มเป้าหมายด้วยปัญญาประดิษฐ์ พร้อมมีระบบกึ่งชุมชนที่สามารถใช้ในการให้พูดคุยเรื่องต่าง ๆ ใช้ประกาศข่าว ค้นหาเพื่อนร่วมอุดมการณ์ หรือรุ่นพี่และบุคคลในสายงานมาให้คำแนะนำนักศึกษา อีกทั้งยังสามารถให้บุคคลภายนอกใช้ในการรับสมัครบุคลากรผ่านเว็บไซต์ได้ ทั้งในรูปแบบงานเต็มเวลาหรือการฝึกงาน ซึ่งจะช่วยให้นักศึกษาเห็นความต้องการของตลาดแรงงานตามจริง เป็นโอกาสในการหาแนวทางพัฒนาตนเองล่วงหน้าหรือการมุ่งเข้าสู่สายงานที่สนใจได้

ดังนั้น หากคณะผู้จัดทำสามารถใช้โครงการนี้ในการช่วยเหลือนักศึกษาคนอื่น ๆ ในปัจจุบันหรืออนาคต รวมไปถึงคณะผู้จัดทำเองได้ สิ่งนี้จะเป็นประโยชน์แก่นักศึกษาอย่างยิ่งใหญ่ในด้านการศึกษาและการทำงาน เสมือนเป็นตัวเบิกทางให้นักศึกษามหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี คณะวิศวกรรมศาสตร์ สาขาวิศวกรรมคอมพิวเตอร์ ในอนาคตหลังจากนี้ ซึ่งเป็นแรงจูงใจที่สำคัญยิ่งกับทางคณะผู้จัดทำที่พบปัญหาด้วยตนเอง และมีอุดมการณ์อยากช่วยเหลือในด้านนี้เช่นกัน

1.2 วัตถุประสงค์

ระบุสิ่งที่ทำในโครงการ ซึ่งจะใช้สำหรับการประเมินว่าโครงการทำสำเร็จหรือไม่

- เพื่อศึกษาและจับจุดสำคัญของข้อมูลภายในเรซูเม่ของนักศึกษากลุ่มเป้าหมาย
- เพื่อพัฒนาโมเดลปัญญาประดิษฐ์สำหรับวิเคราะห์ข้อมูลเรซูเม่ของนักศึกษาออกมาเป็นอาชีพที่เหมาะสมกับตัวนักศึกษาได้
- เพื่อพัฒนาเว็บแอปพลิเคชันสำหรับนำมาใช้เป็นตัวกลางในการใช้งานปัญญาประดิษฐ์ เป็นชุมชนและตัวช่วยด้านการพัฒนาตนเองของกลุ่มเป้าหมายได้
- เพื่อลดปัญหาการค้นหาสิ่งที่เหมาะสมและพัฒนาตนเอง ช่วยเหลือการปรับข้อมูลเรซูเม่ และเป็นชุมชนแก่นักศึกษาในอนาคต

1.3 ขอบเขตของโครงการ

ขอบเขตด้านเว็บแอปพลิเคชัน

- เว็บแอปพลิเคชันจะมุ่งเน้นสนับสนุนไปที่เพียง 2 ขนาดหน้าจอ คือ เดสก์ท็อปและมือถือ

ขอบเขตด้านปัญญาประดิษฐ์

- ผู้ใช้จะต้องกรอกข้อมูลของเรซูเม่ด้วยตัวเองเพื่อให้ปัญญาประดิษฐ์วิเคราะห์ผลลัพธ์เป็นวิธีหลัก
- ปัญญาประดิษฐ์จะรองรับภาษาอังกฤษในการวิเคราะห์เท่านั้น

ขอบเขตด้านเนื้อหาและกลุ่มเป้าหมาย

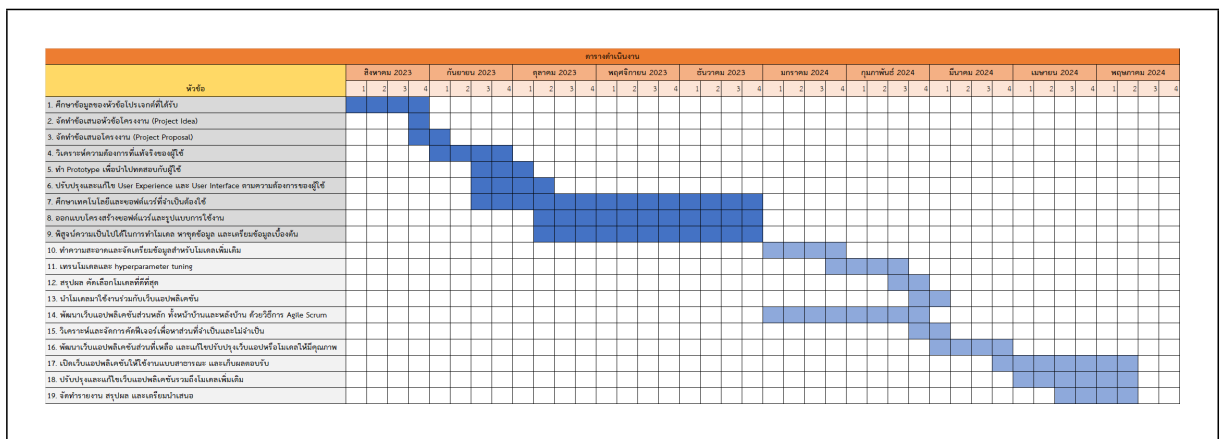
- มุ่งเน้นไปที่นักศึกษาวิศวกรรมคอมพิวเตอร์หลักสูตรปกติ และนานาชาติ
- สายอาชีพที่สนับสนุนจะประกอบด้วย 5 สายอาชีพดังนี้
 - Software Engineer เช่น frontend developer, backend developer และ full-stack developer
 - Designer เช่น UX/UI Designer
 - Data และ AI เช่น Data Engineer, Data Science และ Data Analyst
 - Security เช่น Data Security และ Cyber Security
 - Cloud Management เช่น DevOps

เนื่องจากข้อจำกัดจากข้อมูลที่มีในปัญญาประดิษฐ์ ความนิยมของสายอาชีพนั้น ๆ ในตลาดแรงงาน [?] และนัยยะสำคัญที่ได้มาจากกลุ่มเป้าหมาย

1.4 ประโยชน์ที่คาดว่าจะได้รับ

ทางคณะผู้จัดทำคาดหวังเป็นอย่างยิ่งว่าจะสามารถนำเว็บแอปพลิเคชันนี้มาใช้งานกับนักศึกษาวิศวกรรมคอมพิวเตอร์ทุกชั้นปีซึ่งรวมถึงคณะผู้จัดทำเองด้วย โดยหวังว่าจะช่วยเหลือให้กลุ่มเป้าหมายนี้ สามารถปรับปรุงเรซูเม่ของตนเองให้ดียิ่งขึ้น หรือปรับให้เข้ากับความต้องการของตนเอง เสริมความมั่นใจในการนำไปสมัครงานตามช่องทางที่ตนเองสนใจ รวมไปถึงใช้งานเว็บแอปพลิเคชันเพื่อพัฒนาตนเอง เสาะหาโอกาสเพิ่มเติมในสายอาชีพ รวมไปถึงกลับมาค้นหาคำที่สนใจร่วมงานด้วยในอนาคตหลังจบการศึกษา ซึ่งถือเป็นการสร้างชุมชนที่แข็งแกร่งขึ้นตามกาลเวลา อีกทั้งยังสามารถพัฒนาปัญญาประดิษฐ์ให้มีประสิทธิภาพมากขึ้นจากข้อมูลที่ได้รับระหว่างเปิดให้ใช้งานตามเวลาอีกด้วย ซึ่งเป็นประโยชน์อย่างมากในยุคที่ข้อมูลมีความสำคัญเช่นนี้

1.5 ตารางการดำเนินงาน



รูปที่ 1.1 ตารางการดำเนินงาน

บทที่ 2 ทฤษฎีความรู้และงานที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 Human Computer Interaction (HCI)

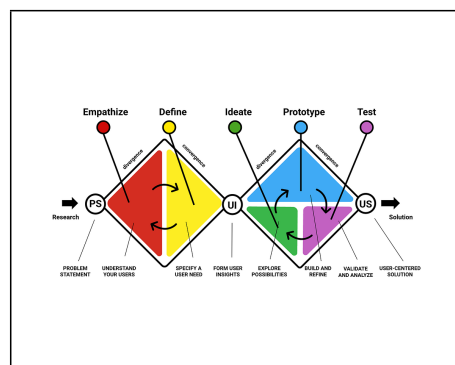
การปฏิสัมพันธ์ระหว่างมนุษย์และคอมพิวเตอร์ (HCI) หมายถึงการศึกษาและออกแบบ วิธีการในการสร้างปฏิสัมพันธ์ระหว่างมนุษย์และคอมพิวเตอร์ได้อย่างลื่นไหลและ ได้รับประสบการณ์ที่ดี แม้ว่าคอมพิวเตอร์ในปัจจุบันจะเป็นเครื่องมือที่มีความสามารถสูง แต่เพื่อให้ผู้ใช้สามารถใช้งานและเข้าถึงข้อมูลได้อย่างมีประสิทธิภาพ ตัวหลักการของ HCI จะทำหน้าที่ออกแบบการปฏิสัมพันธ์โดยมองจากมุมมองของมนุษย์และผู้ใช้ที่เป็นกลุ่มเป้าหมาย และทำให้ดีขึ้นกว่าเดิม

การศึกษาของ HCI เน้นไปที่การเข้าใจความต้องการของผู้ใช้ การออกแบบอินเทอร์เฟซที่ใช้งานง่ายและมีประสิทธิภาพ การทดสอบและปรับปรุงการใช้งาน การทำให้ผู้ใช้มีประสิทธิภาพในการติดต่อสื่อสารกับเทคโนโลยี และการวิจัยเกี่ยวกับปัญหาที่เกี่ยวข้องกับการปฏิสัมพันธ์ระหว่างมนุษย์และคอมพิวเตอร์

การนำไปใช้งาน : ทางคณะผู้จัดทำได้นำ HCI ที่เป็นศาสตร์ที่สำคัญ ในการพัฒนาเว็บแอปพลิเคชันให้ใช้งานง่ายและมีประสิทธิภาพมากขึ้น การประยุกต์ใช้ HCI กับโปรเจกต์เว็บแอปพลิเคชันจะช่วยให้ผู้ใช้สามารถใช้งานแอปพลิเคชัน ได้อย่างสะดวกและมีประสิทธิภาพมากขึ้น

2.1.2 Design Thinking

กระบวนการคิดเชิงออกแบบ (Design Thinking) เป็นเครื่องมือที่ใช้ในการคิดเพื่อแก้ไขปัญหาอย่างเป็นระบบและมีประสิทธิภาพ และจะเน้นไปที่การแก้ไขปัญหาของกลุ่มเป้าหมายเป็นหลัก (User-Centered) โดยกระบวนการคิดเชิงออกแบบจะแบ่งเป็นทั้งหมด 5 ช่วง ได้แก่



รูปที่ 2.1 ขั้นตอนกระบวนการคิดเชิงออกแบบ

1. **Empathize** เป็นการทำความเข้าใจปัญหาที่เกิดขึ้นของกลุ่มเป้าหมายจาก การตั้งสมมุติฐาน การสัมภาษณ์กลุ่มเป้าหมาย หรือการสังเกตพฤติกรรม เป็นต้น
2. **Define** นำข้อมูลจากการสัมภาษณ์กลุ่มเป้าหมายมาวิเคราะห์เพื่อ กำหนดความต้องการของกลุ่มเป้าหมายให้ชัดเจน
3. **Ideate** การระดมสมองเพื่อหาวิธีการแก้ไขปัญหาจากจุดเจ็บปวด (Pain Point) ของกลุ่มเป้าหมายอย่างมีประสิทธิภาพ ตรงประเด็น และให้สามารถใช้งานง่ายที่สุด
4. **Prototype** เป็นการสร้างแบบจำลองของวิธีการแก้ไขปัญหาเพื่อ ที่จะสามารถนำไปทดสอบกับกลุ่มเป้าหมายได้
5. **Test** นำแบบจำลองที่สร้างขึ้นมาทดสอบกับกลุ่มเป้าหมาย เพื่อนำมาปรับปรุงแก้ไขเพื่อให้คำตอบได้อย่างมีประสิทธิภาพมากที่สุด

การนำไปใช้งาน : ทางคณะผู้จัดทำได้นำกระบวนการคิดเชิงออกแบบ (Design Thinking) มาประยุกต์ใช้กับโปรเจกต์ ทำให้ได้เว็บแอปพลิเคชัน ที่ตอบโจทย์ และสามารถใช้งานได้จริง โดยเน้นการเข้าใจผู้ใช้เป็นศูนย์กลาง (User Centric) ผ่านการทดลองและทดสอบกับผู้ใช้อย่างต่อเนื่อง

2.1.3 MVC Structure

เฟรมเวิร์กสำหรับการออกแบบสถาปัตยกรรมระบบต่าง ๆ ด้วยหลักการ Design pattern ที่จะแยกส่วนระบบการทำงานเป็น 3 ส่วน โดยมีหน้าที่ชัดเจนในแต่ละส่วน ดังนี้

1. **Model** เป็นส่วนที่มีหน้าที่ในการคำนวณตรรกะเชิงข้อมูล เช่น รับส่งข้อมูลกับฐานข้อมูลของระบบ รวมไปถึงจัดการกับข้อมูลก่อนส่งไปสู่ระบบอื่น ๆ โดยส่วน Model นี้ จะตอบรับคำขอจากส่วน Controller เป็นหลัก
2. **View** เป็นส่วนสำหรับคำนวณตรรกะในส่วนปฏิสัมพันธ์ผู้ใช้ ซึ่งจะส่งผลต่อสิ่งที่นำไปแสดงผลให้ผู้ใช้งานเห็นทางสายตาโดยตรง หากต้องการใช้ข้อมูลจากฐานข้อมูล ส่วน View นี้จะติดต่อขอข้อมูลจากส่วน Controller ก่อน
3. **Controller** ส่วนที่ใช้สำหรับติดต่อกับระบบทั้งหมดในโครงสร้าง มีหน้าที่ควบคุมการไหลของเส้นคำขอในซอฟต์แวร์ และคำนวณตรรกะเชิงธุรกิจเท่านั้น หากต้องการใช้ข้อมูลจากฐานข้อมูล ก็จะใช้งานระบบส่วน Model และหากต้องการคำนวณส่วนที่จะนำไปปฏิสัมพันธ์กับผู้ใช้ ก็จะติดต่อใช้งานระบบส่วน View

สถาปัตยกรรมแบบ MVC นั้นมีข้อดีที่สามารถทำให้โค้ดภายในระบบมีความง่าย ในการดูแลและปรับปรุง สามารถนำไปทดสอบความใช้งานได้ได้ง่ายเพราะหน้าที่ภายใน แต่ละส่วนชัดเจน และทำงานร่วมกับทีมได้สะดวก

อย่างไรก็ตาม สถาปัตยกรรมแบบ MVC ก็สามารถก่อข้อเสียบางอย่างได้ เช่น ยากที่ทำความเข้าใจหรืออ่านโค้ดทั้งหมดได้หากมีขนาดใหญ่และมีการแบ่งหน้าที่ ในการทำงานชัดเจนระหว่างนักพัฒนา ซึ่งทำให้สถาปัตยกรรมนี้ไม่เหมาะกับการใช้งาน กับระบบที่มีขนาดเล็ก เพราะจะเพิ่มความซับซ้อนโดยใช้เหตุ

การนำไปใช้งาน : ทางคณะผู้จัดทำ พึ่งเล็งในการนำรูปแบบการออกแบบสถาปัตยกรรม MVC ไปใช้กับการออกแบบระบบการทำงานหลังบ้าน ซึ่งมีความเหมาะสมกับเครื่องมืออย่าง NestJS ซึ่งสนับสนุนวิธีการออกแบบสถาปัตยกรรม MVC และเหมาะสมกับประเภทงานอย่างเว็บแอปพลิเคชันตามที่กล่าวไปในเนื้อหา

2.1.4 The 5 Users design process

หลักการที่ให้เหตุผลว่า ทำไมเราควรทดสอบผลิตภัณฑ์กับผู้ใช้ในกลุ่มเป้าหมายด้วย จำนวนราว 5 คน โดยทฤษฎีกล่าวว่า การทดสอบที่มากและกว้างเกินไปนั้นไม่ส่งผลดี แม้น้อย ทั้งยังสิ้นเปลืองทรัพยากร โดยผลลัพธ์ที่ดีที่สุดจะเกิดจากการทดสอบ ไม่เกิน 5 คน และทดสอบด้วยสิ่งเล็ก ๆ ไม่กว้างเกินไปเท่าที่จะเป็นไปได้

งานวิจัยนี้มาจากคุณ Tom Landauer ซึ่งได้แสดงให้เห็นว่า จำนวนปัญหาการใช้งาน หรือข้อผิดพลาดต่าง ๆ จะสามารถเขียนเป็นสมการได้ดังนี้

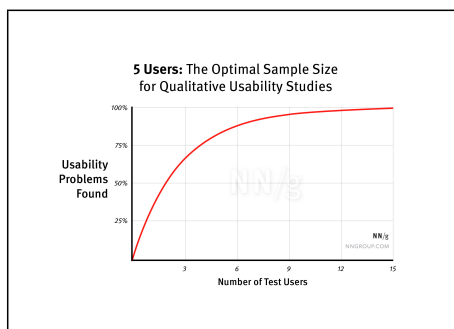
$$N(1 - (1 - L)^n)$$

n คือ จำนวนผู้ทำการทดสอบ

N คือ จำนวนของปัญหาหรือข้อผิดพลาดที่พบทั้งหมดจากการทดสอบ

L คือ สัดส่วนร้อยละของปัญหาหรือข้อผิดพลาดที่พบในระหว่างการทดสอบผู้ใช้รายบุคคล

โดยปกติทั่วไป ค่าเฉลี่ยของ L จะอยู่ 31% ดังนั้น เมื่อทำการแสดงแผนภาพความสัมพันธ์ จะได้กราฟดังนี้



รูปที่ 2.2 กราฟแสดงความสัมพันธ์ของสมการ

เมื่อมองจากอัตราที่ได้รับข้อมูล พบว่า เส้นความชันจะเริ่มต่ำลงตั้งแต่จำนวน 3-5 คน และเริ่มหยุดนิ่ง จึงสามารถสรุปได้ว่าจำนวนผู้ใช้ในการสัมภาษณ์ที่ดี ในการค้นหาข้อผิดพลาดหรือปัญหา ควรอยู่ที่ราว ๆ 5 คน

อย่างไรก็ตาม ทฤษฎีนี้สามารถนำไปปรับใช้เพิ่มเติมได้ เช่น หากผู้ใช้มีกลุ่มเฉพาะ ที่ชัดเจนหลายกลุ่ม เราสามารถเปลี่ยนจำนวนการทดสอบเป็น 3-4 คนต่อกลุ่ม เนื่องจากข้อมูลบางอย่างอาจมีการทับซ้อนกันระหว่างกลุ่มได้ด้วย

การนำไปใช้งาน : ทางคณะผู้จัดทำ ได้นำหลักการนี้ไปปรับใช้กับ การเลือกกลุ่มตัวอย่างและจำนวนในการสัมภาษณ์เพื่อเก็บข้อมูลเชิงคุณภาพ หรือการทดสอบวิธีการแก้ปัญหาที่เราออกแบบกับผู้ใช้ โดยคาดหวังว่า จะได้รับข้อมูลที่เห็นแนวโน้มในภาพรวมจากจำนวนในทฤษฎี ประหยัดเวลาที่ใช้ในการศึกษากลุ่มเป้าหมายและเน้นไปที่ประสิทธิภาพมากขึ้น

2.1.5 Usability Heuristics Principles

การออกแบบส่วนต่อประสานกับผู้ใช้ (User Interface Design) เป็นกระบวนการที่ให้ความสำคัญกับประสิทธิภาพในการใช้งาน และประสบการณ์ของผู้ใช้ ซึ่งหนึ่งในเครื่องมือที่มีประสิทธิภาพในการประเมินการออกแบบนั้นคือ "Usability Heuristics Principles" ซึ่งถูกพัฒนาโดย Jakob Nielsen ในปี 1994 โดยมีเป้าหมายเพื่อช่วยนักออกแบบในการพัฒนาและปรับปรุงการออกแบบ ส่วนต่อประสานกับผู้ใช้ให้ดีขึ้น โดย Usability Heuristics มีหลักการทั้ง 10 ข้อ ดังนี้

1. **Visibility of System Status** ผู้ใช้ควรทราบสถานะปัจจุบันของระบบตลอดเวลา โดยการใช้สถานะแสดงอย่างชัดเจน เช่น แถบความคืบหน้า หรือสถานะการทำงาน
2. **Match between System and the Real World** การออกแบบควรสอดคล้องกับความคาดหวังและความรู้ของผู้ใช้ในโลกแห่งความเป็นจริง เพื่อลดความสับสนและเพิ่มความเรียบง่ายในการใช้งาน
3. **User Control and Freedom** ผู้ใช้ควรมีอิสระในการยกเลิกหรือย้อนกลับการกระทำที่เกิดขึ้นโดยไม่ก่อให้เกิด ความรู้สึกกังวล หรือติดอยู่กับขั้นตอนนั้น ๆ และต้องมีทางออกฉุกเฉินในกรณีที่ผู้ใช้ ไม่ต้องการทำการกระทำเหล่านั้น
4. **Consistency and Standards** การออกแบบควรปฏิบัติตามมาตรฐานที่ได้กำหนดไว้ ตั้งแต่การออกแบบกราฟิกไป จนถึงรูปแบบการใช้งาน เพื่อให้ผู้ใช้งานสามารถจดจำและใช้งานได้ง่ายด้วยการสร้าง Design System เพื่อวางแนวทางในการออกแบบ
5. **Error Prevention** การออกแบบควรป้องกันข้อผิดพลาดจากการเกิดขึ้น ด้วยการมีการเตือนให้ผู้ใช้ ตรวจสอบความถูกต้องก่อนที่จะทำอะไรบางอย่างที่สำคัญหรือร้ายแรง
6. **Recognition rather than Recall** ผู้ใช้ควรสามารถรู้จักและใช้งานส่วนต่อประสานได้อย่างเหมาะสมโดยไม่ต้องใช้ความรู้ หรือความจำที่มากเกินไป
7. **Flexibility and Efficiency of Use** การออกแบบควรสนับสนุนการใช้งานของผู้ใช้ที่มีความสามารถในการใช้งานมากและ น้อย อย่างมีประสิทธิภาพ
8. **Aesthetic and Minimalist Design** การออกแบบควรใช้ข้อความและสัญลักษณ์ให้มีความชัดเจน รวมถึงสวยงามและเรียบง่าย โดยไม่ทำให้ผู้ใช้งานเกิดความสับสนและไม่มากจนเกินไป
9. **Help Users Recognize, Diagnose, and Recover from Errors** การออกแบบควรช่วยให้ผู้ใช้รู้ว่าเกิดข้อผิดพลาดอะไรขึ้น และมีแนวทางสำหรับ การแก้ไขปัญหาให้ผู้ใช้ตามหาได้
10. **Help and Documentation** การออกแบบควรให้ความช่วยเหลือและเอกสารความช่วยเหลือให้ผู้ใช้เมื่อจำเป็น แต่ก็ควรสร้างระบบที่ใช้งานได้โดยไม่ต้องพึ่งพาเอกสารอ้างอิงต่าง ๆ

การใช้งาน 10 Usability Heuristics for User Interface Design เป็นเครื่องมือที่มีประสิทธิภาพในการปรับปรุงประสิทธิภาพการใช้งานและประสบการณ์ของ ผู้ใช้ในเว็บไซต์และแอปพลิเคชัน โดยการปฏิบัติตามหลักการเหล่านี้ จะทำให้สามารถเพิ่มคุณภาพและประสิทธิภาพของส่วนต่อประสานกับผู้ใช้ให้ดียิ่งขึ้น

การนำไปใช้งาน : ทางคณะผู้จัดทำได้นำ Usability Heuristics Principles ไปใช้เพื่อเป็นเกณฑ์ในการวัดความใช้งานง่าย ของเว็บไซต์ แอปพลิเคชันและรับข้อเสนอแนะของผู้ใช้เพื่อนำมาปรับปรุงให้ประสบการณ์ ของผู้ใช้ในการใช้งานเว็บไซต์แอปพลิเคชันดียิ่งขึ้น

2.1.6 Artificial Neural Network (ANN)

เป็นแนวคิดเพื่อนำมาใช้ในการสร้างโมเดล machine learning โดยใช้วิธีการสร้างโครงข่ายประสาทเทียม และมีการสร้างเครื่องที่ชื่อว่า Perceptron เพื่อนำมาใช้ทดลองตั้งแต่ปี 1958

อัลกอริทึมที่ใช้ในการฝึกฝนโมเดล ANN จะประกอบด้วยพื้นฐาน ดังนี้

1. สุ่มค่าเริ่มต้นของ w และ b
2. เลือกค่า learning rate r ระหว่าง 0 และ 1
3. สำหรับจุดข้อมูล (x,y) คำนวณค่า $f(x) = w \cdot x + b$
4. ปรับค่า w และ b โดยใช้สมการ $w = w + r(y - f(x))x$ และ $b = b + r(y - f(x))$
5. ทำซ้ำตามจำนวนครั้งที่ต้องการหรือจนกว่าอัตราความผิดพลาดจะน้อยกว่าที่กำหนด

โดยที่ w เป็นเวกเตอร์น้ำหนัก และ b เป็นค่าไบแอสที่โมเดลจะเรียนรู้ขึ้นมาระหว่างการฝึกฝน

อย่างไรก็ตาม รูปแบบนี้ยังมีข้อจำกัด เช่น ไม่สามารถทำนายความสัมพันธ์แบบ XOR ได้ ดังนั้นจึงมีการเพิ่มน้ำหนักของสมการตัดสินใจนี้ ดังนี้

$$y = x_1 \oplus x_2 = f(x) = \begin{cases} 1, & \text{if } w \cdot x + b = w_1x_1 + w_2x_2 + w_3x_1x_2 + b > 0 \\ 0, & \text{otherwise} \end{cases}$$

และต่อมายังเพิ่มขีดความสามารถให้โมเดลมีความยืดหยุ่นมากขึ้น โดยการพัฒนาขั้นต่อนั้น เป็นการเพิ่ม layer ของ Perceptron เข้าไป โดยมีฟังก์ชันที่ไม่เป็นเชิงเส้นคั่นระหว่าง layers ดังนั้น ฟังก์ชันของแต่ละ layer คือ

$$f(x) = \sigma(w \cdot x + b)$$

โครงสร้างพื้นฐานของโมเดล Multi-layer Perceptron ประกอบไปด้วย 3 ส่วน ได้แก่ input layer, hidden layer และ output layer โดยสามารถมี hidden layer ได้หลาย layer ตามความต้องการของผู้ออกแบบโมเดล

จากสิ่งที่กล่าวมาทั้งหมดนี้ ANN ได้กลายเป็นพื้นฐานของโมเดลประเภทโครงข่ายประสาทเทียม และถูกนำไปปรับปรุงให้กลายเป็นโมเดลรูปแบบใหม่ ๆ เพื่อสร้างจุดเด่นกับงานเฉพาะด้าน เช่น การนำ output ของ layer ท้าย ๆ บ้อนกลับเข้าไปใน layer ก่อนหน้าพร้อมกับข้อมูลใหม่ (RNN) หรือ เชื่อม layer ต่าง ๆ โดยเชื่อมเฉพาะ neuron ที่อยู่ใกล้กันเท่านั้น ไม่ได้เชื่อมหมดทั้ง layer (CNN) **การนำไปใช้งาน** : ทางคณะผู้จัดทำได้ศึกษาพื้นฐานของระบบโครงข่ายประสาทเทียมเพื่อนำมาเป็นส่วนหนึ่งของการพัฒนาโมเดลปัญญาประดิษฐ์ภายในเว็บแอปพลิเคชันของเราทุกอัน รวมถึงถูกนำมาใช้งานในการฝึกฝนโมเดลเพื่อทดสอบประสิทธิภาพและเปรียบเทียบกับโมเดลอื่น ๆ อีกด้วย

2.1.7 Recurrent Neural Network (RNN)

RNN เป็นระบบโครงข่ายประสาท (neural network) ที่จะนำ output จาก state ก่อนหน้ามาเป็น input ในรอบใหม่ ทำให้สามารถเข้าใจข้อมูลประเภทที่มีลำดับได้ดี จำพวก ข้อความ หรือข้อมูลที่เป็นลำดับเวลา (time-series) โดยมีสมการดังนี้

$$h(t) = x(t)W_{in} + h(t-1)W$$

โดยสมการนี้แสดงถึงการที่ใช้ค่า Output ของ $x(t)$ ร่วมกับ Output ของ $h(t-1)$ (output ของ network ที่แล้ว) โดยมี Weight 2 ตัวปรับของ $x(t)$ กับ $h(t-1)$

อย่างไรก็ตาม RNN จะพบข้อปัญหาบางอย่างได้ง่าย เช่น การใช้งานหน่วยความจำจำนวนมากจากการวนรอบ หรือการแบ่งน้ำหนัก (weight) ที่อาจยังทำได้ไม่ดีนักหากใช้วิธีปกติ ดังนั้น จึงมีการนำ RNN ไปพัฒนาต่อจนเกิดเป็นรูปแบบใหม่ (variant) ที่แก้ปัญหานั้นได้ เช่น LSTM และ GRU

การนำไปใช้งาน : ทางคณะผู้จัดทำได้ศึกษาหลักการพื้นฐานของ RNN เพื่อนำไปเป็นพื้นฐานของการศึกษาเนื้อหาอื่น ๆ ต่อไป เช่น LSTM และ GRU ซึ่งอาจนำไปใช้เปรียบเทียบประสิทธิภาพกับหลักการอื่นอีกทีหนึ่ง

2.1.8 Long Short Term Memory (LSTM)

เป็น RNN ที่นำมาพัฒนาต่อ ทำงานได้ดีขึ้นกับข้อมูลที่ต้องมีการเรียนรู้แบบระยะยาว เพราะมีการกำหนดน้ำหนักสำหรับการลืมเอาไว้ให้สำหรับ model ด้วย ซึ่งจะช่วยลดปริมาณหน่วยความจำที่ใช้ได้ ซึ่งเป็นปัญหาที่พบได้จาก RNN ปกติ โดยภายใน LSTM จะมีตัวแปรที่สำคัญ ดังนี้

- **Cell state** เป็นตัวเก็บ state ของ memory cell ใน LSTM
- **Gate** เป็นตัวที่ควบคุมการไหลของข้อมูล ซึ่งเป็นรูปแบบ 0/1 (อนาล็อก) ที่จะควบคุมว่าเมื่อไหร่จะทำการ อ่าน เขียน หรือ ลืม (forget) ซึ่งเหมือนกับประตูที่จะออกคำสั่งว่า เมื่อไหร่ควรเปิดให้ข้อมูลไหลเข้า ไหลออก หรือควรลบทิ้ง

โดยจากที่กล่าวไป คำสั่งที่สำคัญภายใน gate ของ LSTM จะประกอบด้วย

1. **การลืม (forget)** เป็นการลืมข้อมูลเดิมเพื่อไปรับข้อมูลใหม่ โดยจะตัดสินใจลืมนี้ด้วย forget gate ซึ่งจะส่งสัญญาณ 0/1 ออกมา การสร้าง forget gate นี้ เราจะดู input data ที่เข้ามา ประกอบกับ hidden state ก่อนหน้า (ตามหลักการของ RNN) ประกอบการตัดสินใจ โดยจะใช้ sigmoid function เป็นตัวตัดสินใจ ดังสมการ

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f)$$

2. **การเขียน (write)** ก่อนจะมีการเขียนข้อมูลใหม่ จำเป็นต้องมีการตัดสินใจก่อนว่า จำเป็นต้องมีการเปลี่ยนแปลงค่าหรือไม่ หากต้องมีการเปลี่ยนแปลง จะเปลี่ยนแปลงเป็นอะไร โดยการตัดสินใจขั้นแรก จะใช้งานผ่าน input gate และใช้ sigmoid function เป็นตัวตัดสินใจเช่นเดิม (สามารถใช้สมการเดิมได้เหมือนกับสมการการลืม)

ซึ่งในกรณีต้องการเปลี่ยนแปลง จะใช้ Input modulation gate เป็นตัวจัดการ โดยสมการก็จะเหมือนกับ input gate แต่ว่าจะใช้เป็น tanh function ซึ่งค่าที่ได้นั้น เปรียบได้เป็น cell state candidate

$$g_t = \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c)$$

ดังนั้น วิธีการคำนวณค่าใหม่สำหรับการเปลี่ยนแปลง จะเป็นไปดังสมการ

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t$$

หาก gate ใด ๆ ส่งสัญญาณเป็น 0 ก็จะส่งผลให้ไม่นำค่าต่าง ๆ มาใช้ เช่น หากค่าจาก f (forget gate) เป็น 0 ค่าของ c ก็จะถูกลืมและไม่นำมาพิจารณา แต่หากเป็น 1 แสดงว่ามีความต้องการจะเปลี่ยนแปลงข้อมูล ซึ่งจะตัดสินใจว่าควรใช้ค่าใหม่ เป็นค่าใดจาก input modulation gate หากเป็น 1 ก็จะนำ g ไปใช้งานนั่นเอง

3. **การอ่าน (read)** เนื่องจากค่า output ที่ได้รับมาในขั้นตอน ก่อนหน้า เป็น output ณ เวลาต่าง ๆ ใน hidden state นั้น เราจึงสามารถคำนวณค่า ณ เวลานั้นกลับมาได้ด้วยสูตรเดิมเหมือนในขั้นตอนก่อนหน้า เพียงแต่ในขั้นตอนการอ่านนั้น จะมีการใช้งาน output gate มาช่วยตัดสินใจว่า ควรอนุญาตให้อ่านข้อมูล ณ ตอนนั้นหรือไม่ โดยสูตรคำนวณจะเหมือนกับ forget gate และ input gate อย่างที่เคยกล่าวไป นั่นคือ ใช้ sigmoid function กับค่า hidden state ตัวก่อนหน้า กับ input data ที่เข้ามาในตอนนั้น

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o)$$

จากค่าของ open gate ในสูตรดังกล่าว จะเขียนสมการในการอนุญาตให้อ่าน ข้อมูลได้ดังนี้

$$h_t = o_t \odot \tanh(c_t)$$

การนำไปใช้งาน : เป็นหนึ่งในทางเลือกที่ทางคณะผู้จัดทำได้ศึกษา เพื่อนำความรู้ไปพิจารณาใช้งานกับวิธีการแก้ปัญหาที่ทางคณะผู้จัดทำออกแบบ ซึ่ง LSTM สามารถนำไปใช้งานและตอบโจทย์การทำ Text-based Generative AI ได้ อย่างไรก็ตาม คณะผู้จัดทำยังมีข้อจำกัดเรื่องของปริมาณข้อมูลที่สามารถนำมาใช้ และประสิทธิภาพที่เมื่อเทียบกับ model ประเภทอื่นในปัจจุบัน อาจด้อยกว่าเล็กน้อย จึงอาจเป็นตัวเลือกรอง แต่ก็ถือว่าเป็นหนึ่งในตัวเลือกที่น่าสนใจไม่น้อย

2.1.9 SQL และ NoSQL

sql หรือ Structured query Language เป็นภาษาโปรแกรมมิ่งชนิดหนึ่ง ที่ใช้ในการสื่อสารกับ ฐานข้อมูลชนิดที่มีความสัมพันธ์ หรือที่เรียกว่า RDBMS (Relational Database Management System) ซึ่งมีวิธีการเก็บข้อมูลในรูปแบบ ของตาราง (table) โดยภายใน ตารางจะประกอบด้วย

1. **Row (แถว หรือ แนวนอน)** เรียกอีกชื่อว่า Tuple คือ ตัวเนื้อข้อมูล
2. **Column (สดมภ์ หรือ แนวตั้ง)** เรียกอีกชื่อว่า Attribute คือ การระบุชนิดของข้อมูลนั้น ๆ เช่น ที่อยู่, วัน เดือน ปีเกิด, ตัวเลข โดยแต่ละตารางจะเชื่อมความสัมพันธ์กันด้วยข้อกำหนดที่เรียกว่า key ซึ่งมีสองรูปแบบคือ

1. **Primary Key** หมายถึง การกำหนดรูปแบบของ column ที่จะไม่เกิดการซ้ำกันได้ และไม่มีทางเกิดข้อมูลว่าง
2. **Foreign Key** หมายถึง ต้องมีการอ้างอิงถึงข้อมูลประเภทเดียวกัน จากตารางที่มี Primary Key อยู่ภายใน

NoSQL หรือ Non-relational database เป็นฐานข้อมูลที่มีความสัมพันธ์กัน ชัดเจนแบบ เหมาะสำหรับการใช้งานกับ Real-time Web Application โดยประเภทของ NoSQL จะแบ่งออกเป็น 4 แบบหลัก ๆ ได้แก่

1. **Document** ข้อมูลจะเก็บเป็นลำดับชั้นในรูปแบบ Semi-structure data เช่น JSON
2. **Key-value** ข้อมูลจะเก็บในรูปแบบแถว (record) ที่ประกอบด้วย key และ value ที่เชื่อมกันแบบหนึ่งต่อหนึ่ง เข้าถึงข้อมูลได้เร็ว
3. **Graph** ข้อมูลจะเก็บอยู่ในรูปแบบกราฟแผนภูมิ มี Node และ Edge ที่เชื่อมต่อกัน ทำให้ไม่ต้องนำข้อมูลมาเชื่อม (join) เหมือนกับ วิธีการของ RDBMS
4. **Wide-Column** บันทึกข้อมูลในรูปแบบ table (row/column) แต่จะต่างจาก RDBMS ตรงที่แต่ละ row จะไม่กำหนดหรือบังคับ ประเภท column

การนำไปใช้งาน : ทางคณะผู้จัดทำได้พิจารณาจากรูปแบบของข้อมูล ที่ต้องเก็บเป็นหลัก ความสะดวกของบริการที่สามารถนำมาใช้งาน และพิจารณา ในความเหมาะสมของการพัฒนาแล้ว โดยพ่วงถึงว่า จะมีการนำฐานข้อมูลในรูปแบบ NoSQL ประเภท document มาใช้งานเป็นหลัก โดยมีบริการที่น่าสนใจและเอื้อมถึงได้ง่าย เช่น MongoDB

2.2 อัลกอริทึมในการแปลผลภาษา

2.2.1 Term Frequency Inverse Document Frequency (TF-IDF)

เป็นอัลกอริทึมที่ผสมผสานกันระหว่าง Term-Frequency (TF) และ Inverse Document Frequency (IDF) ซึ่งเป็น เทคนิคพื้นฐานเทคนิคหนึ่งที่ใช้ในการวิเคราะห์ค้นหาคำสำคัญของข้อมูลในลักษณะของข้อความ [?]

- **Term-Frequency (TF)** โดยจะคำนวณเป็นอัตราส่วนของจำนวนคำนั้น ๆ ต่อจำนวนคำทั้งหมดในเอกสาร เพื่อหาคำนั้นมี ความถี่เท่าไร

$$TF(\text{ของคำหนึ่ง}) = \frac{\text{จำนวนของคำนั้นๆ ในเอกสาร}}{\text{จำนวนของคำทั้งหมดในเอกสาร}}$$

รูปที่ 2.3 สมการการคำนวณ Term-Frequency (TF)

- **Inverse Document Frequency (IDF)** โดยจะคำนวณความสำคัญของแต่ละคำโดยคำที่พบได้บ่อยจะมีค่า IDF ที่ต่ำ ซึ่งบ่งบอก ว่าคำเหล่านั้นไม่สามารถดึงเอาจุดเด่นของเอกสารออกมาได้ดี

$$IDF(\text{ของคำหนึ่ง}) = \log\left(\frac{\text{จำนวนเอกสารทั้งหมดที่ใช้พิจารณา}}{\text{จำนวนเอกสารที่มีคำนั้นปรากฏอยู่}}\right)$$

รูปที่ 2.4 สมการการคำนวณ Inverse Document Frequency (IDF)

- **คำนวณค่า TF-IDF** โดยเราจะนำ TF กับ IDF มาคำนวณและถ้าหากคำไหนที่มีค่า TF-IDF ที่สูง จะถูกมองว่าเป็นคำที่มีความสำคัญสูง (กล่าวถึงบ่อย แต่ก็ไม่ได้ปรากฏอยู่หลายเอกสารเกินไป) และมีแนวโน้มจะเป็นใจความสำคัญของเอกสาร

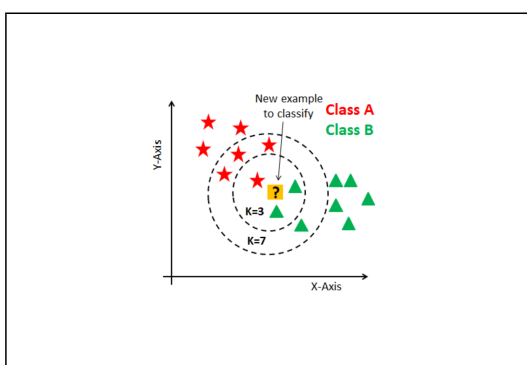
$$TFIDF = TF * IDF$$

2.3 อัลกอริทึมในการแยกประเภทเรซูเม่

2.3.1 อัลกอริทึม I K-Nearest Neighbors (KNN)

เป็นอัลกอริทึมสำหรับการจัดกลุ่มข้อมูล (Classification) ซึ่งอยู่ในกลุ่มของการเรียนรู้แบบมีผู้สอน (Supervised Learning) หลักการทำงาน คือการจัดกลุ่มโดยอิงถึงความใกล้เคียงของข้อมูล เพื่อคาดเดาหรือจำแนกประเภทข้อมูลใหม่ [?] โดยหลักการทำงานสามารถสรุปได้ดังนี้

1. **เลือกค่า K** : กำหนดค่า K ที่ต้องการ ซึ่งเป็นจำนวนของข้อมูลที่ใกล้เคียงที่สุดที่จะใช้ในการตัดสินใจ
2. **คำนวณระยะทาง** : ใช้ระยะทางยูคลิดีียน (Euclidean distance) เพื่อคำนวณหาความคล้ายคลึงระหว่างข้อมูล
3. **หาข้อมูลที่ใกล้เคียงที่สุด** : หลังจากคำนวณระยะทางระหว่างข้อมูลทดสอบกับข้อมูลในชุดข้อมูลการฝึกฝน เราจะเลือกข้อมูล K รายการที่มีระยะทางน้อยที่สุด
4. **คำนวณผลโหวต** : เมื่อเราได้ข้อมูล K รายการที่ใกล้เคียงที่สุดแล้ว เราจะนับจำนวนรายการในแต่ละกลุ่มหรือประเภทข้อมูล และกำหนดกลุ่มหรือประเภทข้อมูลของข้อมูลทดสอบตามจำนวนที่มากที่สุดใน K รายการนั้น
5. **ทำนายผลลัพธ์** : สุดท้ายเราก็ได้กลุ่มข้อมูลที่ถูกแบ่งออกมาพร้อมใช้ในการทำนายต่อไป



รูปที่ 2.5 ลักษณะการทำงานของ K-Nearest Neighbors

2.3.2 อัลกอริทึม II Naive Bayes Classifier

Naive Bayes Classification เป็นหนึ่งใน Classification Model ใช้ในการแบ่งกลุ่มหรือหาเหตุการณ์ที่จะเกิดขึ้นโดยการอิงทฤษฎีความน่าจะเป็นของ Bayes หรือ Bayesian

ซึ่งจะคำนวณว่าจะเกิดเหตุการณ์นั้นหรือไม่โดยจะเพิ่มโอกาสในการเกิดเหตุการณ์เข้าไปด้วย โดยมักใช้ในการวิเคราะห์ข้อมูลที่มีความต่อเนื่องของเหตุการณ์ (Dependent Event) เช่น โอกาสในการเกิดโรคในกลุ่มประชากรที่เราสนใจ [?, ?] ซึ่งจำเป็นจะต้องอาศัยการคำนวณผ่านสูตรดังนี้ และกำหนดให้

$$P(A|B) = \frac{P(B|A) P(A)}{P(B)}$$

รูปที่ 2.6 สมการความน่าจะเป็นของ Bayes หรือ Bayesian

$P(A|B)$ คือความน่าจะเป็นในการเกิดเหตุการณ์ A โดยมี B เป็น Condition

$P(B|A)$ คือความน่าจะเป็นในการเกิดเหตุการณ์ B โดยมี A เป็น Condition

$P(A)$ คือโอกาสในการเกิดเหตุการณ์ A จากเหตุการณ์ทั้งหมด

$P(B)$ คือโอกาสในการเกิดเหตุการณ์ B จากเหตุการณ์ทั้งหมด

2.4 ภาษาคอมพิวเตอรืและเทคโนโลยี

2.4.1 เครื่องมือในการทำโครงการ

2.4.1.1 Figma

เครื่องมือที่ใช้ในการออกแบบส่วนต่อประสานผู้ใช้ (User Interface) รวมถึงระดมสมอง (Brain Storm) และใช้สำหรับให้นักออกแบบสื่อสารกับผู้อื่นให้เข้าใจได้ง่ายและเห็นภาพมากยิ่งขึ้น เช่น การทำ User Persona, User Journey, User Flow, Site Map รวมถึงการทำ Prototype เพื่อนำมาใช้ในการทดสอบการใช้งานของผู้ใช้ (Usability Test) และยังสามารถออกแบบดีไซน์ของเว็บไซต์ (Design System) เพื่อให้นักพัฒนาสามารถนำสไตล์ไปใช้พัฒนาเว็บไซต์ได้ง่ายและเป็นระเบียบมากยิ่งขึ้น

2.4.1.2 Jira

เครื่องมือที่ช่วยในการพัฒนาซอฟต์แวร์ทางอ้อม มีส่วนช่วยในการติดตามและจัดการงานต่าง ๆ ได้ถึงระดับย่อย เหมาะสมกับการทำงานในรูปแบบ Agile อีกทั้งยังสามารถผูกกับบริการต่าง ๆ ภายนอกได้ ทำให้การติดตามงานสะดวกยิ่งขึ้นไปอีก

2.4.2 ภาษาโปรแกรมที่ใช้

2.4.2.1 Typescript

ภาษาที่พัฒนาย่อยออกมาจาก Javascript เพื่อปรับปรุงจุดด้อยต่าง ๆ เช่น การจัดการ interface และประเภทของตัวแปร การเขียนโค้ดที่สนับสนุนรูปแบบ OOP ที่ดีกว่า การตรวจจับข้อผิดพลาดและการรับมือที่ดีกว่า โดยทางเราจะนำมาใช้งานกับ framework ต่าง ๆ ที่กล่าวไปข้างต้น เพื่อให้การทำงานมีประสิทธิภาพสูงสุด

2.4.2.2 Python

ภาษาโปรแกรมที่จัดอยู่ในประเภทระดับสูง ซึ่งถูกนำมาใช้อย่างแพร่หลายในปัจจุบัน มีลักษณะไวยากรณ์ไม่ซับซ้อน สามารถใช้พัฒนาเว็บแอปพลิเคชัน รวมถึงนำไปประยุกต์พัฒนาปัญญาประดิษฐ์ได้หลายประเภท มีไลบรารีจำนวนมาก อีกทั้งยังชุมชนใหญ่ ซึ่งในนี้เราจะนำมาใช้สำหรับการพัฒนาปัญญาประดิษฐ์ของเรา

2.4.3 เทคโนโลยีจัดการระบบหน้าบ้าน (Front-End)

2.4.3.1 NextJS

React framework ซึ่งสามารถใช้สร้าง full-stack web applications ได้ แต่ทางเราจะมุ่งเน้นไปที่การนำมาใช้พัฒนาส่วน frontend โดย NextJS จะมีความสามารถในการเขียนส่วนปฏิสัมพันธ์กับผู้ใช้ในเชิง components เหมือนกับ React และมีส่วนเสริมต่าง ๆ เพื่อให้การทำงานของเรานั้นเป็นไปอย่างมีประสิทธิภาพมากขึ้น เช่น routing, caching , optimizing, configuration ต่าง ๆ

2.4.4 เทคโนโลยีจัดการระบบหลังบ้าน (Back-End)

2.4.4.1 NestJS

Framework ที่ใช้สำหรับพัฒนาระบบ backend ซึ่งเขียนด้วยภาษา TypeScript โดยมีพื้นฐานของการพัฒนา framework มาจาก Express และ Fastify โดยมุ่งเน้นให้นักพัฒนาสามารถพัฒนาระบบ backend ได้อย่างรวดเร็ว และสนับสนุนหลักการทำงานต่าง ๆ ที่นักพัฒนายานิยมใช้งาน โดยทางเราเองก็จะใช้ความสามารถในส่วนนี้เพื่อนำมาใช้พัฒนาเช่นกัน อาทิ REST API และ MVC structure

2.4.5 บริการคลาวด์ (Cloud Service)

2.4.5.1 Google Cloud Platform (GCP)

บริการในรูปแบบ cloud ของทาง Google ซึ่งประกอบด้วยบริการหลากหลายรูปแบบ โดยทางเรามุ่งเน้นที่จะใช้งานบริการส่วนของการหลัก ๆ ดังนี้

- **Cloud Run** : บริการสำหรับการนำ container ที่มีรันบน cloud โดยเราจะนำบริการนี้มา host ทั้งในส่วนของหน้าบ้านและหลังบ้านของเว็บแอป
- **Cloud Build** : บริการสำหรับนำไฟล์โปรเจกต์มา build ให้อยู่ในรูปแบบ container โดยเราจะนำมาใช้คู่กับ docker เพื่อนำโปรเจกต์ส่งสู่ Cloud Run
- **Cloud Storage** : บริการสำหรับเก็บไฟล์มีเดียต่าง ๆ เช่น รูปภาพ เสียง วิดีโอ เพื่อนำมาเก็บข้อมูลของผู้ใช้ที่เป็นมีเดีย
- **Artifact Registry** : บริการสำหรับเก็บไฟล์ container ต่าง ๆ ที่มีทั้งหมด โดยนำมาใช้คู่กับ Cloud Build เพื่อเก็บ container ที่ผ่านการ build เสร็จแล้ว

2.4.6 ระบบฐานข้อมูล

2.4.6.1 MongoDB

Open-source database ประเภท NoSQL ที่มีโครงสร้างแบบ document ซึ่งได้รับความนิยมอย่างสูงเนื่องจากยืดหยุ่นและปรับขนาดได้ง่าย อีกทั้งยังรองรับภาษาที่หลากหลายทั้ง Javascript, Python, Java และอื่น ๆ โดยทางเราเลือกใช้ MongoDB เพราะมีเงื่อนไขที่ยืดหยุ่นเหมาะกับการทำงานของเรา เช่น ไม่จำกัดจำนวนคำขอ API ต่อวัน และมีพื้นที่จัดเก็บที่พอเหมาะอยู่ที่ 512 GB ในระดับการใช้งานฟรี

2.4.7 เครื่องมือช่วยเหลือการพัฒนา (CI/CD Management)

2.4.7.1 Docker

เครื่องมือที่สามารถช่วยจำลองสภาพแวดล้อมของเซิร์ฟเวอร์ด้วยหลักการ container และ images ทำให้จัดปัญหาละเอียดอ่อน เช่น ปัญหาสภาพแวดล้อมในแต่ละเครื่องไม่เหมือนกันส่งผลให้รันไม่ได้ โดยเราจะนำ docker มาใช้เป็นส่วนช่วยเหลือในการนำบริการไปปล่อยสู่สาธารณะผ่าน Google Cloud

2.4.7.2 Github

ระบบควบคุมเวอร์ชัน สามารถสร้างจุด commit เพื่อเสมือนเป็นจุดบันทึกเวอร์ชันหนึ่ง และสามารถนำไปเก็บบนที่เก็บรวม (repository) เพื่อควบคุมเวอร์ชันของไฟล์งาน กับบุคคลภายในทีมได้ โดยทางเราจะนำมาใช้เพื่อควบคุมเวอร์ชันของซอฟต์แวร์โครงการ เพื่อให้ทำงานได้อย่างลื่นไหล เช่น การแยกสาขาของงานตามฟีเจอร์ การติดตามเวอร์ชันของโค้ดเพื่อค้นหาจุดกำเนิดของข้อผิดพลาด อีกทั้งยังระบุบุคคลผู้รับผิดชอบการทำงานส่วนนั้นได้อีกด้วย

2.5 การศึกษาข้อมูลผลิตภัณฑ์ที่ใกล้เคียง

2.5.1 หนังสือหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ มจร.

เป็นหนังสือที่บอกถึงรายละเอียดของแต่ละรายวิชาที่นักศึกษาวิศวกรรมคอมพิวเตอร์ได้เรียนในหลักสูตรตั้งแต่ปี 1 ถึงปี 4 โดยจะแสดงออกมาเป็นผลลัพธ์การเรียนรู้และทักษะที่ได้จากการเรียนในวิชาต่าง ๆ แต่จะไม่ได้บอกถึงอาชีพที่สามารถนำไปต่อยอดจากรายวิชาได้

ซึ่งทางคณะผู้จัดจะนำข้อมูลรายวิชาในแต่ละปีการศึกษามาศึกษาแบ่งว่า แต่ละรายวิชาสามารถนำไปต่อยอดทางใดได้บ้าง เพื่อมาวางแผนเส้นทางการลงวิชาเลือกที่สัมพันธ์กับระดับการศึกษาและความสนใจของผู้ใช้งานแต่ละคน

2.5.2 LinkedIn

LinkedIn เป็นเว็บแอปพลิเคชันชุมชนในสายอาชีพต่าง ๆ ที่ช่วยให้ผู้ใช้สร้างโปรไฟล์อาชีพของตนเองและเชื่อมโยงกับคนที่ใกล้เคียงในสายอาชีพ เว็บไซด์นี้จะช่วยให้ผู้ใช้สามารถสร้างเครือข่ายในสายอาชีพของตน แบ่งปันข้อมูลเกี่ยวกับประสบการณ์การทำงาน ประวัติการศึกษา ทักษะ ความถนัด รวมถึงเผยแพร่เนื้อหาที่เกี่ยวข้องกับสาขาอาชีพของพวกเขาในรูปแบบข่าวสาร ทำให้เชื่อมโยงและแลกเปลี่ยนข้อมูลกับผู้อื่น ๆ ในวงการได้ตลอดเวลา

โดยที่เว็บไซด์มีฟังก์ชันหลายอย่าง ประกอบด้วย :

- **โปรไฟล์ผู้ใช้ :** ผู้ใช้สามารถสร้างโปรไฟล์ส่วนตัวที่แสดงประสบการณ์การทำงาน การศึกษา ทักษะ และข้อมูลอื่นๆ เพื่อให้ผู้อื่นสามารถทราบรายละเอียดเบื้องต้นของพวกเขาได้
- **การเชื่อมโยง :** ผู้ใช้สามารถเชื่อมโยงกับคนอื่นในสายอาชีพ ทำให้สร้างเป็นเครือข่ายในสายอาชีพที่แข็งแกร่งขึ้น และเป็นโอกาสที่ดีให้กับผู้อื่นได้
- **โพสต์และเนื้อหา :** ผู้ใช้สามารถโพสต์เนื้อหาเกี่ยวกับวงการอาชีพ เช่น บทความ ข่าวสาร และความคิดเห็น ซึ่งช่วยในการแบ่งปันความรู้และประสบการณ์
- **ค้นหางาน :** ผู้ใช้สามารถค้นหางานและสมัครงานได้โดยตรงผ่านแพลตฟอร์ม และผู้ประกาศงานก็สามารถค้นหาผู้สมัครที่เหมาะสมกับตำแหน่งงานที่วางอยู่ของตนเองได้โดยง่าย
- **กลุ่มองค์กร :** บริษัทและองค์กรสามารถสร้างหรือเข้าร่วมกลุ่มบน LinkedIn เพื่อแบ่งปันข้อมูลและความรู้ในหมวดหมู่ที่เกี่ยวข้องได้ภายในกลุ่มที่กำหนดเองได้
- **การแสดงความคิดเห็น :** ผู้ใช้สามารถกดถูกใจ แสดงความคิดเห็น หรือแชร์เนื้อหาของผู้อื่น เพื่อแสดงความรับรู้ สนใจ หรือช่วยในการประกาศข่าวสารที่ดี

ซึ่งถือว่าเป็นเว็บแอปพลิเคชันที่เกี่ยวข้องกับการทำงานที่มีความสามารถที่สูง และเน้นไปที่การสร้างคอมมูนิตี้สำหรับการทำงาน เนื่องด้วยคุณสมบัติที่หลากหลายนี้ ทำให้มีผู้ใช้งานเป็นจำนวนมาก ซึ่งทางคณะผู้จัดทำจะนำระบบชุมชนที่สามารถแนะนำงานกับกิจกรรมและระบบจัดเก็บเรซูเม่มาต่อยอดกับโครงงานของเราให้ดียิ่งขึ้น

2.5.3 JobDB

เป็นเว็บแอปพลิเคชันที่รวบรวมตำแหน่งงานต่าง ๆ ในประเทศไทยที่กำลังเปิดรับอยู่ ที่จัดทำขึ้นสำหรับผู้คนที่กำลังมองหางาน โดยที่เว็บไซด์มีฟังก์ชันหลายอย่าง ประกอบด้วย :

- ระบบค้นหาอาชีพที่ต้องการ
- โพสต์ที่จะมีรายละเอียดงานที่เปิดรับ
- ระบบสมัครงาน
- คำแนะนำสำหรับการจัดทำเอกสารการสมัครงาน
- การแจ้งเตือนสำหรับตำแหน่งงานที่ผู้ใช้งานสนใจ

ซึ่งเป็นเว็บแอปพลิเคชันรวบรวมตำแหน่งงานที่เน้นกลุ่มเป้าหมายเป็นผู้ที่กำลังหางานในประเทศไทย โดยรวมมีระบบที่คอยอำนวยความสะดวกในการค้นหางานที่ผู้ใช้งานสนใจ และยังมีฟังก์ชันที่น่าสนใจเป็นอย่างมากกับ ระบบคำแนะนำสำหรับการจัดทำเอกสารการสมัครงาน ซึ่งทางผู้จัดทำโครงงานจะนำฟังก์ชันนี้มาต่อยอดกับโครงงานต่อไป

2.5.4 Padlet

เป็นเว็บแอปพลิเคชันที่ให้บริการบอร์ดข้อความ ที่ผู้ใช้งานสามารถเพิ่มข้อมูลต่าง ๆ ได้ ซึ่งสามารถนำมาประยุกต์ใช้ได้หลาย ๆ วัตถุประสงค์ เช่น การนำมาใช้รื้อรายวิชาเรียนภายในกลุ่มที่กำหนด โดยที่เว็บไซต์มีฟังก์ชันหลายอย่าง ประกอบด้วย :

- **ระบบสร้างหน้า padlet หรือการหน้ากระดานใหม่** : โดยมีให้เลือกรูปแบบของกระดานมากมาย เช่น รูปแบบ wall, canvas, stream, Grid และอื่น ๆ ที่เหมาะสมกับจุดประสงค์ที่ต้องการ
- **ระบบเข้าร่วมและแบ่งปัน padlet** : ทำให้สามารถเข้าไปแก้ไขหรือดูหน้า padlet ของผู้อื่นได้
- **ระบบแกลเลอรี** : รวมคลังหน้า padlet ให้กับผู้ใช้
- **ระบบเครือข่าย** : สามารถเชื่อมโยงผู้ใช้งานเว็บไซต์ Padlet เข้าด้วยกันเพื่อเสริมฟังก์ชันอื่น ๆ ได้
- **ระบบการทำสื่อนำเสนอ** : โดยที่จะรวบรวม padlet ต่าง ๆ มาทำเป็นสไลด์ และยังมี QR-Code ที่สามารถเข้ามาดูสไลด์ได้อีกด้วย
- **ระบบแจ้งเตือน** : ที่สามารถเลือกติดตาม Padlet ที่ตนเองสนใจได้
- **ระบบเชื่อมต่อบริการภายนอก** : ที่รวบรวมบริการไว้มากมาย เช่น ข้อมูลที่ผู้ใช้งานฝากไฟล์ออนไลน์ไว้ แม้กระทั่งระบบปัญญาประดิษฐ์ที่สามารถเพิ่มรูปภาพตามความต้องการของผู้ใช้ และบริการอื่น ๆ ที่เชื่อมต่อจากภายนอก โดยมีจุดแข็งตรงที่เว็บไซต์มีบริการภายนอกเหล่านี้จำนวนมาก

ทางคณะผู้จัดทำเล็งเห็นว่านักศึกษาวิศวกรรมคอมพิวเตอร์หลายคน ที่จะมาอ่านรื้อรายวิชาก่อนที่จะเลือกลงในวิชาเลือกของตนเอง แต่ก็ยังมีข้อเสียที่ข้อมูลอาจจะไม่ได้อัปเดตมากเท่าที่ควร และไม่ได้จัดรูปแบบให้สามารถอ่านได้ง่าย ทางคณะผู้จัดทำจึงอยากจะนำข้อดีของบอร์ดรื้อรายวิชาใน Padlet มาปรับปรุงให้ทันสมัย และอ่านได้ง่ายยิ่งขึ้น เพื่อลดข้อเสียของการใช้งาน

2.5.5 JobThai

เป็นเว็บแอปพลิเคชันสมัครงานที่มีกลุ่มเป้าหมายเป็นคนที่กำลังมองหางานในประเทศไทย ครอบคลุมหลากหลายอาชีพ โดยที่เว็บไซต์มีฟังก์ชันหลายอย่าง ประกอบด้วย :

- **ระบบสมัครสมาชิก** : โดยมีทั้งฝั่งของผู้ที่กำลังหางาน และผู้ที่กำลังต้องการลูกจ้าง ซึ่งแต่ละฝ่ายก็จะมีฟังก์ชันที่รองรับ เช่น ผู้ที่กำลังหางานก็จะสามารถฝากประวัติได้
- **ระบบค้นหางาน** : โดยสามารถคัดกรองได้ด้วยอาชีพที่ต้องการ สถานที่ทำงาน บริษัทที่เปิดรับ ประเภทของธุรกิจ รวมไปถึงเงินเดือนอีกด้วย
- **โพสต์** : ที่จะมีรายละเอียดงานที่เปิดรับ

ซึ่งเว็บแอปพลิเคชันนี้จะเน้นไปที่ผู้ใช้งานที่อยู่ในประเทศไทย และยังมีฟังก์ชันเลือกสถานที่ทำงานที่อยู่ใกล้กับสถานีรถไฟฟ้า นิคมอุตสาหกรรม หรือแม้กระทั่งใกล้กับรถเมล์ ซึ่งถือว่าทำมาเพื่อตอบสนองกับความต้องการของผู้ใช้งานในกรุงเทพที่ติมาก ๆ เพราะการเดินทางก็ถือเป็นหนึ่งในปัจจัยสำคัญในการเลือกงานในปัจจุบัน

2.5.6 Workday

เป็นเว็บแอปพลิเคชันที่มุ่งเน้นไปในการจัดการทรัพยากรบุคคล โดยที่ถูกรออกแบบมาเพื่อรองรับการทำงานสำหรับองค์กรขนาดกลาง จนถึงองค์กรขนาดใหญ่ โดยที่เว็บไซต์มีฟังก์ชันหลายอย่าง ประกอบด้วย :

- **การจัดการข้อมูลพนักงาน** : รวมถึงการจัดการการสร้าง แก้ไข และยุติข้อมูลการทำงานของพนักงาน เช่น ข้อมูลส่วนตัว การจ้างงาน การเลื่อนตำแหน่ง การลางาน เป็นต้น
- **การจัดการงบประมาณ** : การคำนวณเงินเดือน การจ่ายเงินเดือน และการจัดการสวัสดิการสำหรับพนักงาน เช่น ประกันสุขภาพ กองทุนสำรองเลี้ยงชีพ เป็นต้น

- **การวางแผนแบบครบวงจร :** ตั้งแต่ขั้นตอนการรับสมัครงาน การบรรจงาน การพัฒนาพนักงาน และการเลื่อนขั้น

ซึ่งเว็บแอปพลิเคชันนี้จะเน้นไปที่การให้บริการเกี่ยวกับการดูแลข้อมูลของพนักงาน ที่มีระบบที่น่าสนใจอย่างการวางแผนพัฒนาพนักงาน รวมถึงการเลื่อนขั้น ที่ทางคณะผู้จัดทำจะนำมาพัฒนาต่อยอดกับการพัฒนาผู้ใช้งานที่เป็นนักศึกษาวิศวกรรมคอมพิวเตอร์ต่อไป

2.5.7 Camphub

เป็นเว็บแอปพลิเคชันที่รวบรวมค่ายต่าง ๆ ในประเทศไทยที่กำลังเปิดรับสมัครอยู่ ที่จัดทำขึ้นมาสำหรับเด็กประถมจนกระทั่งรวมไปถึงบุคคลทั่วไป

โดยที่เว็บไซต์มีฟังก์ชันหลายอย่าง ประกอบด้วย :

- การประชาสัมพันธ์ค่ายของตนเอง โดยที่ไม่ต้องเสียค่าใช้จ่าย
- การแบ่งประเภทของค่ายหรือมหาวิทยาลัยที่จัด เพื่อง่ายต่อการค้นหา
- รายละเอียดของค่าย เช่น รูปแบบกิจกรรม, วันที่จัดกิจกรรม, จำนวนที่รับ เป็นต้น
- การสมัครค่ายที่ตนเองสนใจ
- บทความต่าง ๆ ที่น่าสนใจ

ซึ่งเว็บแอปพลิเคชันนี้จะเน้นไปที่การรวบรวมข่าวสารกิจกรรมที่เกิดขึ้น ไม่ว่าจะเป็นค่ายแนะนำการศึกษา ค่ายให้ความรู้ ทางคณะผู้จัดทำจึงจะทำการจัดการประชาสัมพันธ์กิจกรรมนี้ ไปใช้กับการแนะนำแนวทางการศึกษา หรือแนวทางการพัฒนาตนเองของผู้ใช้งานโครงการของพวกเขาต่อไป และจะทำให้ดียิ่งขึ้นด้วยการเก็บประวัติการเข้าร่วมของผู้ใช้งาน เพื่อที่จะนำมาคำนวณความเป็นไปได้ในการพัฒนาเส้นทางอาชีพต่อไป

2.5.8 Fuel50

เป็นเว็บแอปพลิเคชันที่มีฟังก์ชัน Career Journey ที่จะให้ผู้ใช้งานสามารถกำหนดตำแหน่งงานในปัจจุบัน และตำแหน่งงานในอนาคตที่ต้องการจะเป็น ซึ่งตัวเว็บไซต์จะมีระบบแนะนำตั้งแต่ตำแหน่งที่จำเป็นต้องเป็นก่อนจะถึงจุดหมาย รวมไปถึงทักษะที่ต้องพัฒนา และทักษะที่จำเป็นต้องเพิ่มเพื่อที่จะสามารถพัฒนาตำแหน่งงานไปได้อย่างมีประสิทธิภาพ

ซึ่งนักศึกษามองว่าฟังก์ชัน Career Journey นี้จะมีประโยชน์อย่างมากสำหรับนักศึกษาวิศวกรรมคอมพิวเตอร์ที่เป็นผู้ใช้งานของเรา ทางคณะผู้จัดทำจึงอยากที่จะนำมาปรับปรุงและแก้ไขเพื่อมาใช้กับแนวทางการเลือกวิชา (Class Journey) เพื่อให้ผู้ใช้งานเห็นภาพการพัฒนาตนเองที่ชัดเจนยิ่งขึ้น และทำให้ผู้ใช้งานมีแรงจูงใจในการพัฒนาตนเองต่อไป

2.5.9 Super Resume

ซูเปอร์เรซูเม่ (Super Resume) คือ แพลตฟอร์มสร้างเรซูเม่ออนไลน์ที่มีผู้ใช้งานมากกว่า 2 ล้านคนในประเทศไทย รูปแบบของซูเปอร์เรซูเม่ได้รับการพัฒนาโดยบริษัทชั้นนำและได้รับการยอมรับจาก HR ของบริษัทชั้นนำกว่า 30,000 บริษัท ซูเปอร์เรซูเม่มีรูปแบบที่เป็นมาตรฐานและครบถ้วน ครอบคลุมข้อมูลสำคัญของผู้สมัครงาน ได้แก่ ข้อมูลส่วนตัว ข้อมูลการศึกษา ข้อมูลประสบการณ์การทำงาน ข้อมูลทักษะและความสามารถ และข้อมูลความถนัดและบุคลิกภาพ

ข้อดีของการใช้ซูเปอร์เรซูเม่ในการสมัครงาน ประกอบด้วย :

- HR ที่คุ้นเคยกับรูปแบบของซูเปอร์เรซูเม่จะช่วยให้สามารถอ่านและเข้าใจข้อมูลของผู้สมัครได้อย่างรวดเร็วและง่ายดาย
- ซูเปอร์เรซูเม่มีข้อมูลที่ครบถ้วนและครอบคลุม ทำให้ผู้สมัครสามารถนำเสนอข้อมูลของตนเองได้อย่างมีประสิทธิภาพ
- ซูเปอร์เรซูเม่สามารถจัดส่งไปยังบริษัทที่ตรงกับความต้องการของผู้สมัครได้โดยตรง
- ซูเปอร์เรซูเม่ยังมีฟังก์ชันที่ช่วยให้ผู้สมัครสามารถติดตามสถานะการสมัครงานและปรับปรุงเรซูเม่ของตนเองได้อีกด้วย

โดยสรุป ซูเปอร์เรซูเม่เป็นแพลตฟอร์มสร้างเรซูเม่ออนไลน์ที่มีประสิทธิภาพและช่วยให้ผู้สมัครงานมีโอกาสในการสมัครงานและสัมภาษณ์งานมากขึ้น ซึ่งคณะผู้จัดทำได้เล็งเห็นว่าการสร้างเรซูเม่เป็นสิ่งที่สำคัญมาก จึงอยากที่จะมีระบบที่ให้คำแนะนำเกี่ยวกับเรซูเม่ขึ้นมาในโครงการของเรา เพื่อให้ผู้ใช้งานจะสามารถทราบได้ว่าควรที่จะเพิ่มเติมรายละเอียดของเรซูเม่อย่างไรบ้าง

2.5.10 JobHack (Resume Checker)

เป็นเว็บแอปพลิเคชันสำหรับการตรวจสอบคุณภาพเรซูเม่ ว่ามีความเหมาะสมกับตำแหน่งที่ผู้ใช้งานกำลังสนใจหรือไม่ โดยจะให้คะแนนออกมา และยังแนะนำส่วนที่ขาดหายอีกทั้งยังมีแนวคำถามที่ผู้สัมภาษณ์อาจจะถามอีกด้วย

ซึ่งนักศึกษามองว่าการนำ Artificial Intelligence มาตรวจสอบคุณภาพของเรซูเม่เป็นฟังก์ชันที่น่าสนใจเป็นอย่างมาก แต่ทาง JobHack ยังคงมีความแม่นยำที่น้อย ซึ่งทางคณะผู้จัดทำมองว่าเป็นสิ่งที่ดีหากสามารถนำมาพัฒนาต่อให้มีประสิทธิภาพมากยิ่งขึ้น

2.5.11 Competitor Analysis

ตารางที่ 2.1 ตารางเปรียบเทียบคุณสมบัติที่สนใจ

	การแนะนำการทำเรซูเม่	แผนภาพสายอาชีพ	ชุมชนแนะนำงานและกิจกรรม	เก็บสะสมเรซูเม่
* Compath	✓	✓	✓	✓
LinkedIn			✓	✓
JobDB			✓	✓
Padlet			✓	
JobThai			✓	✓
Workday			✓	✓
Camphub			✓	
Fuel50		✓		
Super Resume	✓			✓
JobHack	✓			✓

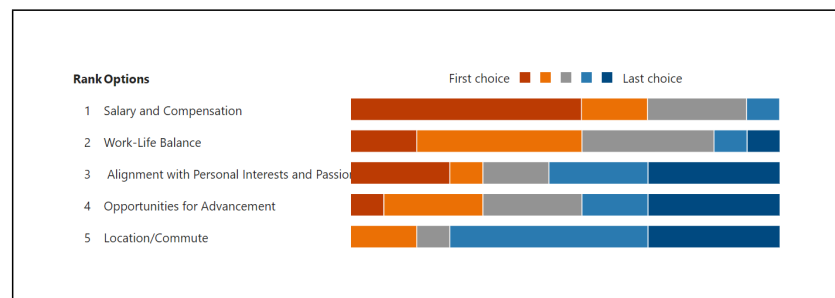
บทที่ 3 การออกแบบและวิธีการดำเนินงาน

3.1 การสำรวจความต้องการกับผู้ใช้

ทางคณะผู้จัดทำได้ทำออกเก็บข้อมูลกับกลุ่มเป้าหมายมาแล้วทั้งหมด 4 ครั้ง โดยแบ่งเป็นการสัมภาษณ์เชิงปริมาณหนึ่งครั้งและเชิงคุณภาพสามครั้ง โดยมีจุดประสงค์ในแต่ละการสัมภาษณ์ต่างกันเพื่อพิสูจน์ความต้องการของกลุ่มเป้าหมายจนกระทั่งโครงการของเราได้ปรับตามความต้องการนั้นจนเป็นโครงการในปัจจุบัน อย่างไรก็ตาม เรายังได้รับข้อมูลที่น่าสนใจเพิ่มเติมโดยสรุป ดังนี้

3.1.1 การสัมภาษณ์เชิงปริมาณผ่านแบบสำรวจ

ทางคณะผู้จัดทำได้ทำแบบสอบถามเพื่อหาอัตราส่วนของพฤติกรรมที่น่าสนใจ โดยได้รับข้อมูลที่สำนัสนใจดังนี้ จากข้อมูลในแบบ



รูปที่ 3.1 ข้อมูลจากแบบสำรวจเชิงปริมาณ

สำรวจข้อนี้ ทำให้ทราบว่าปัจจัยที่มีผลต่อการเลือกสายงานในการทำงานมากสำหรับกลุ่มตัวอย่างก็คือเงินเดือน และความสมดุลของการทำงานกับชีวิตส่วนตัว กล่าวคือกลุ่มตัวอย่างให้ความสำคัญกับความเจริญก้าวหน้า ความมั่นคง หรือเงินเดือนที่สามารถทำให้ดำเนินชีวิตได้อย่างราบรื่น รวมถึงการแบ่งเวลา การพักผ่อนที่เหมาะสมกับการทำงาน โดยมองอาจจะไม่ได้ให้ความสำคัญกับความชื่นชอบมากขนาดนั้น

3.1.2 การสัมภาษณ์เชิงคุณภาพผ่านการสัมภาษณ์ตัวต่อตัว

ทางคณะผู้จัดทำได้ออกสัมภาษณ์กับกลุ่มเป้าหมายแบบตัวต่อตัว และในการสัมภาษณ์แต่ละครั้ง จะสัมภาษณ์ที่จำนวนราว 5 คน ตามหลักการ 5 Users design โดยได้รับข้อมูลที่น่าสนใจในแต่ละครั้งมาดังนี้

การสัมภาษณ์เชิงคุณภาพครั้งที่ 1

จุดประสงค์: เพื่อค้นหารากขอปัญหาที่แท้จริงของกลุ่มเป้าหมาย

ข้อมูลสำคัญ:

- กลุ่มเป้าหมายค้นพบความต้องการของตนเองมาจากการได้ลงมือทำจริงเป็นหลัก
- ส่วนใหญ่แล้วจะได้ลงมือทำจริงตอนโปรเจกต์วิชาเรียนหรือฝึกงาน ซึ่งอยู่ชั้นปีที่ 2 เป็นต้นไปแล้ว
- ยอมรับก่อนเป็นอย่างมาก ว่าในการทำงานจริงต้องมีความสามารถอะไรบ้าง ใช้เครื่องมืออะไร
- กลุ่มเป้าหมายรู้สึกว่าการรู้ตัวช้าเกินไป หากมีโอกาสพัฒนาตนเองได้เร็วกว่านี้จะดีมาก

การสัมภาษณ์เชิงคุณภาพครั้งที่ 2

จุดประสงค์: เพื่อพิสูจน์ความมีคุณภาพของวิธีการแก้ปัญหาที่ออกแบบ

ข้อมูลสำคัญ:

- กลุ่มเป้าหมายอยากได้ตัวช่วยในการทำให้ตนเองสมัครงานได้ง่ายขึ้น โดยหลังการทดลองถามความเห็น พบว่าสิ่งที่ต้องการเป็นหลัก คือ การตรวจสอบและยืนยันได้ ว่าเรซูเม่ของตนเองเหมาะสมกับอาชีพที่ตนเองสนใจขนาดไหนแล้ว
- กลุ่มเป้าหมายรู้สึกสนใจในฟีเจอร์ช่วยเหลือการค้นหาแหล่งพัฒนาตนเอง เพราะเคยรู้สึกว่าตนเองอาจเริ่มพัฒนาช้าเกินไป เพราะรู้ใจตัวเองในช่วงที่อาจเรียนอยู่ชั้นปีที่ 2-3 แล้ว

- การตัดสินใจลงวิชาเรียนค่อนข้างมีจุดขัดใจ เพราะไม่ค่อยมีรีวิวหรือความเห็นของผู้ที่เคยเรียน ถึงแม้จะมีแหล่งชุมชนที่รุ่นพี่เคยมอบให้ แต่รีวิวส่วนใหญ่จะมีความเก่าแล้ว ทำให้ใช้อ้างอิงได้ยาก

การสัมภาษณ์เชิงคุณภาพครั้งที่ 3

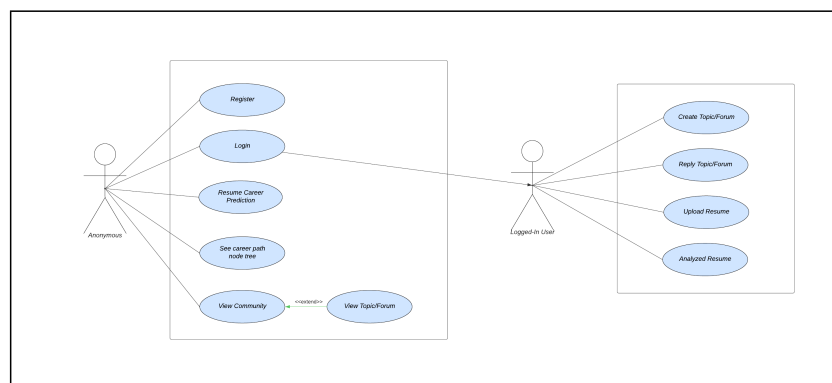
จุดประสงค์: เพื่อทดลองนำ prototype ของเว็บแอปไปพิสูจน์ความรู้สึกในการใช้งานกับผู้ใช้

ข้อมูลสำคัญ:

- ผู้ใช้ไม่มีปัญหาเกี่ยวกับการกรอกข้อมูลเรซูเม่เพื่อวิเคราะห์ แต่หากมีระบบที่กรอกข้อมูลให้อัตโนมัติผ่านไฟล์ pdf ก็ถือว่าเป็นเรื่องดี
- การมีระบบโหวตวิชาที่อยากให้เปิด อาจไม่ได้เป็นการการันตีว่าจะเปิดได้จริง อาจไม่สำคัญมากนัก
- ในอนาคต หากมีระบบที่เป็นตัวช่วยในการสร้างเรซูเม่จากข้อมูลที่มีได้ ก็จะเป็นเรื่องที่ดีเช่นกัน
- ควรมีการแย่ง tag ประเภทของชุมชนเพื่อความสะดวกในการค้นหา

3.2 ความสามารถของระบบ

3.2.1 Use Case Diagram



รูปที่ 3.2 Use Case Diagram

3.2.2 Use Case Narrative

3.2.2.1 Resume Career Prediction

Actor	Anonymous
Goal	ต้องการทราบถึงอาชีพที่เหมาะสมกับตนเอง
Precondition	-
Main success scenario	<ol style="list-style-type: none"> 1. ผู้ใช้หาคำแนะนำอาชีพที่เหมาะสมกับตนเอง 2. ระบบถามข้อมูลของผู้ใช้ 3. ผู้ใช้กรอกข้อมูล 4. ระบบถามเพื่อยืนยันการกรอกข้อมูล 5. ผู้ใช้ยืนยันการกรอกข้อมูล 6. ระบบแสดงอาชีพที่เหมาะสมกับผู้ใช้
Extensions (a)	<ol style="list-style-type: none"> 5a. ผู้ใช้กรอกข้อมูลไม่ครบ 6a. ระบบขึ้นเตือนว่าผู้ใช้ยังกรอกข้อมูลไม่ครบ 7a. กลับไปที่ขั้นตอนที่ 3
Postcondition	ระบบแนะนำให้ผู้ใช้ไปวิเคราะห์ข้อมูลเชิงลึกของอาชีพที่ได้แนะนำไป

3.2.2.2 See career path node tree

Actor	Anonymous
Goal	ต้องการดูแผนผังสายอาชีพ
Precondition	ผู้ใช้งานกำลังดูแผนผังรวมของสายอาชีพ
Main success scenario	1. ใช้เลือกอาชีพที่ต้องการจะดู 2. ระบบแสดงรายละเอียดย่อยเป็นรูปแบบแผนผังของสายอาชีพที่ผู้ใช้เลือก
Extensions (a)	1a. ผู้ใช้ยังไม่ได้เลือกอาชีพที่ต้องการจะดู 2a. ระบบขึ้นเตือนว่าผู้ใช้ยังไม่ได้เลือกอาชีพที่ต้องการดู 3a. กลับไปที่ขั้นตอนที่ 1
Postcondition	-

3.2.2.3 View Community

Actor	Anonymous
Goal	ต้องการดูรายการของกระทู้ต่าง ๆ ในชุมชน
Precondition	ผู้ใช้งานกำลังดูประเภทของกระทู้
Main success scenario	1. ผู้ใช้เลือกประเภทของกระทู้ 2. ระบบแสดงรายการของกระทู้ประเภทที่ผู้ใช้เลือก
Extensions (a)	-
Postcondition	-

3.2.2.4 View Topic/Forum

Actor	Anonymous
Goal	ต้องการดูรายละเอียดของกระทู้
Precondition	ผู้ใช้งานกำลังดูรายการของกระทู้ที่อยู่ในชุมชน (กระทู้ประเภทไหนก็ได้)
Main success scenario	1. ผู้ใช้เลือกกระทู้ที่ต้องการดูรายละเอียด 2. ระบบแสดงรายละเอียดของกระทู้ที่ผู้ใช้เลือก
Extensions (a)	-
Postcondition	-

3.2.2.5 Create Topic/Forum

Actor	Logged-in User
Goal	ต้องการสร้างกระทู้
Precondition	ผู้ใช้งานกำลังดูรายการของกระทู้ที่อยู่ในชุมชน (กระทู้ประเภทไหนก็ได้)
Main success scenario	1. ผู้ใช้เลือกกระทู้ที่ต้องการดูรายละเอียด 2. ระบบถามรายละเอียดภายในกระทู้ 3. ผู้ใช้กรอกรายละเอียดภายในกระทู้ 4. ระบบถามเพื่อยืนยันการลงกระทู้ 5. ผู้ใช้ยืนยันการลงกระทู้ 6. ระบบสร้างและแสดงกระทู้ของผู้ใช้
Extensions (a)	5a. ผู้ใช้กรอกข้อมูลไม่ครบ 6a. ระบบขึ้นเตือนว่าผู้ใช้ยังกรอกข้อมูลไม่ครบ 7a. กลับไปที่ขั้นตอนที่ 3
Postcondition	กระทู้ของผู้ใช้แสดงอยู่ในชุมชน

3.2.2.6 Reply Topic/Forum

Actor	Logged-in User
Goal	ต้องการแสดงความคิดเห็นในกระทู้
Precondition	ผู้ใช้งานกำลังดูรายการของกระทู้ที่อยู่ในชุมชน (กระทู้ประเภทไหนก็ได้)
Main success scenario	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกส่วนที่ต้องการแสดงความคิดเห็น 2. ระบบถามถึงรายละเอียดความคิดเห็น 3. ผู้ใช้กรอกความคิดเห็น 4. ระบบถามเพื่อยืนยันการแสดงความคิดเห็น 5. ผู้ใช้ยืนยันการแสดงความคิดเห็น 6. ระบบสร้างและแสดงความคิดเห็นของผู้ใช้ในกระทู้
Extensions (a)	<ol style="list-style-type: none"> 4a. ผู้ใช้ยังไม่ได้กรอกความคิดเห็น 5a. ระบบขึ้นเตือนว่าผู้ใช้ยังไม่ได้กรอกความคิดเห็น 6a. กลับไปที่ขั้นตอนที่ 3
Postcondition	ความคิดเห็นของผู้ใช้แสดงอยู่ในกระทู้

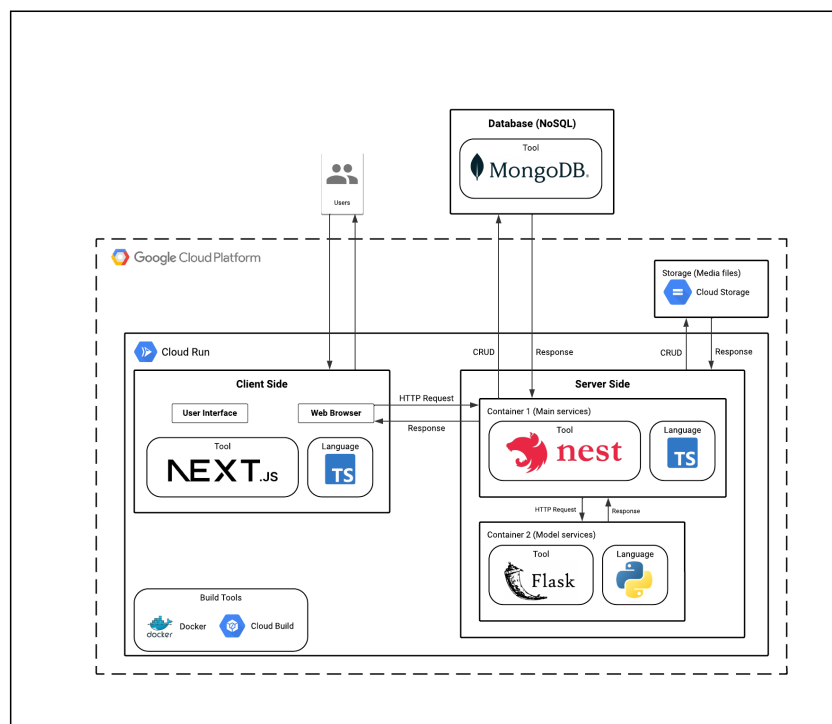
3.2.2.7 Upload Resume

Actor	Logged-in User
Goal	ต้องการอัปโหลดเรซูเม่
Precondition	ผู้ใช้งานกำลังอยู่ดูข้อมูลส่วนตัวของตนเอง
Main success scenario	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกเรซูเม่ที่ต้องการอัปโหลด 2. ระบบถามเพื่อยืนยันการอัปโหลดเรซูเม่ 3. ผู้ใช้ยืนยันการอัปโหลดเรซูเม่ 4. ระบบอัปโหลดและแจ้งเตือนว่าอัปโหลดเรซูเม่สำเร็จ
Extensions (a)	<ol style="list-style-type: none"> 3a. ผู้ใช้เลือกเรซูเม่ที่ต้องการอัปโหลด 4a. ระบบขึ้นเตือนว่าผู้ใช้ยังไม่ได้เลือกเรซูเม่ที่ต้องการอัปโหลด 5a. กลับไปที่ขั้นตอนที่ 1
Postcondition	เรซูเม่แสดงอยู่ในข้อมูลส่วนตัวของผู้ใช้

3.2.2.8 Analyzed Resume

Actor	Logged-in User
Goal	ต้องการดูข้อมูลเชิงลึกของอาชีพ
Precondition	ผู้ใช้งานต้องมีอาชีพที่ระบบแนะนำให้
Main success scenario	<ol style="list-style-type: none"> 1. ผู้ใช้เลือกอาชีพที่ต้องการดูข้อมูลเชิงลึก 2. ระบบแสดงข้อมูลเชิงลึกของอาชีพ
Extensions (a)	<ol style="list-style-type: none"> 1a. ผู้ใช้ต้องวิเคราะห์อาชีพใหม่ 2a. ระบบถามถึงข้อมูลของผู้ใช้ 3a. ผู้ใช้กรอกข้อมูลใหม่ 4a. ผู้ใช้ยืนยันการกรอกข้อมูล 5a. ระบบแสดงอาชีพที่เหมาะสมกับผู้ใช้ 6a. กลับไปที่ขั้นตอนที่ 1
Postcondition	-

3.3 สถาปัตยกรรมของระบบ



รูปที่ 3.3 System Architecture

จากแผนผังสถาปัตยกรรมระบบ เราได้ใช้บริการ Google Cloud เป็นหลักเนื่องจากสะดวกต่อการ scaling ในอนาคต สามารถจัดการและเปลี่ยนแปลงเวอร์ชันได้รวดเร็วระหว่างการพัฒนา โดยสถาปัตยกรรมแต่ละส่วนมีหน้าที่หลักดังนี้

3.3.1 ส่วนผู้ใช้ (Client-Side)

รับผิดชอบในการติดต่อปฏิสัมพันธ์กับผู้ใช้ แสดง user interface ผ่านทาง web browser และเชื่อมต่อกับ server เพื่อขอข้อมูลและใช้บริการต่าง ๆ

3.3.2 ส่วนเซิร์ฟเวอร์ (Server-Side)

รับผิดชอบในการจัดการตรรกะและระบบคำนวณต่าง ๆ ทุกรูปแบบและส่งกลับไปที่ผู้ใช้ โดยทางคณะผู้จัดทำได้แบ่งระบบ server เป็นสองบริการหลัก ประกอบด้วย

1. บริการหลัก (Main services)

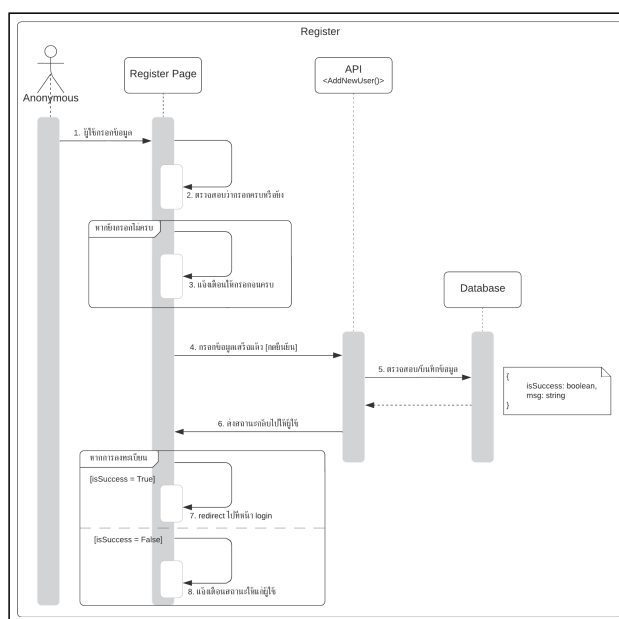
บริการที่จะติดต่อกับผู้ใช้โดยตรงเพียงหนึ่งเดียว มีหน้าที่ในการจัดการทุกคำขอจากผู้ใช้ผ่านทาง API (Application Programming Interfaces) เช่น ดึงข้อมูลจากฐานข้อมูล คำนวณค่าต่าง ๆ และติดต่อกับบริการอื่น ๆ

2. บริการโมเดล (Model services)

บริการสำหรับใช้คำนวณเกี่ยวกับปัญหาประติษฐ์ของทางคณะผู้จัดทำเท่านั้น เนื่องจากกินทรัพยากรสูง การแยกบริการออกมาจากบริการหลักจึงสามารถดูแลและจัดการได้ง่ายกว่า โดยจะรับค่ามาจากส่วนบริการหลักและส่งค่ากลับให้

3.4 Sequence Diagram

3.4.1 Register

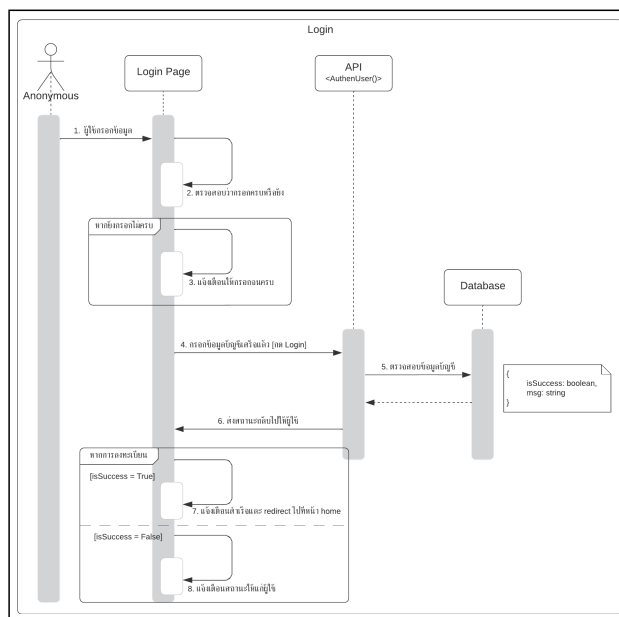


รูปที่ 3.4 Register Sequence Diagram

สถานการณ์: ผู้ใช้สมัครสมาชิก

1. ผู้ใช้กรอกข้อมูลในหน้าสมัครสมาชิก
2. user interface จะทำการรับมือในกรณีที่ผู้ใช้กรอกข้อมูลยังไม่ถูกต้อง
3. เมื่อข้อมูลถูกต้องแล้ว ผู้ใช้ทำการกดยืนยัน และส่งคำขอไปที่บริการส่วนหลัก
4. บริการส่วนหลักทำการตรวจสอบข้อมูลผู้ใช้เดิม หากยังไม่มีข้อมูลจะบันทึกข้อมูลใหม่ลงฐานข้อมูลและส่งสถานะกลับสู่ผู้ใช้ หากมีอยู่แล้วจะทำการส่งสถานะกลับสู่ผู้ใช้ทันที
5. เมื่อผู้ใช้ได้รับสถานะ หากสำเร็จที่ส่งผู้ใช้ไปที่หน้า login หากไม่สำเร็จจะแจ้งเตือนให้แก่ผู้ใช้

3.4.2 Login

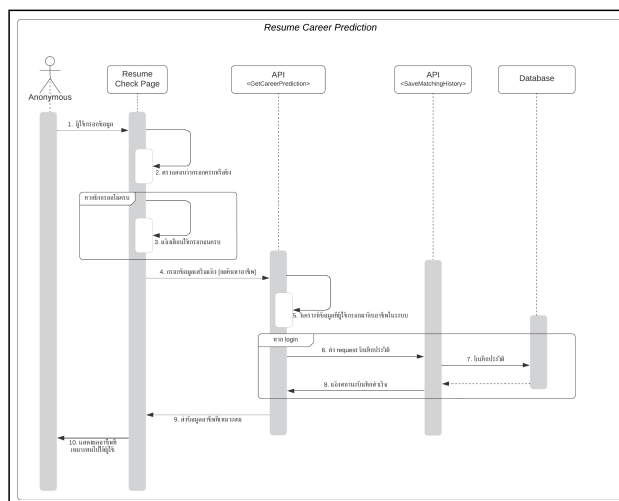


รูปที่ 3.5 Login Sequence Diagram

สถานการณ์: ผู้ใช้ล็อกอิน

1. ผู้ใช้กรอกข้อมูลในหน้าล็อกอิน
2. user interface จะทำการรับมือในกรณีที่ผู้ใช้กรอกข้อมูลยังไม่ถูกต้อง
3. เมื่อข้อมูลถูกต้องแล้ว ผู้ใช้ทำการกดยืนยัน และส่งค่าไปที่บริการส่วนหลัก
4. บริการส่วนหลักตรวจสอบข้อมูลผู้ใช้ที่มีอยู่ และส่งสถานะกลับสู่ผู้ใช้
5. เมื่อผู้ใช้ได้รับสถานะ หากสำเร็จที่ส่งผู้ใช้ไปที่หน้าหลัก หากไม่สำเร็จจะแจ้งเตือนให้แก่ผู้ใช้

3.4.3 Resume Prediction

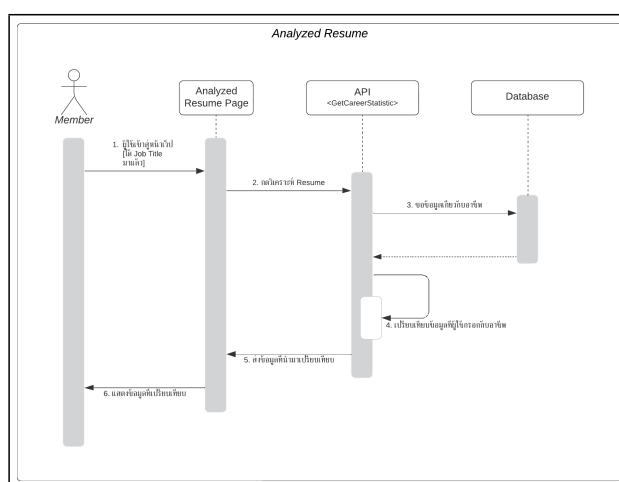


รูปที่ 3.6 Resume Prediction Sequence Diagram

สถานการณ์: ผู้ใช้ใช้งานระบบวิเคราะห์เรซูเม่

1. ผู้ใช้กรอกข้อมูลเรซูเม่ของตัวเอง
2. user interface รับมือกรณีที่ผู้ใช้กรอกข้อมูลไม่ครบ และแจ้งเตือนแก่ผู้ใช้
3. ผู้ใช้กรอกข้อมูลถูกต้องและส่งคำขอสู่บริการหลัก
4. บริการหลักจัดการวิเคราะห์ข้อมูลเรซูเม่ของผู้ใช้ ในกรณีที่ผู้ใช้ login จะทำการบันทึกค่ารับเข้าต่าง ๆ ลงสู่ฐานข้อมูลเพื่อสำรองข้อมูลให้ผู้ใช้
5. ส่งข้อมูลกลับสู่ผู้ใช้และแสดงผล

3.4.4 Analyze Resume

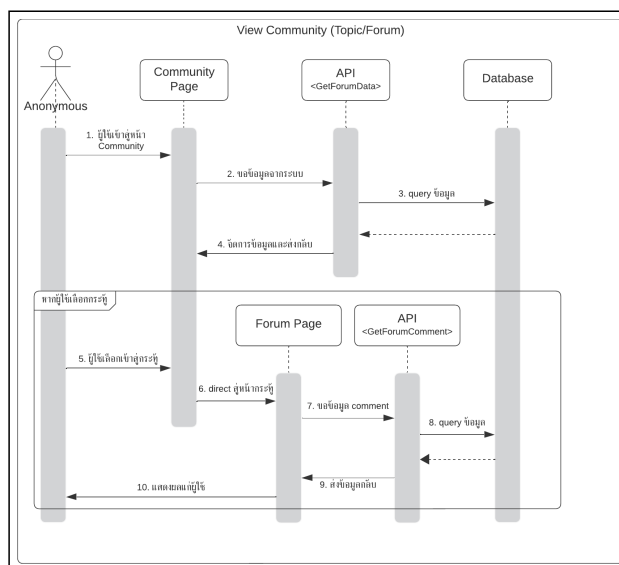


รูปที่ 3.7 Analyze Resume Sequence Diagram

สถานการณ์: ผู้ใช้เข้าสู่ข้อมูลเชิงลึกจากการวิเคราะห์เรซูเม่

1. ผู้ใช้เข้าสู่หน้าวิเคราะห์ข้อมูลเชิงลึกจากช่องทางใดทางหนึ่ง เช่น ผ่านการ redirect จากหน้าวิเคราะห์เรซูเม่ หรือเข้าสู่หน้าข้อมูลเชิงลึกโดยตรง (ต้องมีข้อมูลมาก่อน)
2. user interface ส่งคำขอสู่ระบบบริการหลัก และดึงข้อมูลจากฐานข้อมูล
3. บริการหลักคำนวณข้อมูลและส่งกลับสู่ผู้ใช้
4. แสดงผลแก่ผู้ใช้

3.4.5 View Forum

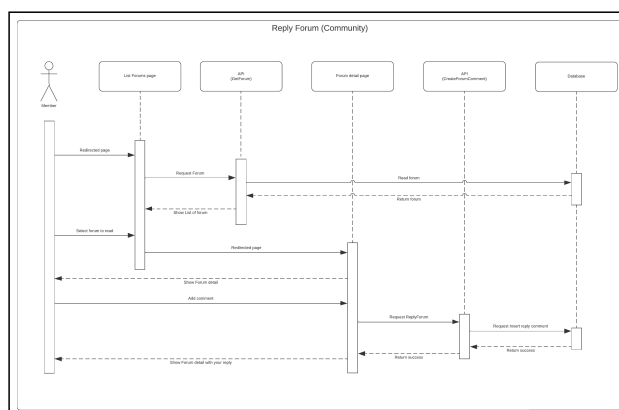


รูปที่ 3.8 View Forum Sequence Diagram

สถานการณ์: ผู้ใช้เข้าดูหน้าชุมชน

1. ผู้ใช้เข้ากดเข้าสู่หน้าชุมชน
2. user interface ส่งคำขอสู่ระบบบริการหลักและดึงข้อมูลจากฐานข้อมูล
3. บริการหลักส่งข้อมูลกลับสู่ผู้ใช้และแสดงผล

3.4.6 Reply Forum



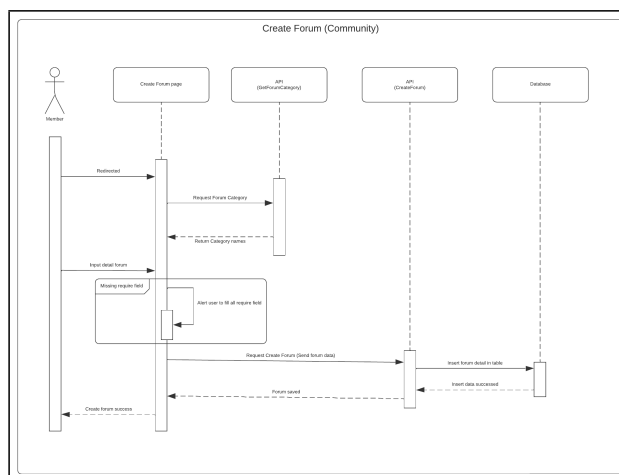
รูปที่ 3.9 Reply Forum Sequence Diagram

สถานการณ์: ผู้ใช้ตอบกลับข้อความในกระทู้ชุมชน

1. ผู้ใช้กดเข้าหน้าเลือกกระทู้จากชุมชน
2. user interface ส่งคำขอสู่บริการหลัก เพื่อดึงข้อมูลจากฐานข้อมูล
3. ส่งรายการกระทู้กลับสู่ผู้ใช้

4. ผู้ใช้เลือกกระทู้และเข้าสู่หน้ากระทู้
5. ผู้ใช้กดพิมพ์ข้อความและกดตอบกลับกระทู้
6. user interface ทำการส่งข้อมูลสู่บริการหลักเพื่อบันทึกข้อความใหม่นี้สู่ฐานข้อมูล
7. ส่งสถานะสำเร็จแก่ผู้ใช้และแสดงผล

3.4.7 Create Forum

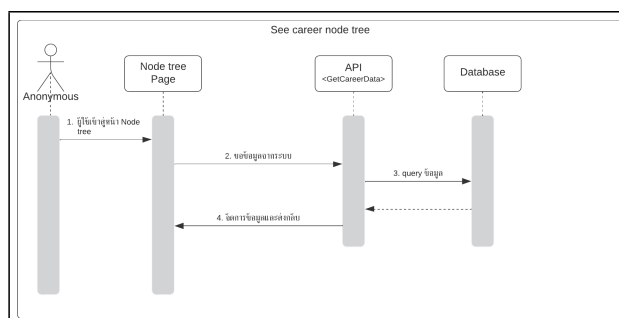


รูปที่ 3.10 Create Forum Sequence Diagram

สถานการณ์: ผู้ใช้สร้างกระทู้ใหม่

1. ผู้ใช้กดสร้างกระทู้ใหม่
2. user interface ส่งคำขอร้องจากบริการหลักเพื่อดึงข้อมูลประเภทกระทู้ที่อาจมีได้ และส่งกลับสู่ผู้ใช้
3. ผู้ใช้เลือกประเภทของกระทู้และกรอกข้อมูลกระทู้ทั้งหมด
4. user interface ตรวจสอบข้อมูลที่ผู้ใช้กรอก หากผิดพลาดจะแจ้งเตือนแก่ผู้ใช้ หากไม่มีปัญหาจะส่งคำขอสู่บริการหลัก
5. บริการหลักบันทึกข้อมูลกระทู้ลงสู่ฐานข้อมูล และส่งสถานะกลับสู่ผู้ใช้ พร้อมแสดงผล

3.4.8 See Career Node Tree



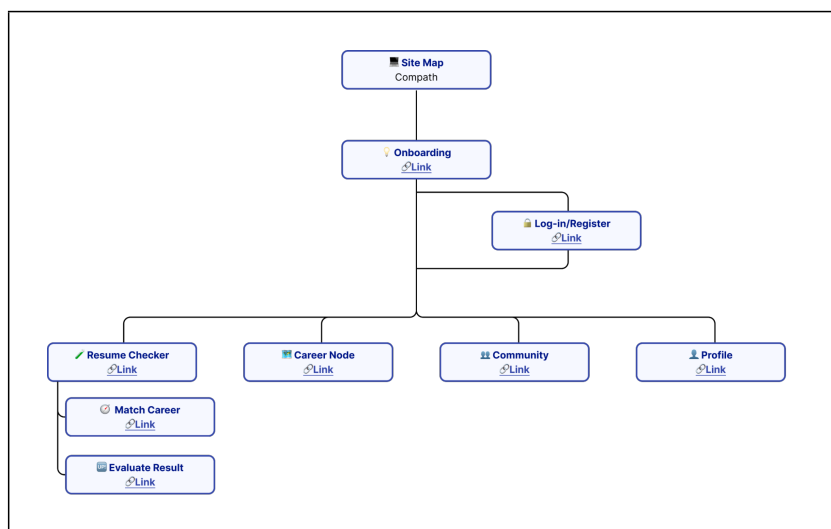
รูปที่ 3.11 See Career Node Tree Sequence Diagram

สถานการณ์: ผู้ใช้เข้าดูหน้าสายใยอาชีพ

1. ผู้ใช้เข้าสู่หน้าสไลด์โฮมาซีฟ
2. user interface ขอข้อมูลจากบริการหลักและดึงข้อมูลจากฐานข้อมูล
3. บริการหลักส่งข้อมูลกลับสู่ผู้ใช้และแสดงผล

3.5 แผนผังเว็บไซต์ (Site Map)

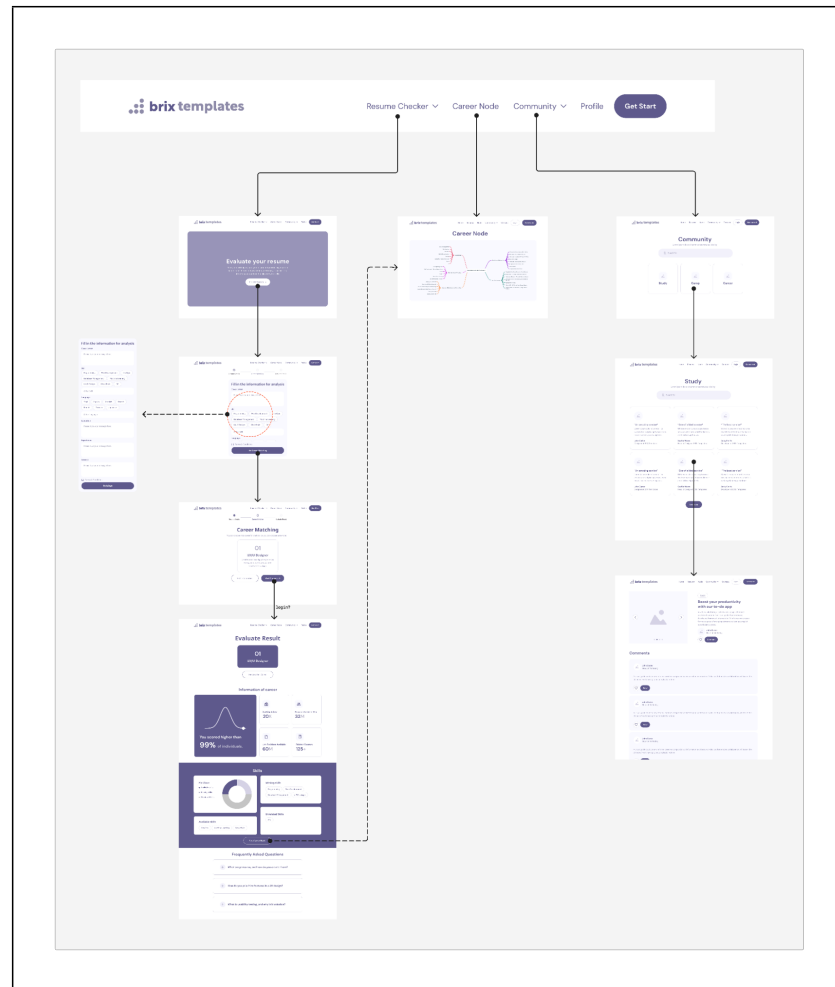
แผนผังเว็บไซต์(Site Map) เป็นโครงสร้างแบบภาพรวมที่แสดงถึงโครงสร้างของเว็บไซต์ทั้งหมด ซึ่งมีวัตถุประสงค์หลักคือการใช้ให้ผู้ใช้และเครื่องมือการค้นหาเข้าใจโครงสร้างและลำดับของเนื้อหาบนเว็บไซต์ได้ง่ายขึ้น แผนผังเว็บไซต์ช่วยในการนำทางผู้ใช้ไปสู่ส่วนต่าง ๆ ของเว็บไซต์, ลดความสับสนและเพิ่มประสิทธิภาพในการค้นหาข้อมูล



รูปที่ 3.12 SiteMap

3.6 แผนที่การนำทางเว็บไซต์ (Navigation Map)

แผนที่การนำทางเว็บไซต์ (Navigation Map) เป็นเครื่องมือที่ใช้ในการอธิบายหรือแสดงภาพรวมของโครงสร้างของเว็บไซต์ที่มีหลายหน้า โดยจุดประสงค์หลักของ Navigation Map คือการช่วยผู้ใช้ในการเข้าใจโครงสร้างและลำดับขั้นตอนในการนำทางภายในระบบ



รูปที่ 3.13 Navigation Map

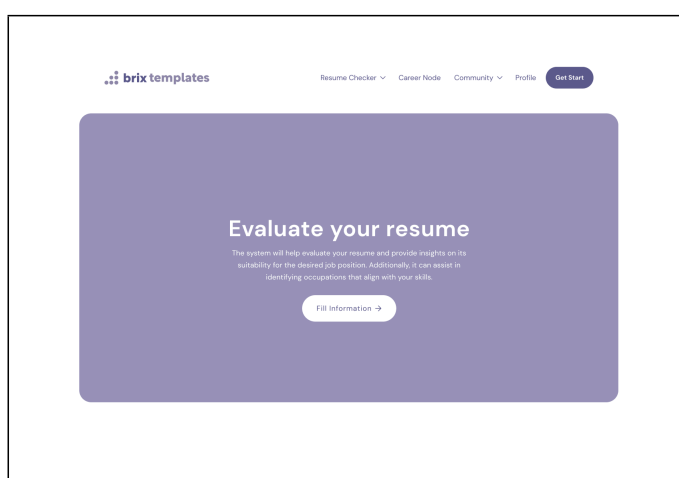
3.7 Wireframe

การสร้าง wireframe เป็นขั้นตอนสำคัญในกระบวนการออกแบบเว็บหรือแอปพลิเคชัน เนื่องจากช่วยให้ทีมออกแบบและผู้เกี่ยวข้องเข้าใจโครงสร้างพื้นฐานของหน้าจอและการจัดวางองค์ประกอบหลัก ๆ โดยไม่จำเป็นต้องใส่รายละเอียดกราฟิกหรือสี นอกจากนี้ยังช่วยให้เห็นถึงภาพรวมประสบการณ์ผู้ใช้ ทำให้สามารถทดสอบและปรับปรุงก่อนที่จะเริ่มการพัฒนา และทำให้กระบวนการออกแบบเป็นไปอย่างมีประสิทธิภาพและรวดเร็วขึ้น โดยเราเลือกที่จะสร้าง wireframe ในส่วนที่เป็นฟีเจอร์สำคัญเพื่อที่จะสามารถนำไปทดสอบกับผู้ใช้ในกระบวนการคิดเชิงออกแบบ (Design Thinking)

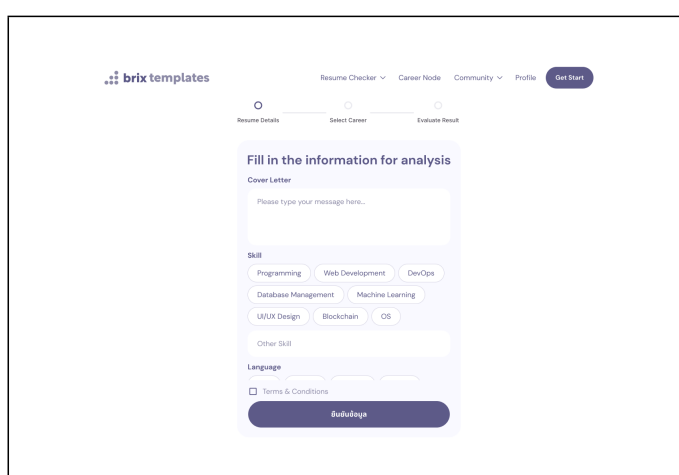
3.7.1 Resume Checker

Resume Checker เป็นฟีเจอร์ที่ให้ผู้กรอกข้อมูลของเพื่อนำไปวิเคราะห์อาชีพที่เหมาะสม โดยผู้ใช้งานกรอกข้อมูลเกี่ยวกับประสบการณ์ทำงาน, การศึกษา, ทักษะ, และคุณสมบัติอื่น ๆ ที่สำคัญ

- หน้าหลัก



รูปที่ 3.14 หน้าหลักสำหรับกดเริ่มวิเคราะห์ Resume



รูปที่ 3.15 หน้ากรอกข้อมูลของเพื่อนำไปวิเคราะห์

- ส่วนประกอบแบบโต้ตอบ

Fill in the information for analysis

Cover Letter

Please type your message here...

Skill

Programming
 Web Development
 DevOps
 Database Management
 Machine Learning
 UI/UX Design
 Blockchain
 OS

Other Skill

Language

Thai
 English
 Spanish
 French
 French
 German
 Japanese

Other Language

Education

Please type your message here...

Experience

Please type your message here...

Interest

Please type your message here...

☐ Terms & Conditions

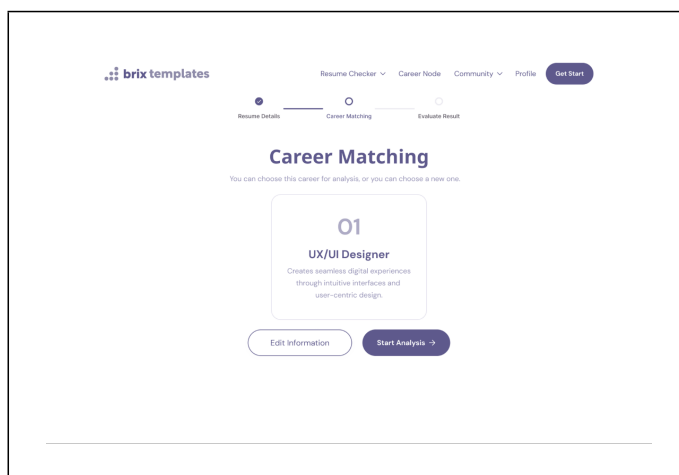
ยืนยันข้อมูล

รูปที่ 3.16 ช่องสำหรับกรอกข้อมูลของเพื่อนำไปวิเคราะห์

3.7.2 Match Career

Match Career เป็นฟีเจอร์สำหรับแสดงอาชีพที่เหมาะสมกับข้อมูลที่ใช้กรอกมาทำให้ผู้ใช้สามารถรู้ถึงอาชีพที่เหมาะสมกับตนเอง และข้อมูลคร่าว ๆ ของอาชีพนั้น โดยผู้ใช้สามารถกลับไปแก้ไขข้อมูลที่กรอกได้เพื่อนำมาวิเคราะห์อาชีพใหม่ รวมถึงสามารถดูวิเคราะห์ข้อมูลเชิงลึกต่อไปได้แต่ผู้ใช้ต้องสมัครและล็อกอินเพื่อเก็บข้อมูลก่อน

- หน้าหลัก



รูปที่ 3.17 หน้าแสดงอาชีพที่วิเคราะห์ได้

3.7.3 Evaluate Result

Evaluate Result เป็นฟีเจอร์สำหรับแสดงข้อมูลเชิงลึกของอาชีพที่วิเคราะห์ได้ในตอนแรก โดยจะแสดงข้อมูลที่มีโยชน์เพื่อให้ผู้ใช้สามารถนำไปพัฒนาทักษะของตนเองและเรซูเม่ต่อไปได้

- หน้าหลัก

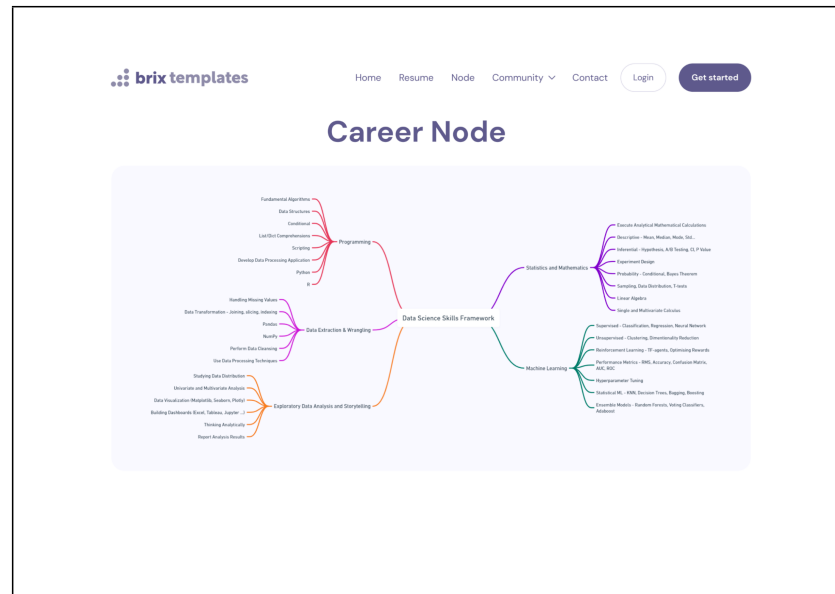


รูปที่ 3.18 หน้าแสดงข้อมูลเชิงลึกของอาชีพ

3.7.4 Career Node

Career Node เป็นฟีเจอร์ที่ช่วยให้ผู้ใช้ได้เห็นภาพชัดเจนมากขึ้นเกี่ยวกับทักษะที่ต้องใช้ในอาชีพต่าง ๆ โดยจะนำเสนอข้อมูลนี้ในรูปแบบของโหนดหรือโครงสร้างเครือข่ายซึ่งช่วยให้ข้อมูลสามารถสืบค้นและเข้าใจได้ง่ายและรวดเร็ว

- หน้าหลัก

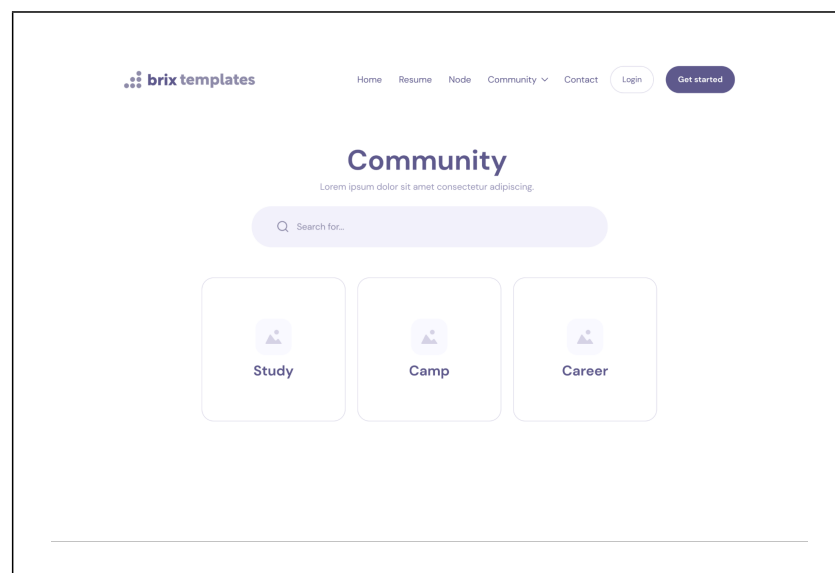


รูปที่ 3.19 หน้าแสดงทักษะที่ใช้ในสายอาชีพ

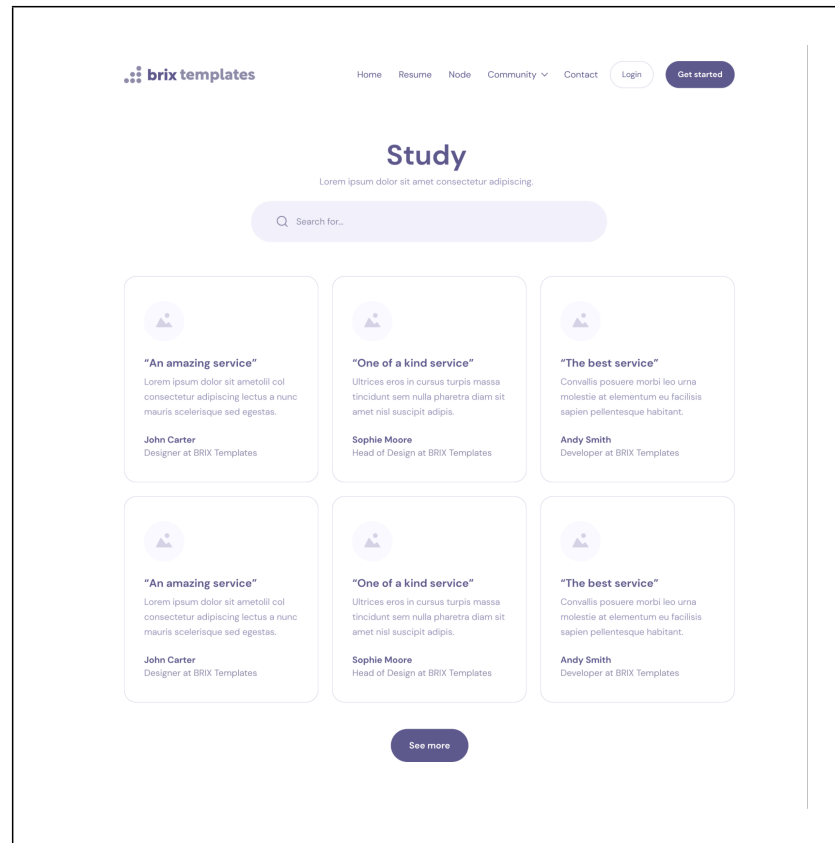
3.7.5 Community

Community เป็นฟีเจอร์ที่เสริมสร้างพื้นที่สำหรับการแลกเปลี่ยนข้อมูลและความคิดเห็นระหว่างสมาชิกในระบบ โดยผู้ใช้สามารถเข้ามาสร้างและตั้งกระทู้เพื่อนำเสนอความรู้, ประสบการณ์, หรือสนทนาเกี่ยวกับหัวข้อที่น่าสนใจ

- หน้าหลัก



รูปที่ 3.20 หน้าเลือกหมวดหมู่ของกระทู้



รูปที่ 3.21 หน้าแสดงรายการกระทู้

รูปที่ 3.22 หน้าแสดงรายละเอียดของกระทู้

3.8 ขั้นตอนการพัฒนาโมเดล

3.8.1 การรวบรวมข้อมูล (Get Data)

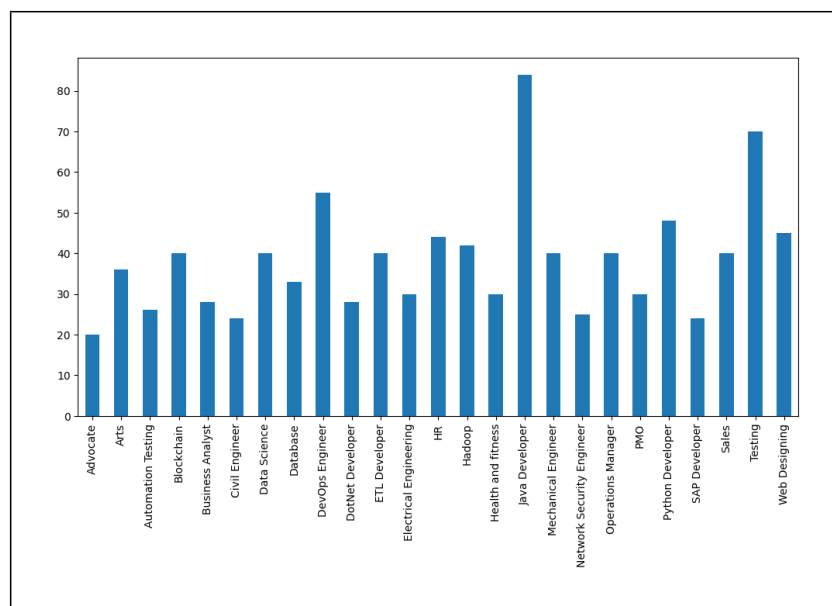
ทางคณะผู้จัดทำได้เลือกนำชุดข้อมูล Updated Resume Dataset [?] จาก Kaggle ซึ่งเป็นชุดข้อมูล ที่ประกอบไปด้วยอาชีพทั้งหมด 25 หมวด และรายละเอียดภายในเรซูเม่ของแต่ละอันก็จะแตกต่างกันไปในแต่ละอัน ซึ่งมีเรซูเม่ ทั้งหมด 962 เล่ม ซึ่งทางคณะผู้จัดทำ จำเป็นต้องทำการเตรียมข้อมูลที่เหมาะสมเพื่อให้โมเดลมีประสิทธิภาพ

3.8.2 การเตรียมข้อมูล (Clean, Prepare and Manipulate Data)

หลังจากที่ทางคณะผู้จัดทำได้ทำการสำรวจชุดข้อมูล ทำให้พบว่ารายละเอียดของเรซูเม่ยังไม่สามารถนำไปใช้ได้ทันที เนื่องจากข้อมูลเรซูเม่ ได้มีตัวอักษรพิเศษรวมอยู่เป็นจำนวนมาก เช่น à, □, Ç, √, ∩ รวมไปถึงมีการใส่ลิงก์ภายนอกอีกด้วย ทางคณะผู้จัดทำจึงจำเป็นต้องทำให้ตัวหนังสือภายในเรซูเม่ เป็นเพียงตัวอักษรปกติเท่านั้น

เนื่องจากทางคณะผู้จัดทำเล็งเห็นกลุ่มเป้าหมายเป็นนักศึกษาวิศวกรรมคอมพิวเตอร์ มจร. จึงจำเป็นอย่างยิ่งที่ต้องเลือกเฉพาะข้อมูลที่เป็น Software Engineer, Designer, Data, Security, Cloud Management ซึ่งเราจำเป็นอย่างยิ่งที่จะต้อง ครอบคลุมอาชีพ และทำการรวมบางอาชีพเข้าด้วยกัน

และเพื่อให้โมเดลสามารถที่จะเข้าใจภาษาของมนุษย์ได้ คณะผู้จัดทำจึงได้ใช้อัลกอริทึมในการแปลงภาษา Term Frequency Inverse Document Frequency (TF-IDF)



รูปที่ 3.23 จำนวนของเรซูเม่ในแต่ละหมวดอาชีพ

Category	Resume
0 Data Science	Skills * Programming Languages: Python (pandas...
1 Data Science	Education Details \nMay 2013 to May 2017 B.E...
2 Data Science	Areas of Interest Deep Learning, Control Syste...
3 Data Science	Skills á¤ R á¤ Python á¤ SAP HANA á¤ Table...
4 Data Science	Education Details \n MCA YMCAUST, Faridab...

รูปที่ 3.24 ลักษณะข้อมูลของชุดข้อมูล

3.8.3 การเทรนโมเดล (Train Model)

หลังจากที่ทางคณะผู้จัดทำได้ทำการเตรียมข้อมูลสำเร็จแล้ว คณะผู้จัดทำก็ได้ทำโมเดลขึ้นมาทั้งหมด 2 โมเดลเพื่อเปรียบเทียบ ซึ่งเป็นโมเดลสำหรับการจัดหมวดหมู่ ได้แก่

1. Navie Bayes Classification
2. Long Short Term Memory (LSTM)
 - Embedding layer | max input dimension is 10,000 and embedding dimension size is 100
 - Bidirectional LSTM 128 units | dropout rate 40%
 - Bidirectional LSTM 64 units | dropout rate 40%
 - Dense layer 25 units | softmax as activation function

และคณะผู้จัดทำก็ทำการใส่ข้อมูลที่เตรียมเอาไว้เพื่อทำการเทรนโมเดล โดยที่โมเดลจะทำการทำนายว่าข้อมูลเรซูเม่ที่ใส่เข้าไป เป็นอาชีพใดจากทั้งหมด 5 อาชีพ (Software Engineer, Designer, Data, Security, Cloud Management) ซึ่ง สามารถแสดงผลออกมาเป็นความน่าจะเป็นของแต่ละสายอาชีพได้

3.8.4 การทดสอบข้อมูล (Test Data)

หลังจากที่คณะผู้จัดทำได้เทรนโมเดลจนสำเร็จแล้ว ก็ได้ลองนำเรซูเม่ของคณะผู้จัดทำมาลองกับโมเดล ซึ่งผลลัพธ์ที่ได้ก็ค่อนข้างน่าพอใจ คาดว่าเนื่องจากมีสายอาชีพที่ต้องทำนายค่อนข้างน้อย ทำให้มีโอกาสที่จะทำนายได้ถูกต้องค่อนข้างสูง

3.8.5 การพัฒนาโมเดล (Improve)

ทางคณะผู้จัดทำคาดหวังว่าจะพัฒนาโมเดลได้ด้วยการหาชุดข้อมูลสำหรับการเทรนมากขึ้นกว่านี้ และเพิ่มจำนวน Epoch สำหรับการเทรน Long Short Term Memory (LSTM) ที่มากกว่านี้ และคาดหวังว่าในอนาคตจะมีสายอาชีพที่ทำการทำนายเพิ่มมากขึ้นกว่านี้

บทที่ 4 ผลการดำเนินงาน

จัดทำในอนาคต

บทที่ 5 บทสรุป

จัดทำในอนาคต