



ชื่อโครงการ

Predict Tomorrow

เสนอ

อาจารย์ สนิท แสงเหลา

จัดทำโดย

นาย	จิรภัทร สุวรรณลัมัย	6487019
นางสาว	พรรพษา อารีวงศ์สถิตย์	6487049
นางสาว	รวีพิชญ์ เล้าแสงชัยวัฒน์	6487056

ITDS283 Mobile Application

คณะเทคโนโลยีสารสนเทศและการสื่อสาร สาขาวิทยาการและ
เทคโนโลยีดิจิทัล

ชั้นปี ที่ 2 ภาคเรียนที่ 2 /2565

กิตติกรรมประกาศ

โครงการ “ Predict Tomorrow” สามารถดำเนินการจนสำเร็จลุล่วงไปด้วยดี เนื่องจากได้รับความอนุเคราะห์จากบุคคลที่มีส่วนเกี่ยวข้องที่ช่วยให้เราเสร็จสิ้นโครงการนี้ ขอขอบคุณครอบครัวผู้จัดทำโครงการ การสนับสนุนจากเพื่อนร่วมภาควิชา ที่ให้ความร่วมมือ คำแนะนำ ความรู้ คำปรึกษา ตลอดจน แนวทางแก้ไขข้อผิดพลาดของโครงการ ซึ่งช่วยให้โครงการนี้สำเร็จได้ด้วยดี ขอขอบคุณมา ณ ที่นี้

จิรภัทร สุวรรณลัมัย

พรวรรษา อาริยวงศ์สถิตย์

รวีพิชญ์ เล้าแสงชัยวัฒน์

Predict Tomorrow

นาย	จิรภัทร สุวรรณลัมย์	6487019	DST/B
นางสาว	พรวพษา อารีย์วงศ์สถิตย์	6487049	DST/B
นางสาว	รวีพิชญ์ เล้าแสงชัยวัฒน์	6487056	DST/B

B.Sc.(Digital Science and Technology)

ที่ปรึกษาโครงการ : อาจารย์ สนิท แสงเหลา

บทคัดย่อ

โครงการชิ้นนี้เป็นส่วนหนึ่งของวิชา Mobile Application เป้าหมายของความคิดริเริ่มนี้คือเพื่อให้ผู้ที่ใช้งาน Application ของเราสามารถทำนายสภาพภูมิอากาศและสามารถดูการแต่งตัวที่เหมาะสมรวมถึงสิ่งของที่เป็นในแต่ละสภาพภูมิอากาศ

หลังจากการสนทนากันในทีม เราได้ข้อสรุปว่าหากผู้ใช้สามารถที่จะดูสภาพอากาศล่วงหน้าได้ 7 วัน รวมไปถึงสามารถที่จะดูการแต่งตัวและสิ่งของจำเป็นที่สอดคล้องกับสภาพภูมิอากาศแล้วนั้น จะช่วยให้ชีวิตประจำวันเป็นไปอย่างสะดวกสบายมากยิ่งขึ้น ดังนั้นโครงการนี้จะครอบคลุมการทำนายสภาพภูมิอากาศล่วงหน้า 7 วัน การแต่งตัว และสิ่งของที่จำเป็น ผ่านแอปพลิเคชันของเรา

สารบัญ

กิจกรรมประกาศ	ii
บทคัดย่อ	iii
1 บทนำ	6
1.1 แรงจูงใจ	6
1.2 คำชี้แจงปัญหา	7
1.3 วัตถุประสงค์ของโครงการ	7
1.4 ขอบเขตโครงการ	7
1.5 ประโยชน์ที่คาดว่าจะได้รับ	7
1.6 Organization of the document	8
2 ภูมิหลัง	9
2.1 literature review	9
3 การวิเคราะห์และออกแบบ	10
3.1 System Architecture Overview	10
3.2 System Structure Chart	11
3.3 กระบวนการวิเคราะห์และออกแบบ	12
3.3.1 Data Flow Diagram	12
3.3.2 Data Dictionary	14
3.4 ฐานข้อมูลและการออกแบบ	21
3.4.1 ER-Diagrams	21
3.4.2 Relation -Schema	22
3.4.3 โครงสร้างของไฟล์	24
3.5 การออกแบบ I/O	27
3.5.1 การออกแบบ interface	27

3.5.2 Transaction Diagram	28
4 Implementation	29
4.1 ฮาร์ดแวร์และ system environment	29
4.2 คู่มือการนำไปใช้งานและเทคนิค	35
4.2.1 คู่มือการเชื่อมต่อ	35
4.2.2 เทคนิคต่างๆที่ใช้	35
5 ปัญหาความท้าทายและผลสรุปการแก้ปัญหา	36
6 Testing และการวัดผล	37
6.1 Unit Tests	37
6.1.1 Test Performed On No 1: Register	37
6.1.2 Test Performed On No 2: Login	38
6.1.3 Test Performed On No3. Weather	39
6.2 System Integration test	39
6.2.1 Test Scenario	39
7 บทสรุป	52
7.1 ประโยชน์	52
7.1.1 ประโยชน์ต่อผู้พัฒนาโครงการ	52
7.1.2 ประโยชน์ต่อผู้ใช้งาน	52
7.2 ปัญหาและข้อจำกัด	53
7.3 สิ่งที่ต้องทำต่อไป	53
8 สรุปความก้าวหน้าของโครงการ	54
9 อ้างอิง	55
10 BIOGRAPHIES	56
11 LINK OF GITHUB OF THE PROJECT	57

Introduction (แนะนำโครงการ)

เพื่อแนะนำให้ผู้อ่าน รู้แรงจูงใจ ปัญหา วัตถุประสงค์ของโครงการ ขอบเขตของโครงการ ประโยชน์ที่คาดว่าจะได้รับ และการจัดระเบียบของเอกสาร

กล่าวคือเพื่อให้ผู้อ่านได้เข้าใจมากขึ้นเกี่ยวกับสิ่งที่เราทำ เป้าหมายของโครงการเรา ประโยชน์ ขนาดของงาน แผนการในโครงการ และข้อมูลภาพรวมของรายงาน

Motivation (แรงจูงใจ)

การเปลี่ยนแปลงทางสภาพอากาศนั้นเป็นปรากฏการณ์ทางธรรมชาติที่สามารถเปลี่ยนแปลงได้ตลอดเวลา ไม่สามารถคาดเดาได้ว่าวันนี้หรือพรุ่งนี้จะเกิดอะไรขึ้น เพราะฉะนั้นการที่เราสามารถคาดคะเนแต่ละวันของเราได้นั้นคงเป็นสิ่งที่ดี และเพื่อทำให้เราสามารถรับมือกับสภาพอากาศในแต่ละวันได้จึงต้องมีการพยากรณ์อากาศขึ้น

และเนื่องจากปัจจุบันมีการพยากรณ์สภาพอากาศได้อย่างแม่นยำมากกว่าในอดีต ทำให้ในแต่ละวันของเราการดูพยากรณ์อากาศก็ถือเป็นสิ่งสำคัญเพื่อเตรียมตัวรับมือกับสภาพอากาศต่างๆ เช่น ร้อน, หนาว, ฝนตก, ลมแรง เป็นต้น

เป็นผลให้เรามีความสนใจที่จะทำแอปพลิเคชันพยากรณ์อากาศ โดยสามารถให้คำแนะนำเพิ่มเติมได้ อาทิเช่น วันนี้มีเปอร์เซ็นต์ฝนตกค่อนข้างสูง ผู้ใช้ควรพกร่มก่อนออกจากบ้าน เป็นต้น และเพื่อเป็นเครื่องมือที่ช่วยให้ชีวิตประจำวันของผู้ใช้ดียิ่งขึ้น

Problem Statement (คำชี้แจงปัญหา)

1. ผู้คนส่วนใหญ่ไม่ค่อยเช็คพยากรณ์อากาศ
2. ผู้คนขาดความพร้อม และหลงลืมการเตรียมตัวในกรณีต่างๆ

Objectives of the project (วัตถุประสงค์ของโครงการ)

- เพื่อให้ผู้ใช้สามารถเตรียมตัวรับมือกับสภาพอากาศในแต่ละวันได้
- เพื่อให้คำแนะนำด้านต่างๆเพิ่มเติมในแต่ละวัน
- เพื่อให้ผู้ใช้สามารถเข้าถึงแอปและรับรู้การพยากรณ์ได้อย่างง่ายและมีความสะดวก

Scope of the project (ขอบเขตโครงการ)

พยากรณ์สภาพอากาศ คำแนะนำต่างๆ เช่น วันนี้มีเปอร์เซ็นต์ที่ฝนจะตก 80% คำแนะนำควรพกร่มหรือเสื้อกันฝน โดยสังเขป และสามารถให้ผู้ใช้เตรียมรับมือกับสถานการณ์ต่างๆได้

Expected benefits (ประโยชน์ที่คาดว่าจะได้รับ)

ผู้ใช้สนใจในการดูพยากรณ์อากาศและสามารถเตรียมตัวรับมือกับสถานการณ์ต่างๆในชีวิตประจำวันได้

Organization of the document

เอกสารนี้ประกอบด้วย 6 บท ได้แก่:

1. บทนำ – บทนำเป็นบทแรกในโครงงานนี้ที่ รวมถึงแรงจูงใจ คำชี้แจงปัญหา วัตถุประสงค์ของโครงการ ขอบเขตของโครงงานและผลประโยชน์ที่คาดว่าจะได้รับ
2. ภูมิหลัง – บทที่สองเป็นภูมิหลังที่มีคำอธิบายของโครงงาน
3. การวิเคราะห์และการออกแบบ – บทที่สามคือการวิเคราะห์และการออกแบบของโครงงานมุ่งเน้นที่กระบวนการและการออกแบบระบบเช่นระบบภาพรวม System architecture ,process analysis ,and design, data flow diagram, process description, data stores, data element, database analysis and design, ER-diagram เป็นต้น
4. การนำไปใช้ – บทที่สี่นี้กล่าวถึงฮาร์ดแวร์ ระบบสภาพแวดล้อม พร้อมทั้งแสดงคู่มือและเทคนิคการใช้งาน
5. การทดสอบและประเมินผล – บทที่ห้านี้มุ่งเน้นไปที่การทดสอบและการประเมินระบบที่รวมถึง includes unit tests, system integration tests, และ test scenarios.
6. บทสรุป – บทสุดท้ายเป็นบทสรุปถึงประโยชน์ของผู้พัฒนาโครงงานและผู้ใช้ ปัญหาและข้อจำกัด และการทำงานในอนาคต

บทที่ 2 ภูมิหลัง

แอปพลิเคชันครอบคลุมเฉพาะการพยากรณ์สภาพอากาศใน 7 วัน และการบอกลักษณะการแต่งตัวรวมไปถึงสิ่งของที่จำเป็นต้องใช้ โดยบทนี้ผู้อ่านจะได้ทราบข้อมูลที่เกี่ยวข้องของโครงการที่เราค้นคว้าและศึกษามากขึ้น

2.1 Literature Review

หลังจากที่เราได้สำรวจการใช้แอปพลิเคชันเกี่ยวกับการตรวจสอบสภาพอากาศที่มีอยู่แล้วนั้น ถึงแม้ว่าแอปตรวจสอบสภาพอากาศจะมีอยู่ค่อนข้างมาก แต่ทางผู้จัดทำมีความตั้งใจอยากแสดงสิ่งของที่จำเป็นหรือแนะนำสไตล์การแต่งตัวให้กับผู้ใช้ เช่น หากฝนตก จำเป็นต้องมีร่ม หรือเสื้อกันฝนเตรียมเอาไว้ ซึ่งหลายๆคนอาจประสบกับปัญหาการลืม หรือไม่มีสิ่งของเหล่านั้นอยู่ และเพื่อเป็นการเตรียมตัวได้ท่วงทัน และรับรู้สถานการณ์ล่วงหน้าจากการพยากรณ์อากาศ เราจึงได้จัดทำ แอปพลิเคชันที่ชื่อว่า "Tomorrow" ขึ้น ซึ่งมีไอดีเดียวในการแต่งตัวและสิ่งของที่จำเป็นให้เข้ากับสภาพอากาศในแต่ละวัน เพื่อความคล่องตัวและสะดวกสบายมากยิ่งขึ้น โดยแพลตฟอร์มของเราสามารถใช้ได้ทั้งใน ระบบ IOS และ Android เพื่อความสะดวกในการใช้งาน

คณะผู้จัดทำใช้ flutter และการเขียนโปรแกรม Dart ซึ่งเป็นส่วนหนึ่งของรายวิชา ITDS 283 Mobile Application Development ตลอดทั้งการจัดทำแอปพลิเคชันนี้ขึ้น

บทที่ 3 Analysis and Design

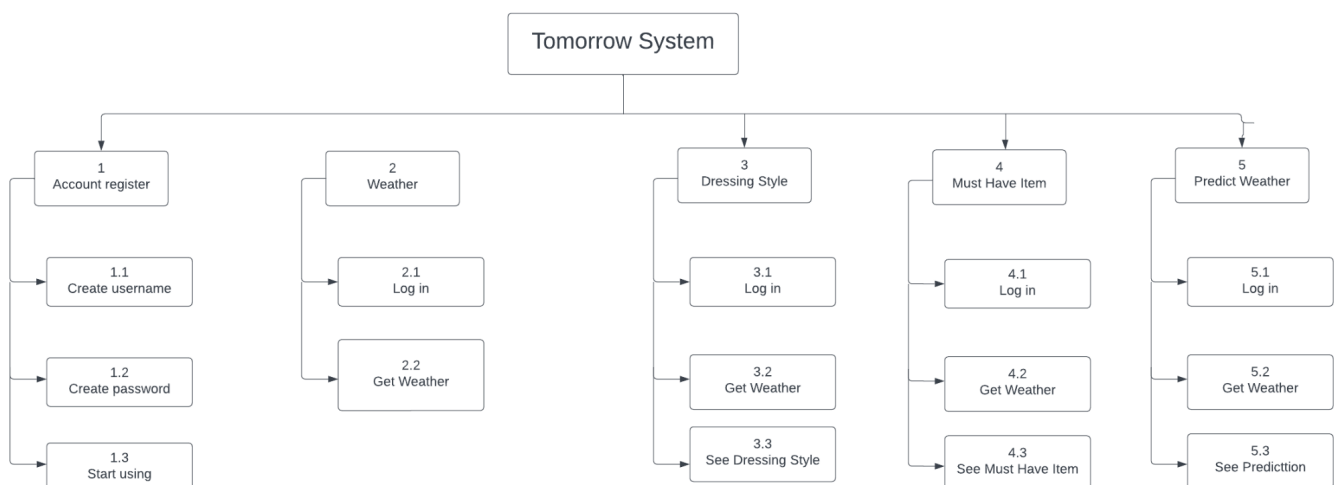
ในบทนี้เป็นการวิเคราะห์และการออกแบบแอปพลิเคชันของกลุ่มเรา ผู้ที่อ่านจะได้รู้เกี่ยวกับ System Architecture, System Structure Chart, และ I/O Design

3.1 System Architecture Overview

Tomorrow Application จะดำเนินการสร้างโดย Flutter ซึ่ง Flutter เป็นระบบชุดพัฒนาซอฟต์แวร์ UI แบบ Open Source ที่สร้างโดย Google ประกอบด้วย ส่วนสำคัญ คือ SDK (Software Development Kit) ที่มีเครื่องมือมากมาย เพื่อ compile code ของ Tomorrow Application ให้เป็น Native machine code และอีกส่วนหนึ่งคือ Framework (UI Library based on widgets) ซึ่งเป็นชุดองค์ประกอบ UI ที่สามารถใช้ซ้ำได้ นอกจากนี้เรายังใช้ภาษาโปรแกรมที่เรียกว่า Dart ซึ่งมีความชัดเจน และไวยากรณ์ไม่ซับซ้อนมาก เหมาะสำหรับ การพัฒนาแอปพลิเคชัน Android และ IOS

3.2 System Structure Chart

Tomorrow Application เป็นแอปที่มุ่งเน้นเพื่อให้ผู้ใช้สามารถเตรียมตัวล่วงหน้าได้ เนื่องจาก Tomorrow Application นั้นดึง API มาจากแหล่งพยากรณ์อากาศ ซึ่งสามารถพยากรณ์ล่วงหน้าได้ 7 วัน ทำให้ผู้ใช้สามารถเลือกชุดและเตรียมสิ่งของจำเป็นต่างๆให้เข้ากับวันนั้นๆได้



Project : การพยากรณ์สภาพอากาศ
System : Tomorrow

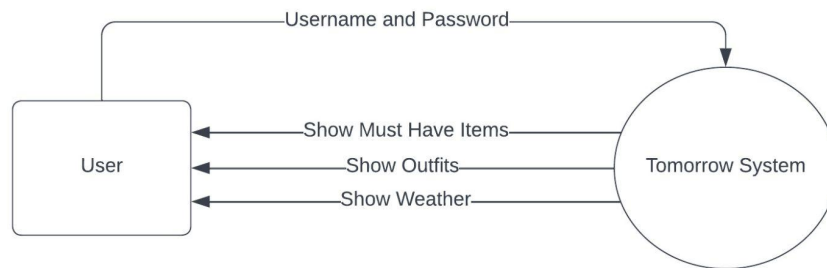
Major Advisor : อาจารย์ สนิท แสงเหลา

Description : Tomorrow System มี 5 ฟังก์ชันหลัก ได้แก่ Account register, Weather, Dressing Style, Must Have Item และ Show Predict Weather

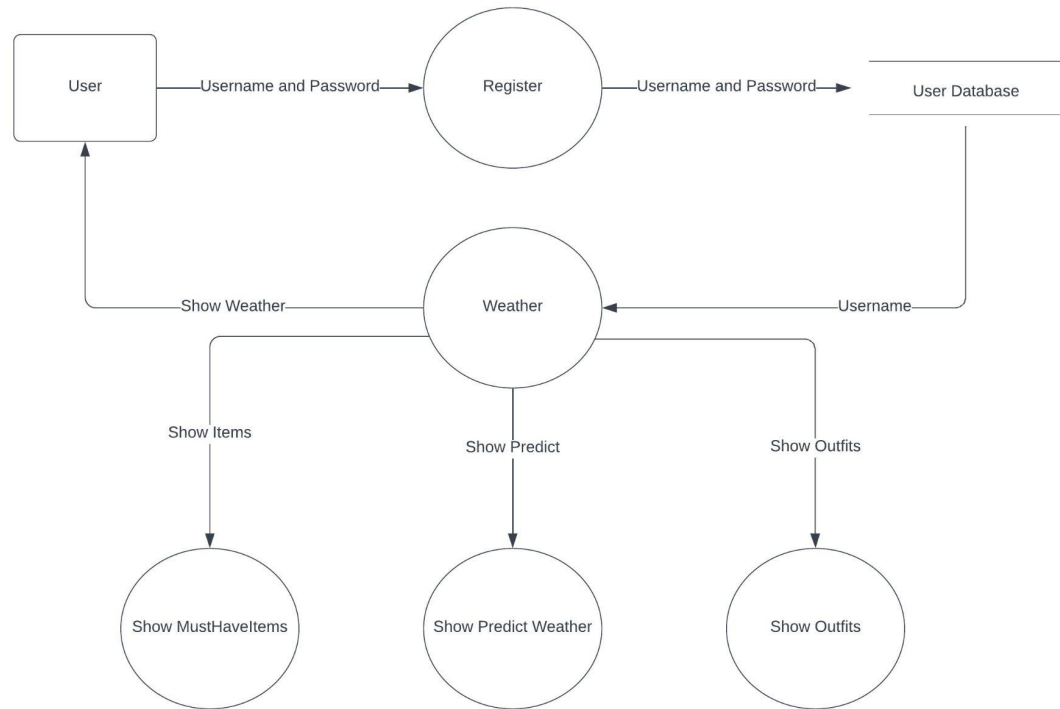
3.3 Process Analysis and Design

3.3.1 Data Flow Diagram

ในส่วนของ Data Flow Diagram ของเรา จะแสดงภาพรวมว่าข้อมูลเข้าและออก ระหว่างผู้กระทำ ซึ่งได้แก่ ผู้ใช้และผู้ดูแลระบบและระบบอุณหภูมิตomorrow



Project : การพยากรณ์สภาพอากาศ System : Tomorrow	Major Advisor : อาจารย์ สนิท แสงเหลา
Description : มี 1 actor คือ User ที่มีส่วนร่วมในระบบ Tomorrow	



Project : การพยากรณ์สภาพอากาศ
System : Tomorrow

Major Advisor : อาจารย์ สนิท แสงเหลา

Description : มี 1 actor คือ User และมี 5 process คือ Register, Weather, Show Must Have Items, Show Predict Weather, และ Show outfits

3.3.2 Data Dictionary

Data Dictionary เป็นวิธีการจัดทำเอกสารและการอธิบายกระบวนการที่เก็บข้อมูล (Data Flow) ที่เกิดขึ้นใน Data Flow Diagram(DFD) ประกอบด้วย 3 ส่วน ดังภาพด้านล่าง

- Process Descriptions
- Data Stores
- Data Elements

3.3.2.1 Process Description

ส่วนนี้จะให้คำอธิบายโดยละเอียดของแต่ละกระบวนการที่มีอยู่ในระบบนี้ ประกอบด้วย ข้อมูลขาเข้า ข้อมูลขาออก และ Logic Summary

Table1 : List of all Processes

No	Process	Name	Description
1	1	Register	Process for registration
2	1.1	Create username	Process for set username
3	1.2	Create password	Process for set password
4	1.3	Start using	Process for starting program
5	2	Weather	Process for letting user see the weather
6	2.1	Log In	Process for logging before using
7	2.2	Get Weather	Process for letting user see the weather information
8	3	Dressing Style	Process for letting users to see outfit
9	3.1	See Dressing Style	Process for letting users to see outfit
10	4	Must Have Item	Process for letting users to see must have item

11	4.1	See Must Have Item	Process for letting users to see must have item
12	5	Predict Weather	Process for letting users to see predict weather
13	5.1	See Prediction	Process for letting users to see predict weather

Table2 : Process Description of Weather

process	1-Register
คำอธิบาย	Process for registration
Inbound data	- Username/ Password
Outbound data	- Username/Password

Logic Summary

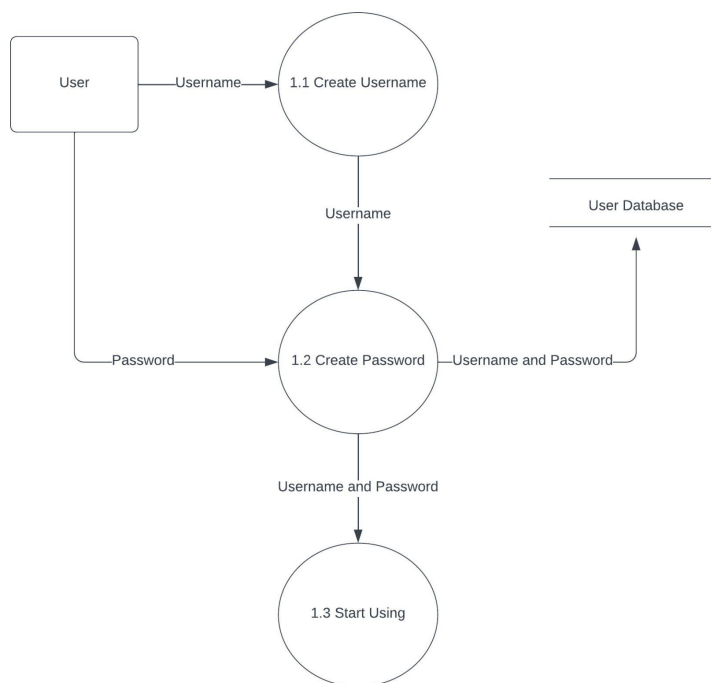


Table2 : Process Description of Weather

process	2- Weather
คำอธิบาย	Process for letting user see the weather
Inbound data	<ul style="list-style-type: none"> - Username/ Password - UserID
Outbound data	<ul style="list-style-type: none"> - Username

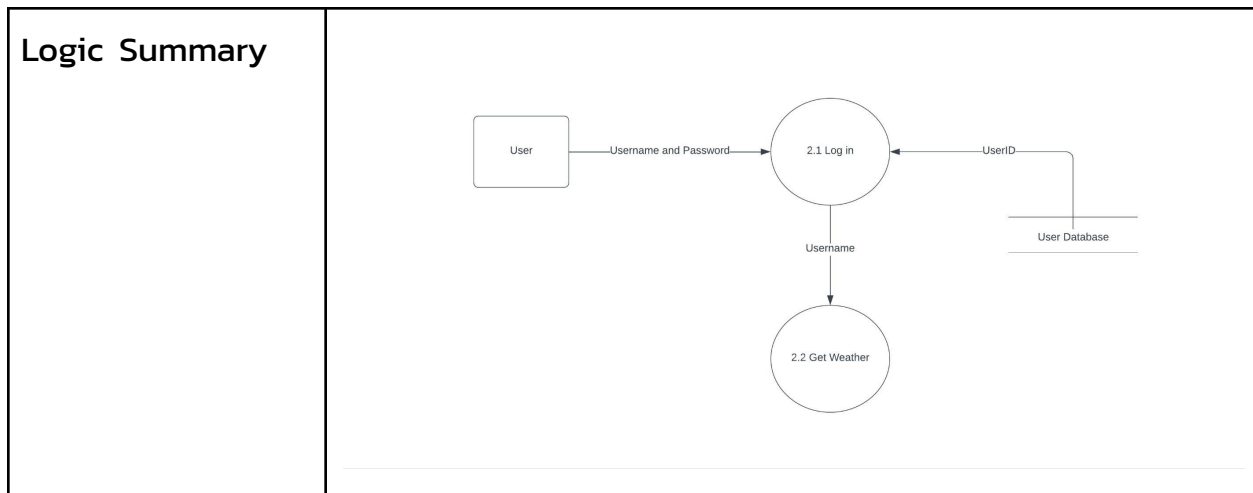


Table3 : Process Description of must have Items

process	3- See Dressing Style
คำอธิบาย	Process for letting users to see outfit
Inbound data	- Username
Outbound data	- Choose Outfits
Logic Summary	<pre> graph TD User[User] -- "Username and Password" --> 31((3.1 Log in)) UserDatabase[User Database] -- "UserID" --> 31 31 -- "Username" --> 32((3.2 Get Weather)) 32 -- "Choose Outfits" --> 33((3.3 See Dressing Style)) </pre>

Table4 : Process Description of must have Items

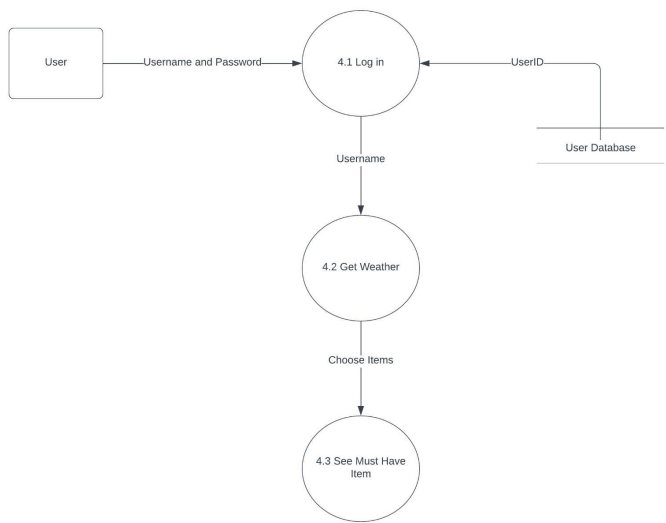
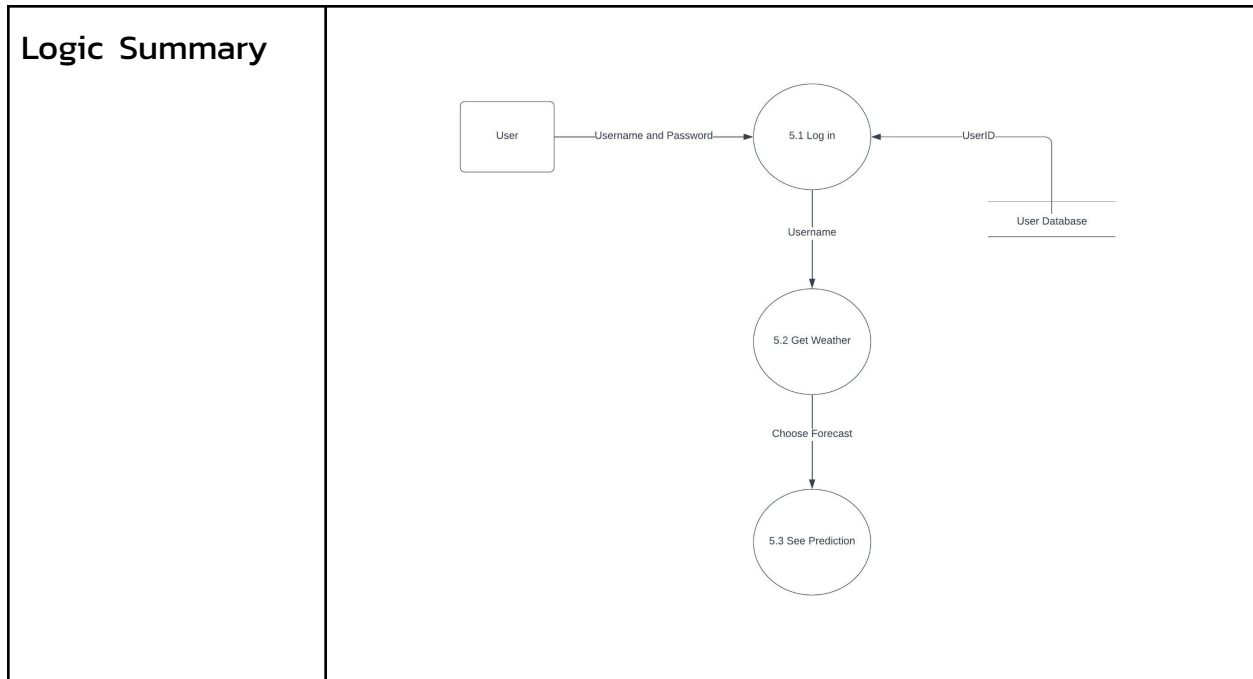
process	4- See Must Have Items
คำอธิบาย	Process for letting users to see must have items
Inbound data	- Username
Outbound data	- Choose Items
Logic Summary	 <pre> graph TD User[User] -- "Username and Password" --> 41((4.1 Log in)) 41 <--> UserID UserDB[User Database] 41 -- "Username" --> 42((4.2 Get Weather)) 42 -- "Choose Items" --> 43((4.3 See Must Have Item)) </pre>

Table5 : Process Description of Show Predict Weather

process	5- Predict Weather
คำอธิบาย	Process for letting users to see predict weather
Inbound data	- Username
Outbound data	- Choose Forecast



3.3.2.2 Data Stores

ในส่วนนี้จะเป็นการอธิบาย data stores ที่มีอยู่ใน data flow diagrams และประกอบด้วย Data Store Name, คำอธิบาย ,Inbound Data,และ Outbound Data

Table5 : List of All Data Stores

No.	Data_Store	Name	Description
1	User Database	User Database	รวบรวม username และ password ของ users

Table 7: Data Store Description of User_DB

Data Store Name	User Database–User Database
คำอธิบาย	รวบรวม Username และ Password สำหรับ User
Inbound Data	<ul style="list-style-type: none"> Username Password
Outbound Data	<ul style="list-style-type: none"> Username

3.3.2.3 Data Element

ส่วนประกอบข้อมูล หรือ data flow ที่มีอยู่ได้อธิบายไว้ในส่วนนี้ ดังตารางด้านล่าง

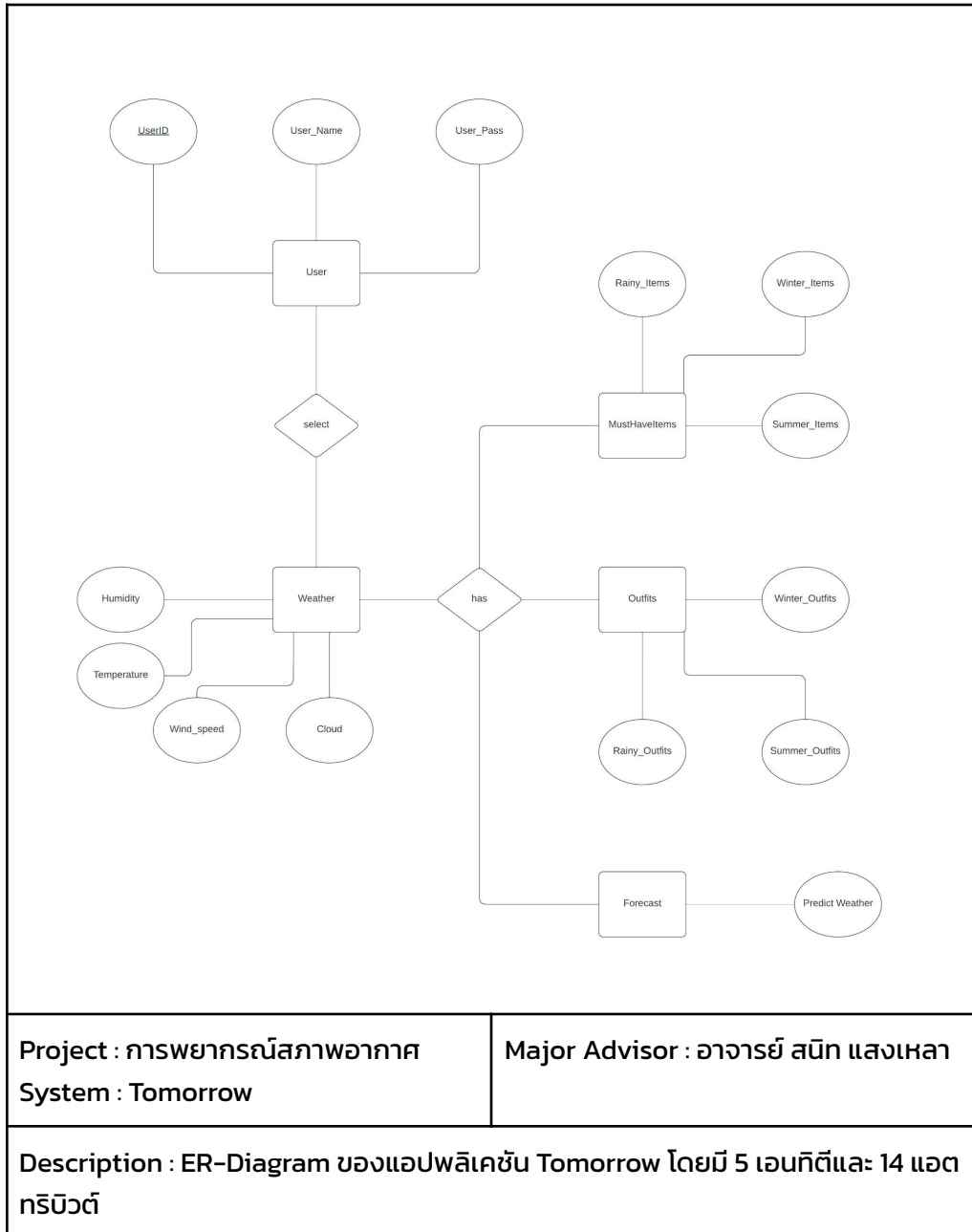
Table 8: List of All Data Elements

<i>SEQ</i>	<i>Data Element Name</i>	<i>From Process/Source/Data Store</i>	<i>To Process/Source/Data Store</i>
1	<i>Username</i>	<i>User</i>	<i>Login/Registration</i>
2	<i>Password</i>	<i>User</i>	<i>Login/Registration</i>

3.4 Database Analysis and Design

ในหัวข้อนี้จะกล่าวถึงหัวข้อการวิเคราะห์และออกแบบฐานข้อมูล เช่น ER-Diagram, Relational Diagram, และ Interface design โดยจะมีการอธิบายขั้นตอนการใช้แอปพลิเคชันด้วยเช่นกัน

3.4.1 ER-Diagram



3.4.2 Relational Schema

หัวข้อนี้อธิบาย attributes จากตารางในฐานข้อมูล โดยความหมายของสัญลักษณ์อธิบายได้ดังนี้

- Attributes : ตัวหนาและขีดเส้นใต้คือคีย์หลัก (Primary Key)
- *Attributes* : ตัวเอียงคือ Foreign Keys
- Attributes : ตัวหนา ตัวเอียง และขีดเส้นใต้เป็นทั้ง Primary Key และ Foreign Keys

ตารางในระบบสามารถแบ่งได้เป็น 3 กลุ่ม ดังนี้ :

- Master File Table
- Base File Table
- Transaction File Table

Table	Table Name	Table Type	Description
1	Weather	Master	เก็บข้อมูลสภาพอากาศ จัดเก็บหมวดหมู่ของใช้ และเสื้อผ้า
2	Must Have Items	Master	เก็บข้อมูลสิ่งของที่ จำเป็น
3	Outfits	Master	เก็บข้อมูลเสื้อผ้าที่แนะนำ
4	Forecast	Master	เก็บข้อมูลการพยากรณ์ อากาศล่วงหน้า
5	User	Base	จัดเก็บข้อมูลของผู้ใช้

1. Relational Schema of Master File Tables

- Weather
- Must Have Items
- Outfits

2. Relational Schema of Base File Tables are

- User

User

<u>UserID</u>	User_Name	User_Pass
---------------	-----------	-----------

Weather

Humidity	Temperature	Wind_speed	Cloud
----------	-------------	------------	-------

MustHaveItems

Rainy_Items	Winter_Items	Summer_Items
-------------	--------------	--------------

Outfits

Rainy_Outfits	Winter_Outfits	Summer_Outfits
---------------	----------------	----------------

Forecast

Predict Weather

Relational Schema

3.4.3 File Structure

ส่วนนี้แสดงรายละเอียดของแต่ละส่วนของไฟล์ ได้แก่ field name, field description, field data type, field length, null value, primary key และ foreign key

File Structure Of Weather

Table Name : Weather					
Table Type : Master					
Description : Weather รวบรวมข้อมูลสภาพอากาศในหน้าหลัก					
Field Name	Type	Description	Key	Reference	Null
Humidity	Number	เก็บข้อมูลความชื้น	-	-	-
Temperature	Number	เก็บข้อมูลอุณหภูมิ	-	-	-
Wind_speed	Number	เก็บข้อมูลความเร็วลม	-	-	-
Cloud	Number	เก็บข้อมูลปริมาณเมฆ			

File Structure Of Must Have Items

Table Name : Must Have Items					
Table Type : Master					
Description : Must Have Items รวบรวมข้อมูลสิ่งของที่จำเป็น					
Field Name	Type	Description	Key	Reference	Null
Rainy_Items	Image	เก็บข้อมูลของใช้ในฤดูฝน	-	-	-
Winter_Items	Image	เก็บข้อมูลของใช้ในฤดูหนาว	-	-	-
Summer_Items	Image	เก็บข้อมูลของใช้ในฤดูร้อน	-	-	-

File Structure Of Outfits

Table Name : Outfits					
Table Type : Master					
Description : Outfits รวบรวมข้อมูลเสื้อผ้าที่จำเป็น					
Field Name	Type	Description	Key	Reference	Null
Rainy_Outfits	Image	เก็บข้อมูลเสื้อผ้าในฤดูฝน	-	-	-
Winter_Outfits	Image	เก็บข้อมูลเสื้อผ้าในฤดูหนาว	-	-	-
Summer_Outfits	Image	เก็บข้อมูลเสื้อผ้าในฤดูร้อน	-	-	-

File Structure Of Forecast

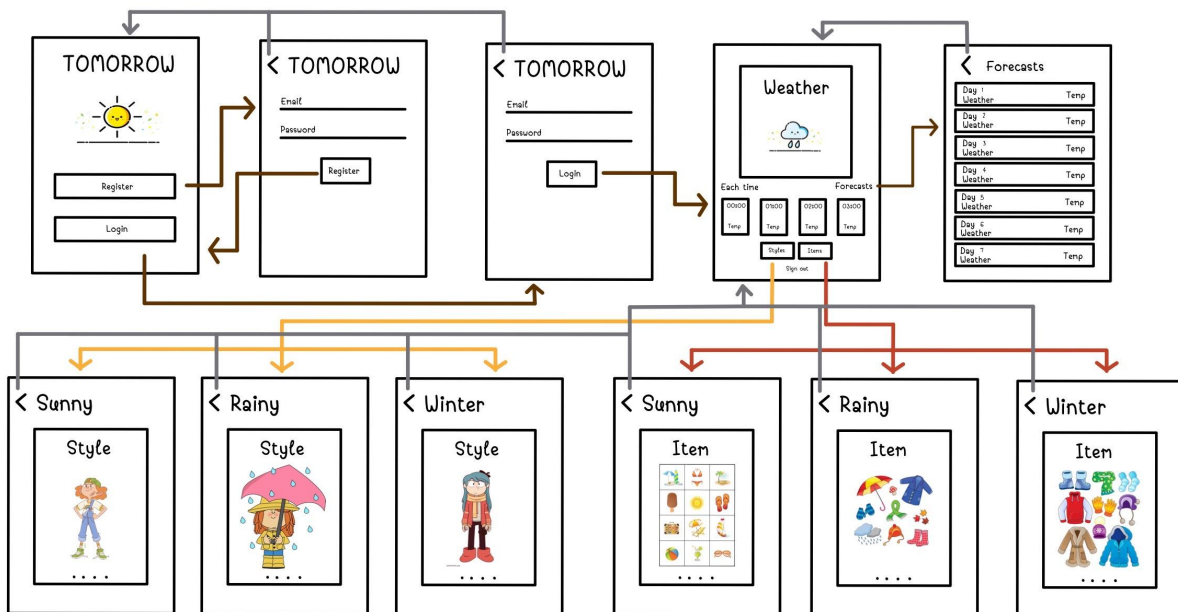
Table Name : Forecast					
Table Type : Master					
Description : Forecast รวบรวมข้อมูลการพยากรณ์อากาศล่วงหน้า 7 วัน					
Field Name	Type	Description	Key	Reference	Null
Predict Weather	Text, Number	เก็บข้อมูลการพยากรณ์อากาศล่วงหน้า 7 วัน	-	-	-

File Structure Of User

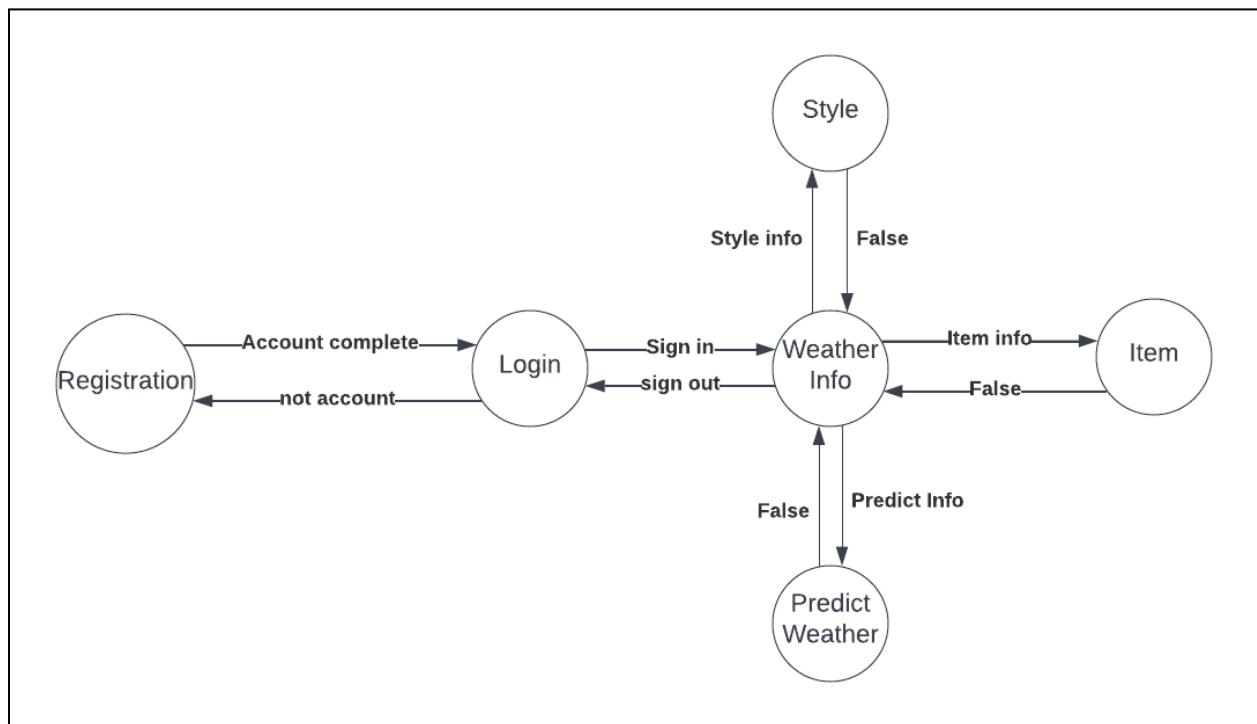
Table Name : User					
Table Type : Base					
Description : User รวบรวมข้อมูลของผู้ใช้					
Field Name	Type	Description	Key	Reference	Null
UserID	Number	เก็บข้อมูลไอดีของผู้ใช้แต่ละคน	PK	-	-
User_Name	Text	เก็บข้อมูลชื่อผู้ใช้	-	-	-
User_Pass	Text	เก็บข้อมูลรหัสผ่านของผู้ใช้	-	-	-

3.5 I/O Design

3.5.1 Interface Design



3.5.2 Transition Diagram



ในส่วนนี้คือแผนภาพการเปลี่ยนแปลงของ application Tomorrow โดยเป็นเส้นทางการกำหนดทิศทางแต่ละหน้า เริ่มต้นด้วยสถานะการเข้าระบบ โดยผู้ใช้งานต้องเข้าสู่ระบบก่อนใช้แอปพลิเคชัน ถ้าไม่เช่นนั้นจะต้องทำการลงทะเบียน เมื่อผู้ใช้เข้าสู่ระบบโดยสมบูรณ์แล้ว คุณจะเจอหน้า Weather Info ที่แสดง สภาพภูมิอากาศ อุณหภูมิ และจะมีการทำนายอุณหภูมิล่วงหน้า 7 วัน สิ่งที่จะต้องมามี โดยในหน้านี้คุณสามารถเลือกที่จะเข้าไปดู style การแต่งตัวที่เหมาะสมกับสภาพภูมิอากาศได้ หรือกดเข้าไปดู must have item ที่จำเป็นต้องมี และยังสามารถดูการทำนายล่วงหน้า 7 วัน ได้อีกด้วย โดยเมื่อดูเสร็จจะมีปุ่มที่คลิกเลือกให้ย้อนกลับไปยังหน้า Weather Info ได้

บทที่ 4

IMPLEMENTATION

บทนี้เป็นการอธิบายเกี่ยวกับการดูแลสภาพแวดล้อมของฮาร์ดแวร์และซอฟต์แวร์ที่เหมาะสมเพื่อให้อุปกรณ์ทำงานได้อย่างมีประสิทธิภาพ รวมถึงคู่มือและเทคนิคการใช้งานของระบบของเรา

4.1 Hardware and System Environment

- Operating System and Utilities Applications

- o Hardware

- Laptop hardware 2: o Software AMD Ryzen 9 5900HX with Radeon Graphics 3.30 GHz ติดตั้ง RAM แล้ว 16.0 GB

- Operation system: Microsoft Windows 11

- Android studio, Firebase, Flutter, Visual Studio, Microsoft Word, Google Chrome, Microsoft Edge

- Web Server Software

- o Google Cloud and Firebase

Firebase คือบริการหนึ่งของ Google เป็นการบริการข้อมูลแบบออนไลน์ในรูปแบบ Real Time Database สำหรับ Application และ Web Application

Firebase มอบแพลตฟอร์มที่ครอบคลุมสำหรับการสร้าง ปรับใช้ และขยายแอปพลิเคชันมือถือและเว็บคุณภาพสูง ฐานข้อมูลแบบเรียลไทม์ บริการยืนยันตัวตน พื้นที่เก็บข้อมูลบนคลาวด์ โซลูชันโฮสติ้ง และเครื่องมือวิเคราะห์ทำให้เป็นร้านค้าครบวงจรสำหรับการพัฒนาแอป ช่วยให้นักพัฒนาสามารถมุ่งเน้นไปที่การสร้างแอปที่ยอดเยียมสำหรับผู้ใช้งานของตน

Google ระบุว่า Cloud Firestore ถูกออกแบบมาให้เป็น data backend โดยสมบูรณ์แบบที่สามารถจัดการด้านความปลอดภัยและการยืนยันตัวตน, โครงสร้างพื้นฐาน, การซิงค์ และอื่น ๆ รวมถึงออกแบบมาให้อินทิเกรตเข้ากับ GCP และ Firebase จึงทำให้สามารถใช้ร่วมกับเซอร์วิสต่าง ๆ บนแพลตฟอร์มทั้งสองได้ง่าย

ในโครงการนี้ เรามุ่งเน้นไปที่การนำระบบคลาวด์ไปใช้เป็นส่วนใหญ่ ซึ่งเป็นฐานข้อมูลแบบ NoSQL แบบเรียลไทม์ที่ปรับขนาดได้จำนวนมาก และไม่ใช้ภาษา SQL ในการจัดการข้อมูล แต่ออกแบบให้มีความยืดหยุ่นในการใช้งาน

o Firebase Cloud Firestore

Cloud Firestore เป็นหนึ่งในบริการฐานข้อมูลของ Firebase โดยมีต้นแบบมาจาก Realtime database ซึ่งเป็น NoSQL database มีโครงสร้างข้อมูลแบบ Document สามารถใช้ทั้ง iOS, Android และ Web ตลอดจนสามารถ integrate ได้กับของที่มีอยู่ใน Firebase รวมถึง Google Cloud Platform, Cloud function

Flexibility มีโครงสร้างการเก็บข้อมูลแบบลำดับชั้น ข้อมูลทั้งหมดใน 1 record จะถูกจัดเก็บใน document

Expressive querying สามารถ query เพื่อหา document แต่ละอันหรือเฉพาะเจาะจง หรือเพื่อถึง document ทั้งหมดใน Collection

Realtime updates เหมือนกับ Realtime database ตรงที่ใช้การซิงโครไนซ์ข้อมูลเพื่ออัปเดตข้อมูลบนอุปกรณ์ที่เชื่อมต่อ แต่ Cloud Firestore ได้รับการออกแบบมาให้ง่ายต่อการดึงข้อมูลในครั้งเดียวอย่างมีประสิทธิภาพ

Offline support จะ cache ข้อมูลที่แอปที่ใช้งานอยู่ ดังนั้นแอปจึงสามารถเขียนอ่าน และค้นข้อมูลได้แม้ว่าอุปกรณ์จะออฟไลน์อยู่

Designed to scale เป็น infrastructure ที่ดีที่สุดของ Google Cloud

โดย Cloud Firestore รองรับการเติบโตของข้อมูลในอนาคตได้ โดยเราสามารถออกแบบโครงสร้างได้ทุกรูปแบบที่จะสามารถทำงานได้อย่างดีที่สุดในของแอปเรา

o Firebase Authentication

Firebase Authentication เป็น Service หนึ่งของ Firebase ที่ช่วยเรื่องของการทำ Authentication ให้เป็นเรื่องง่าย มีความปลอดภัยสูง โดยที่ตัว Firebase จะจัดการพวก Login, Register หรือ Session การล็อกอินของ User ให้เรา มี UI พื้นฐานไว้ให้ใช้ รองรับการ Login ด้วย Email กับ Password แบบปกติ หรือจะเป็นการ Login เข้าสู่ระบบด้วย Provider ต่างๆ ไม่ว่าจะเป็น Gmail, Twitter, Facebook, Github, ผ่านมือถือ หรืออื่นๆอีกมากมาย เรียกได้ว่าครบจบในที่เดียว

ในโครงการของเรา ใช้ Firebase Authentication เพื่อใช้ฟังก์ชันการเข้าสู่ระบบโดยใช้อีเมลและรหัสผ่าน

- Database Management System (DBMS)
- Programming and Scripting Tools

o Android Studio

เป็นเครื่องมือพัฒนา (IDE : Integrated Development Environment) ที่ถูกสร้างขึ้นมาเพื่อการพัฒนาแอนดรอยด์แอปพลิเคชัน บนพื้นฐานของแนวคิด IntelliJ โดยมีความสามารถในการทำงานเด่น ๆ ในขณะนี้ ดังต่อไปนี้

- มีความยืดหยุ่นในการใช้งานด้วยใช้ Gradle-based
- การสร้างตัวแปรและการสร้างไฟล์ APK ในหลาย ๆ แม่แบบ
- แม่แบบที่ช่วยอำนวยความสะดวกในการใช้งานคุณสมบัติที่ถูกใช้งานบ่อย ๆ
- ตัวช่วยแก้ไขรูปแบบ (Layout) ที่รองรับการลากและวาง เครื่องมือที่จะตรวจจับ ประสิทธิภาพการใช้งาน,การทำงานร่วมกันรุ่นและปัญหาอื่นๆ
- การสนับสนุนนิวทริกอนสำหรับแพลตฟอร์ม Google Cloud ทำให้มันง่ายต่อการรวม Google Cloud Messaging และ App Engine และอื่น ๆ

o Dart Programming language

Dart เป็นภาษาการเขียนโปรแกรมเชิงวัตถุแบบ Open Source วัตถุประสงค์ทั่วไป โดยมีไวยากรณ์แบบ C ที่พัฒนาโดย Google ในปี 2554 จุดประสงค์ของการเขียนโปรแกรม Dart คือการสร้างส่วนต่อประสานผู้ใช้ส่วนหน้าสำหรับเว็บและแอปมือถือ อยู่ระหว่างการพัฒนา คอมไพล์เป็นโค้ดเนทีฟสำหรับสร้างแอปบนอุปกรณ์เคลื่อนที่ โดยได้รับแรงบันดาลใจจากภาษาโปรแกรมอื่นๆ เช่น Java, JavaScript, C# และเป็นแบบ Strongly Typed เนื่องจาก Dart เป็นภาษาที่ compile ดังนั้นจึงไม่สามารถเรียกใช้ code ได้โดยตรง คอมไพเลอร์จะแยกวิเคราะห์และถ่ายโอนไปยังรหัสเครื่องแทน

มีสิ่งที่จะช่วยสนับสนุนการทำงานให้สะดวกมากยิ่งขึ้น เช่น คลาส หรือ ฟังก์ชัน ซึ่งแตกต่างจากภาษาโปรแกรมอื่นๆ ภาษา Dart มีข้อจำกัด คือ ไม่รองรับ Array โดยตรง

แต่สนับสนุนการรวบรวมซึ่งใช้เพื่อทำซ้ำโครงสร้างข้อมูล เช่น Array ข้อมูลทั่วไป และการพิมพ์เพิ่มเติม

ประสิทธิภาพ

รวดเร็ว ใช้คอมไพเลอร์ source tools เพื่อจัดเรียงโค้ดให้เร็วขึ้น ดังนั้นจึงมี UI ที่มีประสิทธิภาพมากกว่าเมื่อเปรียบเทียบกับภาษาโปรแกรมอื่นๆ

สะดวกในการใช้

Dartpad เป็นเครื่องมือเขียนโปรแกรม Dart ช่วยให้งานในภาษาโดยตรงจากเบราว์เซอร์ได้ง่ายขึ้นโดยไม่จำเป็นต้องติดตั้งซอฟต์แวร์ใดๆ

Flutter Framework

Framework Flutter โดย Dart ใช้เพื่อสร้างแอปพลิเคชันข้ามแพลตฟอร์ม (แอปพลิเคชันที่เข้ากันได้กับระบบปฏิบัติการมือถือหลายระบบ) ติดอันดับ 1 ใน 3 framework การพัฒนาแอปมือถือและมียอดขายกว่า 1.1 หมื่นล้านครั้ง framework Flutter ของ Dart ถูกใช้เพื่อสร้างแอปสำหรับอีคอมเมิร์ซยอดนิยม เช่น alibaba หรือ ebay บริษัทอื่นๆ ที่ใช้ Flutter ได้แก่ BMW, Google Pay, ByteDance, Hamilton และ Toyota เนื่องจาก Flutter ใช้ Dart ซึ่งหากเรียนรู้จะเป็นประโยชน์ต่อการพัฒนา Flutter

ความยืดหยุ่น

Dart ใช้กับ JavaScript ได้ นอกจากนี้ นักพัฒนาสามารถเรียกใช้โปรแกรมบน Dart ได้เลย แม้ว่าจะมีข้อผิดพลาดเล็กน้อยในการ compile ก็ตาม

อีกเหตุผลสำคัญที่คุณควรเรียนรู้ Dart เป็นเพราะความยืดหยุ่นและความเข้ากันได้กับ JavaScript ยิ่งไปกว่านั้น นักพัฒนาสามารถเรียกใช้โปรแกรมบน Dart ได้ แม้ว่าจะมีข้อผิดพลาดเล็กน้อยในการคอมไพล์ก็ตาม

คุณลักษณะของภาษาการเขียนโปรแกรม Dart

Dart เป็นภาษาโปรแกรมแบบ Open Source พร้อม Feature ที่มีประสิทธิภาพที่ใช้งานได้ฟรี และเป็นที่ยอมรับในหมู่ software developer

Object-oriented

ภาษาโปรแกรมเชิงวัตถุ Dart ใช้ข้อมูลเป็นวัตถุแทนที่จะถือว่าเป็นฟังก์ชันหรือตรรกะ ซึ่งรองรับแนวทางการเขียนโปรแกรมเชิงวัตถุทั้งขั้นพื้นฐานและขั้นสูง เช่น class, interface และ abstract

Asynchronous

เป็นโปรแกรมภาษาแบบ Asynchronous ที่ช่วยให้ทำงานพร้อมกันได้ดียิ่งขึ้น โดยสามารถทำงานพร้อมกันได้ เรียก isolate entity

รองรับหลาย Platform

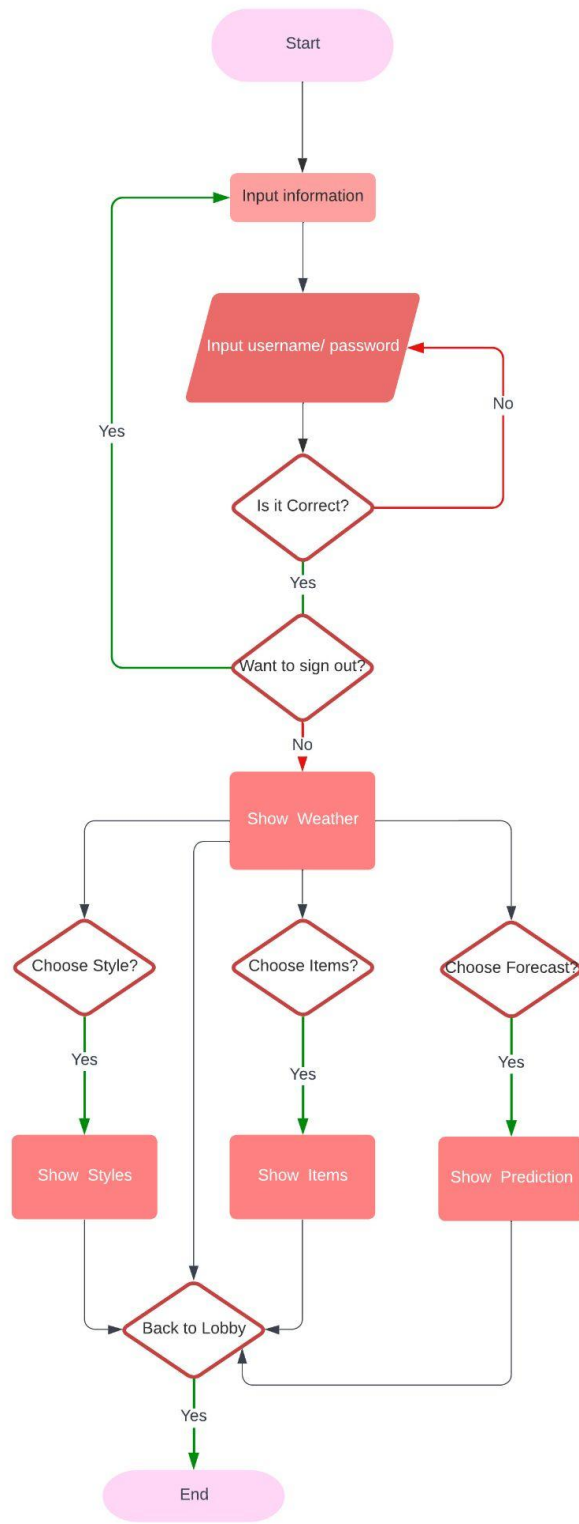
สามารถทำงานบนระบบปฏิบัติการได้หลากหลาย เช่น Windows, Linux, macOS โดย Feature ที่ช่วยทำให้ทำงานในระบบปฏิบัติการที่ต่างกันได้ คือ Dart Virtual Machine (Dart VM)

Library ที่สร้างขึ้น

Dart ประกอบด้วย library มากมาย เช่น Input-Output (IO), Software Development Kit (SDK), Convert, Core, Asynchronous, Auth ซึ่งช่วยเพิ่มประสิทธิภาพการทำงานให้สมบูรณ์ยิ่งขึ้น

ส่วนประกอบ

ในส่วนนี้จะเป็นการอธิบายส่วนประกอบของ applications "Tomorrow"



โดยจากรูป Flow Chart ข้างบน คือการแสดงผลรวมของระบบดังนี้

ขั้นแรกคือการลงทะเบียนผู้ใช้ ซึ่งผู้ใช้ต้องใส่ข้อมูล เช่น ชื่อ อีเมล ชื่อผู้ใช้และรหัสผ่าน เป็นต้น

ขั้นที่สองคือการล็อกอินเข้าสู่ระบบ โดยผู้ใช้ต้องป้อนชื่อผู้ใช้และรหัสผ่าน หากกรอกถูกต้อง ระบบจะเข้าสู่หน้าหลักของแอป แต่ถ้าหากผู้ใช้ต้องการออกจากระบบจะกลับไปสู่หน้าการเข้าสู่ระบบ และถ้าหากผู้ใช้กรอกข้อมูลไม่ถูกต้องผู้ใช้จะต้องป้อนข้อมูลใหม่อีกครั้ง

ขั้นที่สามในหน้าหลักของแอปพลิเคชันจะแสดงเนื้อหาเกี่ยวกับสภาพอากาศในวันและเวลาปัจจุบันและสามารถเลือกดูฟิวเจอร์ต่างๆได้ เช่น เลือกดูสัปดาห์ ไอเทม และดูพยากรณ์อากาศล่วงหน้าได้ 7 วัน หรือเลือกกลับไปยังหน้าการเข้าสู่ระบบได้

ขั้นต่อไปหากผู้ใช้กดเลือกดูสัปดาห์ ไอเทม หรือการทำนาย ระบบจะพาผู้ใช้ไปสู่หน้าที่มีคำแนะนำสัปดาห์ สิ่งของที่จำเป็นและ หน้าการพยากรณ์ล่วงหน้า โดยในแต่ละหน้าผู้ใช้สามารถเลือกกลับไปสู่หน้าหลักของระบบได้

สุดท้ายหากผู้ใช้ต้องการออกจากระบบสามารถกด Back to Lobby และเลือกออกจากระบบ

4.2 คู่มือการนำไปใช้งานและเทคนิค

4.2.1 คู่มือการเชื่อมต่อ

สำหรับการดำเนินโครงการนั้น จำเป็นต้องตั้งค่าปลั๊กอินบางตัว เช่น FlutterFire FlutterFire เป็นชุด plug in Flutter ที่เชื่อมต่อแอปพลิเคชัน Flutter กับ firebase เราจำเป็นต้องติดตั้ง "firebase_core" เพื่อเชื่อมต่อแอปพลิเคชันกับ Firebase ก่อนใช้บริการ Firebase ดังนั้นเราจึงจำเป็นต้องเรียกใช้คำสั่ง "flutter pub add firebase_core" จากนั้นมันจะติดตั้งปลั๊กอินตามที่แสดงในบรรทัดแรกในกล่อง ปลั๊กอินที่เราต้องติดตั้งคือ "cloud_firestore" เพื่อเริ่มใช้งาน Cloud Firestore ในโครงการ Flutter ของเรา ดังนั้นเราจึงจำเป็นต้องเรียกใช้คำสั่ง "flutter pub add cloud_firestore" จากนั้นมันจะติดตั้งปลั๊กอินเข้ามาในโค้ด

4.2.2 เทคนิคต่างๆที่ใช้

ขั้นตอนแรกคือการป้อนข้อมูลเข้าสู่ระบบจากนั้นข้อมูลจะแสดงในมุมมอง listview และผู้ใช้สามารถดูข้อมูลสภาพอากาศ สิ่งของที่ต้องเตรียมตัวได้ โดย listview สร้างผ่านฟังก์ชัน ListView.builder() เมื่อสร้างเสร็จแล้วจะปรากฏข้อมูลออกมาเป็นแถวที่สวยงาม

บทที่ 5

ปัญหาความท้าทายและผลการแก้ปัญหาที่เป็นไปได้

ปัญหาของทีมเราเกิดจากเวลา login เข้าสู่ระบบ จะไม่สามารถ login ได้หากรันผ่าน chrome จำเป็นต้องรันผ่าน mobile เท่านั้น

โดยเราได้แก้ไขโดย

- **รวบรวมข้อมูล:** รวบรวมรายละเอียดจากสมาชิกในทีมที่ประสบปัญหา สอบถามเฉพาะเจาะจง เช่น ข้อความแสดงข้อผิดพลาดที่พบระหว่างกระบวนการเข้าสู่ระบบ แพลตฟอร์มและเวอร์ชันของ Chrome ที่กำลังใช้ เวอร์ชัน Flutter ที่กำลังใช้ และการกำหนดค่าระบบที่เกี่ยวข้อง
- **ทำซ้ำปัญหา:** พยายามทำซ้ำปัญหาโดยลงชื่อเข้าใช้ระบบโดยใช้ Flutter ใน Chrome บนอุปกรณ์ต่างๆ จดบันทึกข้อความแสดงข้อผิดพลาดหรือลักษณะการทำงานที่ไม่คาดคิดซึ่งพบในระหว่างกระบวนการ
- **ตรวจสอบความเข้ากันได้ของ Flutter และ Chrome:** ตรวจสอบให้แน่ใจว่าเวอร์ชันของ Flutter ที่ใช้นั้นเข้ากันได้กับ Chrome เวอร์ชันที่ใช้อยู่ ตรวจสอบเอกสารประกอบของ Flutter และบันทึกประจำรุ่นสำหรับปัญหาที่ทราบหรือข้อกำหนดด้านความเข้ากันได้เมื่อใช้ Flutter กับ Chrome
- **ตรวจสอบความต้องการของระบบ:** ยืนยันว่าระบบเป็นไปตามข้อกำหนดที่จำเป็นสำหรับการเรียกใช้แอปพลิเคชัน Flutter ใน Chrome เช่น การติดตั้ง Flutter SDK เวอร์ชันล่าสุด Dart SDK เวอร์ชันที่เข้ากันได้ และการพึ่งพาและปลั๊กอินที่จำเป็นสำหรับการพัฒนาเว็บ Flutter
- **ตรวจสอบการกำหนดค่าเว็บ Flutter:** ตรวจสอบให้แน่ใจว่าโครงการ Flutter ได้รับการกำหนดค่าอย่างถูกต้องสำหรับการพัฒนาเว็บ ตรวจสอบการตั้งค่าโปรเจกต์ Flutter รวมถึงการอ้างอิงและการกำหนดค่าที่เกี่ยวข้องกับการพัฒนาเว็บ เพื่อให้แน่ใจว่าการตั้งค่าอย่างถูกต้อง
- **ตรวจสอบการกำหนดค่าเว็บของโครงการ Flutter :** เพื่อให้แน่ใจว่าการตั้งค่าอย่างเหมาะสมสำหรับการพัฒนาเว็บ ตรวจสอบการตั้งค่าโปรเจกต์ Flutter รวมถึงการอ้างอิงและการกำหนดค่าการพัฒนาเว็บ เพื่อให้แน่ใจว่าการกำหนดค่าอย่างถูกต้อง

**** ทดสอบกับเบราว์เซอร์อื่นๆ:** เพื่อตรวจสอบว่าปัญหาเกิดเฉพาะ Chrome หรือส่งผลกระทบต่อเบราว์เซอร์หลายตัวหรือไม่ ให้ลองเรียกใช้แอปพลิเคชัน Flutter บนเบราว์เซอร์อื่นที่ไม่ใช่ Chrome เช่น Firefox หรือ Safari วิธีนี้สามารถช่วยในการระบุสาเหตุที่แท้จริงของปัญหาได้

บทที่ 6

การทดสอบและประเมินผล

ในบทนี้จะกล่าวถึงการทำงานของโปรแกรม การทดสอบ และประเมินผลโดยรวม มี 2 ส่วน คือ การทดสอบ Unit ประกอบด้วย ส่วนแรก คือ Unit Tests และส่วนที่สอง คือ System integration tests ดังนั้น ผู้อ่านจะได้เรียนรู้เพิ่มเติมเกี่ยวกับขั้นตอนการทดสอบ Unit Tests, ผลการทดสอบ, สถานการณ์ตัวอย่าง และคำอธิบายที่ครอบคลุมเกี่ยวกับวิธีใช้โปรแกรม

6.1 Unit Tests

สำหรับ unit tests กระบวนการที่สำคัญ สำหรับ unit testing

Process ที่เลือก มีดังนี้

- Process 1 : หน้า Register
- Process 2: หน้า Login
- Process 3: หน้า Weather

6.1.1 การทดสอบใน process 1 : Register

สำหรับฟังก์ชัน Register จะมี 3 อย่าง Username, Email, Password

Process ที่ดำเนินการแล้ว	เงื่อนไขในการทดสอบ	ผลลัพธ์ที่แท้จริง
ข้อความเตือน: ผู้ใช้ไม่ได้ส่งอะไรไประบบ	ระบบจะแสดงข้อความแจ้งเตือนภายในช่องว่างทุกช่อง เพื่อแจ้งให้ผู้ใช้ทราบอย่างครบถ้วนถึงปัญหา	Pass
ข้อความเตือน รูปแบบEmail: ผู้ใช้ส่งรูปแบบอีเมลผิด	ระบบจะแสดงข้อความแจ้งเตือนภายในใต้กล่องอีเมลไม่ถูกต้อง: "กรุณารอกอีเมลให้ถูกต้อง"	Pass
ข้อความเตือน รูปแบบEmail: ผู้ใช้ส่งอีเมลที่มีอยู่แล้วในระบบ	ระบบจะแสดงกล่องข้อความแจ้งเตือน ด้านล่างหน้าจอแจ้งว่า มีอีเมลอยู่ในระบบแล้ว:	Pass

	“มีบัญชีที่ใช้อีเมลนี้แล้ว”	
ข้อความเตือน รูปแบบ Password: ผู้ใช้ส่งรหัสผ่านที่น้อยกว่า 6 อักขระ	ระบบแสดงกล่องข้อความด้านล่างหน้าจอว่าความยาวของรหัสผ่าน ไม่ควรน้อยกว่า 6 ตัวอักษร : “ รหัสผ่านควรมีอย่างน้อย 6 อักขระ ”	Pass

6.1.2 การทดสอบใน process 2 : Login

สำหรับฟังก์ชันการเข้าสู่ระบบจะมีช่องให้ผู้ใช้อัปโหลดข้อมูล 2 ช่องคือ อีเมลและรหัสผ่าน

Process ที่ดำเนินการแล้ว	เงื่อนไขในการทดสอบ	ผลลัพธ์ที่แท้จริง
ข้อความเตือน: ผู้ใช้ส่งข้อมูลที่ไม่ได้กรอกเข้าสู่ระบบ	ระบบจะแสดงข้อความเตือนใต้ช่องว่างทุกช่อง เพื่อแจ้งให้ผู้ใช้กรอกข้อมูล	Pass
ข้อความเตือน: อีเมล - ผู้ใช้ส่งอีเมลที่ไม่ได้ลงทะเบียนไปยังระบบ	ระบบจะแสดงข้อความเตือนว่าอีเมลดังกล่าวไม่สามารถใช้เข้าสู่ระบบได้เนื่องจากไม่มีการบันทึกในระบบ	Pass
ข้อความเตือน: รหัสผ่าน - ผู้ใช้ส่งรหัสผ่านผิดเข้าสู่ระบบ	ระบบจะแสดงข้อความเตือนว่ารหัสผ่านไม่ถูกต้อง	Pass

6.1.3 การทดสอบใน process 3 : Weather

สำหรับฟังก์ชันนี้ผู้ใช้ จะมีการใช้งาน อยู่ 2 ส่วน เลือก styles และ เลือก items

Process ที่ดำเนินการแล้ว	เงื่อนไขในการทดสอบ	ผลลัพธ์ที่แท้จริง
Page routing: ผู้ใช้จะคลิกเข้าไปในปุ่ม "styles"	ระบบจะอนุญาตให้ผู้ใช้ดู styles ที่เหมาะสมกับสภาพอากาศ	Pass
Page routing: ผู้ใช้จะคลิกเข้าไปในปุ่ม "items"	ระบบจะอนุญาตให้ผู้ใช้ดู items ที่เหมาะสมกับสภาพอากาศ	Pass

6.2 System Integration Test

หัวข้อนี้มีวัตถุประสงค์เพื่อตรวจสอบว่าระบบทำงานได้ถูกต้องตาม function ที่ต้องการหรือไม่

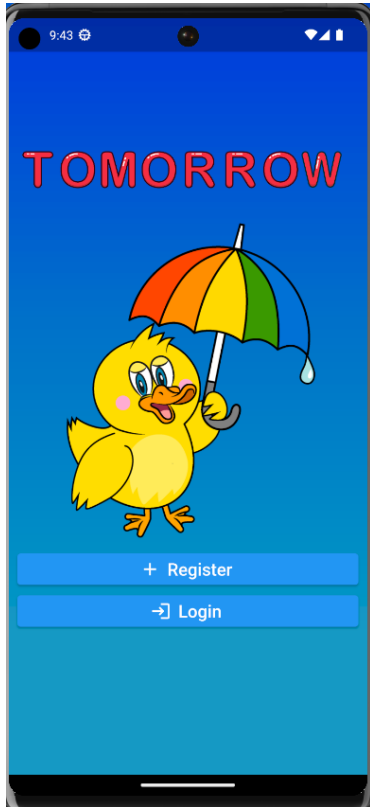
6.2.1 Test Scenario

เพื่อทดสอบการทำงานทุกด้านของระบบอย่างละเอียด ผู้จัดทำได้กำหนดสถานการณ์การทดสอบไว้ทั้งหมด 3 function หลัก ดังนี้

1. **Register** ผู้ใช้ลงทะเบียนเข้าสู่ระบบ
2. **Login** ผู้ใช้เข้าสู่ระบบ
3. **Weather** ผู้ใช้ใช้ function สภาพภูมิอากาศ

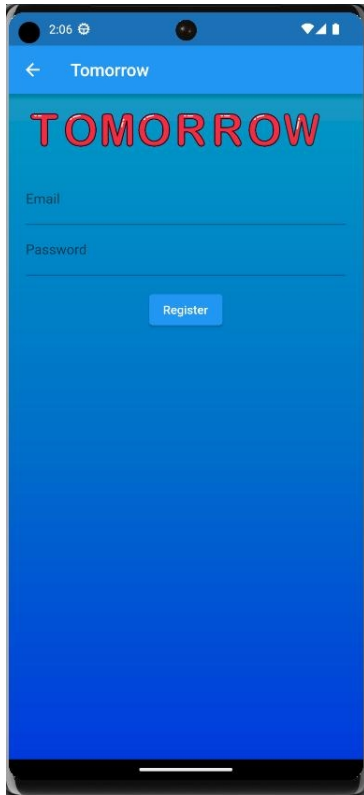
นอกจากนี้ ในส่วนนี้จะครอบคลุมการตรวจสอบและฟังก์ชันที่จำเป็นในการใช้ระบบของเรา การทดสอบสถานการณ์นี้จะช่วยให้เกิดประโยชน์ต่อผู้ที่ต้องการใช้งานแอปพลิเคชัน ในส่วนถัดไปเป็นการกล่าวถึงรายละเอียดการทดสอบในแต่ละ function

6.2.1.1 Registration ผู้ใช้ลงทะเบียนเข้าสู่ระบบ

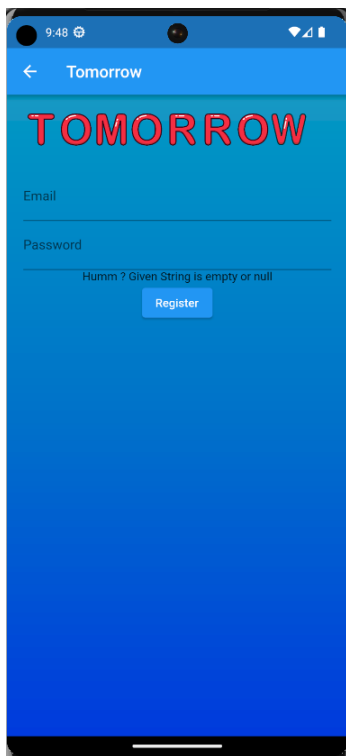


ขั้นตอนนี้ผู้ใช้ที่เข้าสู่หน้าแรกของแอปพลิเคชัน

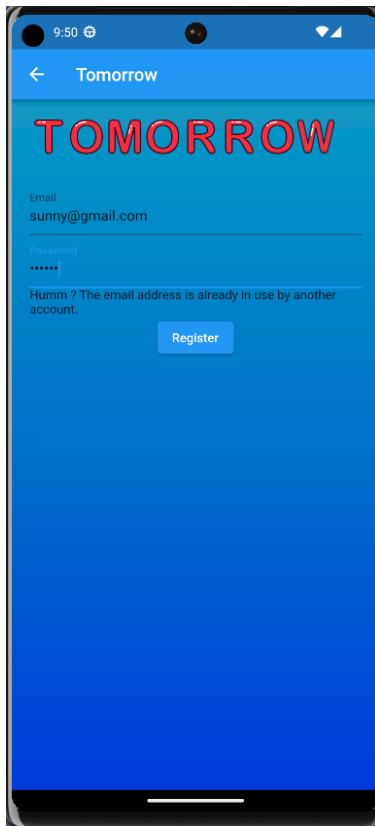
- เลือกระหว่างลงทะเบียนหรือเข้าสู่ระบบ
- ในผู้ใช้เริ่มแรก ผู้ใช้เข้าสู่แอปพลิเคชันและเลือกลงทะเบียนเพื่อเข้าสู่ระบบ



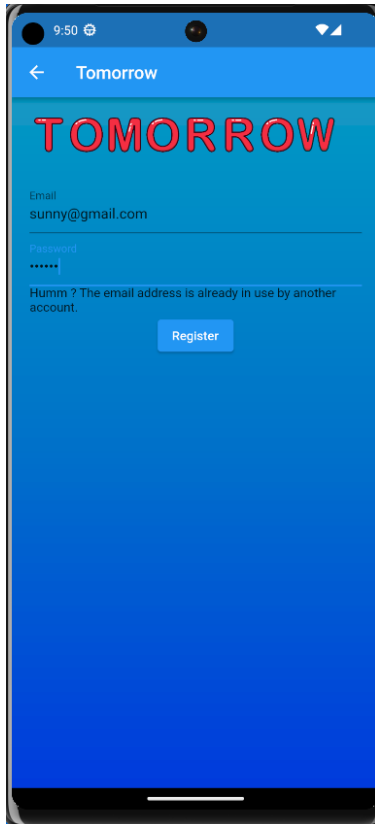
- หลังจากที่ผู้ใช้คลิกปุ่ม “ลงทะเบียน” ระบบจะนำผู้ใช้อย่างงหน้านี้
- หน้า Register มีสองช่องให้ผู้ใช้กรอกข้อมูล ได้แก่ Email และ Password
- มีปุ่ม 2 ปุ่มคือปุ่ม “Register” และปุ่มสำหรับย้อนกลับไปที่หน้าที่แล้ว
- ผู้ใช้สามารถคลิก “Register” เพื่อลงทะเบียนเข้าสู่ระบบ หรือคลิกปุ่มเครื่องหมายลูกศรเพื่อย้อนกลับ



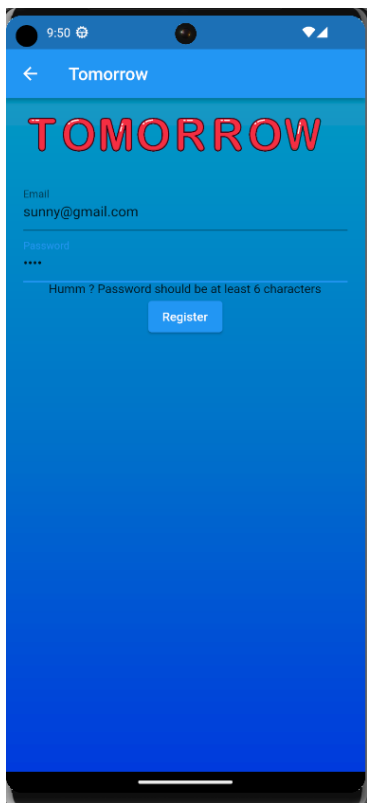
- กรณีที่ผู้ใช้กด “Register” โดยที่กรอกข้อมูลไม่ครบถ้วน ข้อความเตือนจะปรากฏข้างใต้



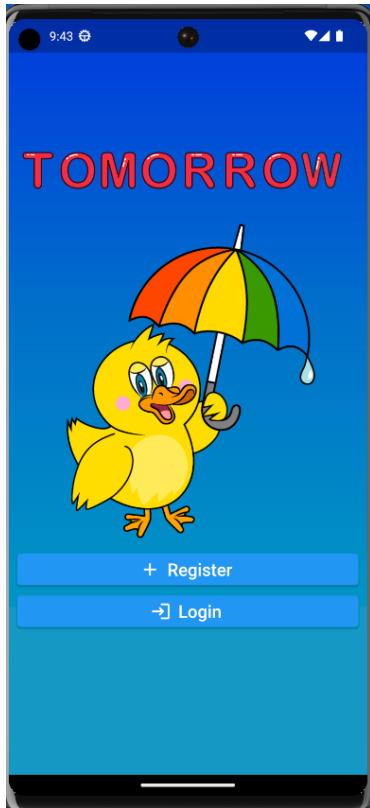
- ขั้นตอนนี้จะเป็นการใส่ข้อมูลของผู้ใช้ตามลำดับ คือ E-mail และ รหัสผ่าน ดังตัวอย่างคือ
- E-mail sunny@gmail.com
- รหัสผ่าน 123456



- หากอีเมลของผู้ใช้มีรูปแบบอีเมลที่มีอยู่แล้วจะปรากฏข้างใต้กล่องเพื่อแจ้งให้ผู้ใช้ใส่อีเมลที่ถูกต้อง

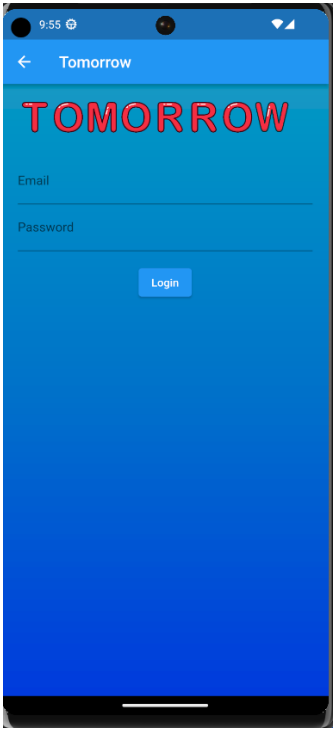
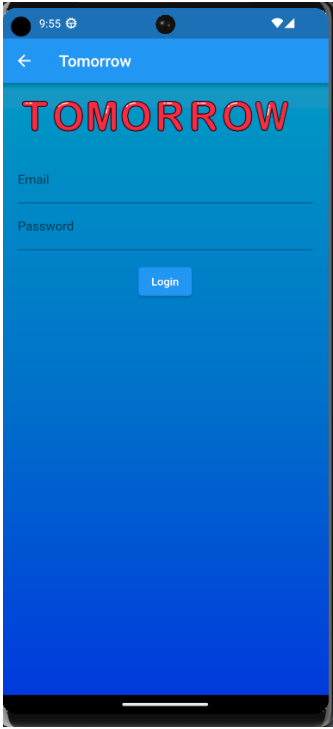


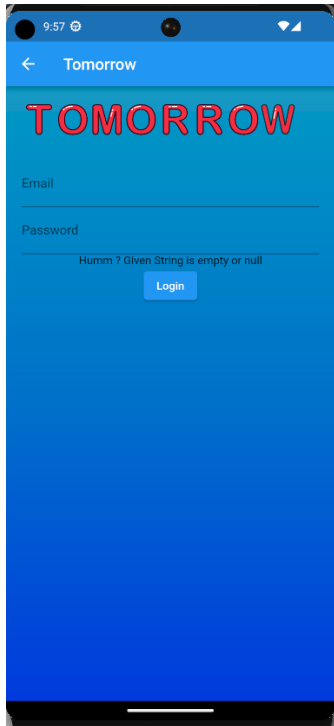
- ในกรณีที่จำนวนรหัสผ่านของผู้ใช้น้อยกว่า 6 ตัวอักษร ระบบจะทำการแจ้งข้อความเตือนว่ารหัสผ่านต้องมีอักขระอย่างน้อย 6 ตัว



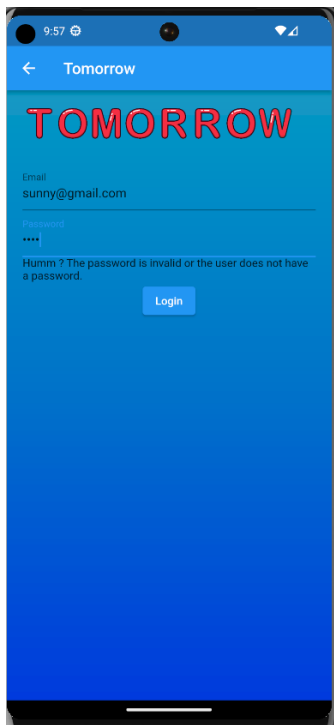
- สุดท้ายจะเชื่อมไปยังหน้านี้เมื่อผู้ใช้ลงทะเบียนเข้าสู่ระบบสำเร็จ เพื่อดำเนินการล็อกอินเข้าใช้งานต่อไป

6.2.1.2 Login ผู้ใช้ login เข้าสู่ระบบ

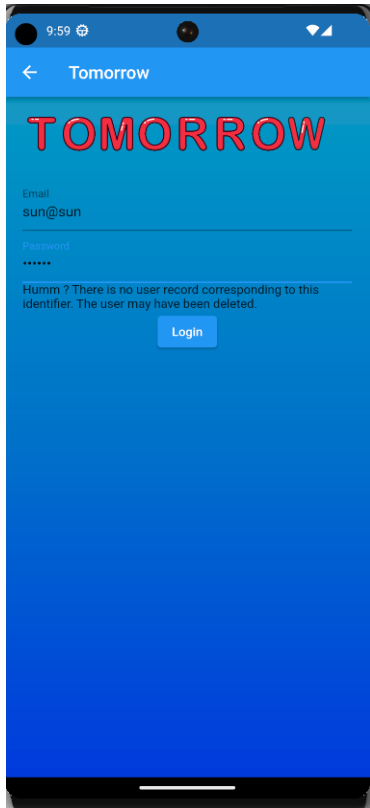
	<p>ขั้นตอนการเข้าสู่ระบบ สามารถเริ่มต้นด้วยผู้ใช้คลิกเลือกที่ปุ่ม log in ในหน้าก่อนหน้า</p> <ul style="list-style-type: none">- กรณีที่ผู้ใช้ได้ลงทะเบียบเสร็จจะสามารถใส่ข้อมูล(Ussername,Password)- ถ้าผู้ใช้ใส่ข้อมูลเสร็จแล้ว จะสามารถ เข้า "Login"
	<ul style="list-style-type: none">- เมื่อผู้ใช้มาถึงหน้าเข้าสู่ระบบ จะเห็นว่า มี 2 ช่องให้ใส่ คือ email กับ password- ใต้ช่องที่กรอกข้อมูล จะมีปุ่มให้เลือก คือ "Login"



- คล้ายกับหน้า register ระบบจะแสดงข้อความเตือนใต้ช่องว่าง หากผู้ใช้คลิก "Login" แล้วข้อมูลที่ใส่ยังไม่ครบถ้วน

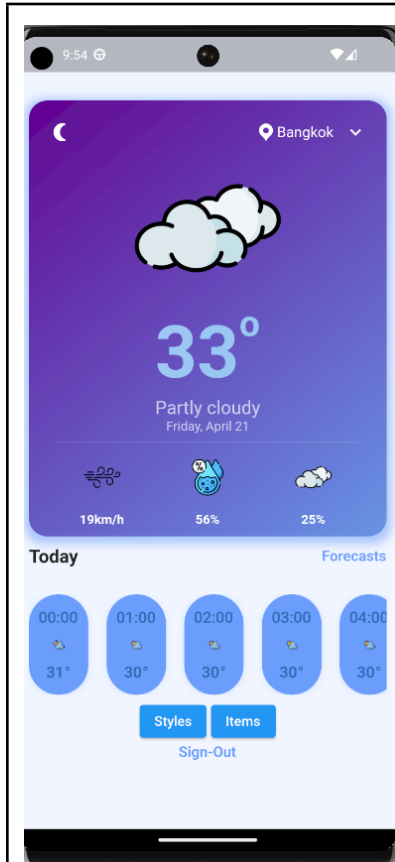


- กรณีผู้ใช้ใส่รหัสผิด ระบบจะทำการแจ้งเตือนข้อความว่ารหัสยังไม่ถูกต้อง



- หากผู้ใช้ใส่อีเมลผิดเงื่อนไขที่มีอยู่ในฐานข้อมูล ข้อความเตือนจะปรากฏด้านล่าง เพื่อแจ้งให้ผู้ใช้ใส่อีเมลให้ถูกต้อง

6.2.1.3 Weather ผู้ใช้ใช้ฟังก์ชัน Weather



- หลังจากเข้าสู่ระบบ ระบบจะนำผู้ใช้งานมาหน้านี้ซึ่งเป็นหน้าหลักที่แสดง อุณหภูมิ , ความเร็วลม , ความหนาแน่นของเมฆ รวมไปถึง อุณหภูมิในแต่ละช่วงเวลา
- ในหน้านี้จะมี 4 ปุ่ม คือ , "Items" ดูสิ่งที่จำเป็นในแต่ละสภาพภูมิอากาศ , "Styles" ดูการแต่งกายตามสภาพอากาศ , "Forecast" ทำนายสภาพอากาศล่วงหน้า 7 วัน และ "Sign-out" ออกจากระบบ
- ในกรณีนี้ ผู้ใช้ได้ทำการคลิกในส่วน "Styles"



- User จะเห็นหน้านี้หลังจากเลือก button "Styles"
- ในหน้านี้จะเป็นหน้าที่เราสามารถที่จะเลือกดู styles การแต่งตัวที่เหมาะสมกับสภาพอากาศได้ ซึ่งดังตัวอย่างจะเป็นสไตล์ของสภาพอากาศร้อน หรืออุณหภูมิสูง



- User จะเห็นหน้านี้หลังจาก เลือกปุ่ม "Items"
- ในหน้านี้จะเป็นหน้าที่เราสามารถที่จะเลือกดู items การแต่งตัวที่เหมาะสมกับสภาพอากาศได้ ซึ่งดังตัวอย่างจะเป็นสไตล์ของสภาพอากาศร้อน หรืออุณหภูมิสูง
-



- User จะเห็นหน้านี้หลังจาก เลือกปุ่ม "Forecast"
- User สามารถดูการทำนายสภาพอากาศล่วงหน้า ได้ 7 วัน

บทที่ 7

สรุปผล

บทนี้กล่าวถึงบทสรุปของโครงการ อธิบายถึงประโยชน์ ปัญหา ข้อจำกัด และมุมมองของผู้จัดทำต่อการทำงานในอนาคต

7.1 ประโยชน์ (Benefits)

หัวข้อนี้จะอธิบายถึงประโยชน์ของโครงการสำหรับผู้จัดทำและผู้อ่าน

7.1.1 ประโยชน์ต่อผู้พัฒนาโครงการ

- ได้เรียนรู้การเขียนภาษาที่หลากหลาย
- การทำงานเป็นทีม การสื่อสาร และพัฒนาแอปพลิเคชันร่วมกัน
- ได้ฝึกฝนสกิลการเขียนโค้ดเพิ่มมากขึ้น
- ฝึกฝนการแก้ปัญหาเล็กๆน้อยๆที่อาจมองข้าม
- เรียนรู้การใช้เครื่องมือ และฟังก์ชันที่เหมาะสม
- ได้ศึกษาเพิ่มเติมเกี่ยวกับการพยากรณ์สภาพอากาศ การดึง API ต่างๆ
- เรียนรู้การใช้ Flutter และ Dart ได้ชำนาญมากยิ่งขึ้น
- เรียนรู้การใช้ฐานข้อมูลใหม่ๆ คือ Firebase Firestore ให้มากขึ้น โดยได้รับความรู้มากมาย เช่น ข้อมูลเกี่ยวกับ Firestore, ฟังก์ชัน Authentication, วิธีเชื่อมต่อไฟล์โครงการกับ Firebase Firestore, และวิธีประมวลผลโปรแกรมและฐานข้อมูล
- ฝึกฝนการอ่าน Error และหาวิธีการแก้ไข

7.1.2 ประโยชน์ต่อผู้ใช้งาน

- ผู้ใช้สามารถเตรียมตัวรับมือกับสภาพอากาศในแต่ละวันได้
- ให้คำแนะนำด้านต่างๆเพิ่มเติมในแต่ละวัน เช่น การแต่งกาย หรือ ของที่ควรเตรียม
- ผู้ใช้สามารถเข้าถึงแอปและรับรู้การพยากรณ์ได้อย่างง่ายและมีความสะดวก
- สามารถดูการพยากรณ์ล่วงหน้าได้ 7 วัน สำหรับผู้ที่ต้องการแพลนหรือทราบสภาพอากาศล่วงหน้า
- ผู้ใช้สามารถเข้าถึงได้ง่าย เพราะแอปของเราใช้งานง่าย สะดวก และสรุปให้กระชับสามารถดูผลแต่ละวันได้อย่างง่าย ไม่หลายขั้นตอน

7.2 ปัญหาและข้อจำกัด

- โปรแกรมใช้ทรัพยากรจำนวนมากในคอมพิวเตอร์ ทำให้เกิดอาการกระตุกอยู่บ้างขณะรันโปรแกรม
- ความไม่สอดคล้องบางประการเมื่อเรียกใช้โปรแกรม ในการรันบางครั้งอาจมีข้อผิดพลาดแทรกซ้อน
- การดึง API ที่ฟรีมีข้อจำกัด ต้องอยู่ในระยะเวลาที่สามารถใช้ได้เท่านั้น หากเกินอาจต้องชำระเงินเพิ่มเติม
- การเชื่อมต่อหน้าให้เชื่อมโยงจากการแบ่งงานกันทำ ทำให้พบปัญหาเล็กน้อยที่ต้องแก้บ่อยครั้ง

7.3 สิ่งที่ต้องทำต่อไป

- UX/UI จะมีความสั่นไหวและเหมาะสมในการใช้งานมากขึ้น เพื่อดึงดูดและให้ผู้ใช้รู้สึกพึงพอใจขณะใช้งานแอปพลิเคชัน
- จัดรูปแบบให้มีความหลากหลายและดูง่ายมากขึ้น
- เพิ่มการแนะนำให้มีความหลากหลายมากขึ้นทั้งในด้านของเสื้อผ้า หรือ ไอเทม
- เพิ่มการพยากรณ์ล่วงหน้าเป็น 1 เดือน
- เพิ่มเนื้อหาให้มีความสมจริงและสะดวกในการใช้งาน
- ในส่วนที่มีปัญหาหรือการแสดงผลต่างๆ จะได้รับการปรับปรุงแก้ไขให้ข้อมูลมีความทันสมัย สั่นไหว และไม่กระตุก

บทที่ 8

สรุปความก้าวหน้าของโครงการ

เราได้เสร็จสิ้นขั้นตอนต่างๆ ของโครงการเรียบร้อยแล้ว ตามรายงานความคืบหน้าของเรา การอภิปรายเกี่ยวกับข้อกำหนดที่ต้องปฏิบัติตามกระบวนการที่ใช้ในการเลือกหัวข้อ และข้อกำหนดทั้งหมดจะครอบคลุมในขั้นแรก เมื่อพูดถึงการวางแผน เราได้มีการพูดคุยและกำหนดเวลา ว่าแต่ละองค์ประกอบโครงการควรใช้เวลาานเท่าใดจึงจะเสร็จสิ้น ซึ่งเป็นไปอย่างราบรื่น และเมื่อมีเหตุการณ์ไม่คาดฝันเกิดขึ้นกลุ่มของเราได้มีการพูดคุย ปรึกษากัน เป็นผลให้เราดำเนินการแต่ละขั้นตอนที่จำเป็นได้สำเร็จ ครั้งหลังของโครงการต้องใช้ความเร่งรีบอย่างมาก เพื่อให้งานเสร็จในเวลาอันสั้นที่สุด เราได้เตรียมการและพูดคุยรายละเอียดในทุกแง่มุม กล่าวโดยสรุปได้ว่าโครงการของเราประสบความสำเร็จในระดับที่สมาชิกในกลุ่มทุกคนพึงพอใจ

REFERENCES

=<https://www.mindphp.com/forums/viewtopic.php?t=79921>

=<https://www.somkiat.cc/hello-cloud-firestore/>

=<http://www.arduino-makerzone.com/article/54/firebase-basic-ep1-firebase-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3>

=<https://medium.com/firebaseethailand/%E0%B9%80%E0%B8%82%E0%B9%89%E0%B8%A1%E0%B8%82%E0%B9%89%E0%B8%99%E0%B8%81%E0%B8%B1%E0%B8%9A-firebase-cloud-firestore-%E0%B8%A3%E0%B8%B0%E0%B8%9A%E0%B8%90%E0%B8%B2%E0%B8%99%E0%B8%82%E0%B9%89%E0%B8%AD%E0%B8%A1%E0%B8%B9%E0%B8%A5%E0%B8%97%E0%B8%B5%E0%B9%88%E0%B9%80%E0%B8%9B%E0%B8%B4%E0%B8%94%E0%B8%95%E0%B8%B1%E0%B8%A7%E0%B9%83%E0%B8%AB%E0%B8%A1%E0%B9%88%E0%B8%A5%E0%B9%88%E0%B8%B2%E0%B8%AA%E0%B8%B8%E0%B8%94%E0%B8%88%E0%B8%B2%E0%B8%81-firebase-%E0%B9%81%E0%B8%9A%E0%B8%88%E0%B8%B1%E0%B8%94%E0%B9%80%E0%B8%95%E0%B9%87%E0%B8%A1-d001e43e2be7>

=[รู้จัก Firebase Authentication ตั้งแต่ Zero จนถึง Hero | by Jirawatee | 🌟 Firebase Thailand | Medium](#)

=[\[Firebase\] คืออะไร มาดูวิธีสร้าง Project และทำความเข้าใจกับ Firebase | by Jedsada Saengow | JED-NG | Medium](#)

BIOGRAPHIES

ชื่อ

ปีที่เข้าศึกษา

นายจิรภัทร สุวรรณมัย

โรงเรียนโพธิสารพิทยากร , 2021:

High School Diploma

Mahidol University, 2025:

Bachelor of Science (DST)

ชื่อ

ปีที่เข้าศึกษา

นางสาว พรสพษา อาริยวงศ์สถิตย์

โรงเรียนนครสวรรค์ , 2021:

วุฒิการศึกษา มัธยมศึกษา

มหาวิทยาลัยมหิดล, 2025:

Bachelor of Science (DST)

ชื่อ

ปีที่เข้าศึกษา

นางสาว รวิพชัญ เล้าแสงชัยวัฒน์

โรงเรียนสตรีวิทยา, 2020:

วุฒิการศึกษา มัธยมศึกษา

มหาวิทยาลัยมหิดล, 2025:

Bachelor of Science (DST)

LINK OF GITHUB OF THE PROJECT

<https://github.com/6487019Jirapat/tomorrow>