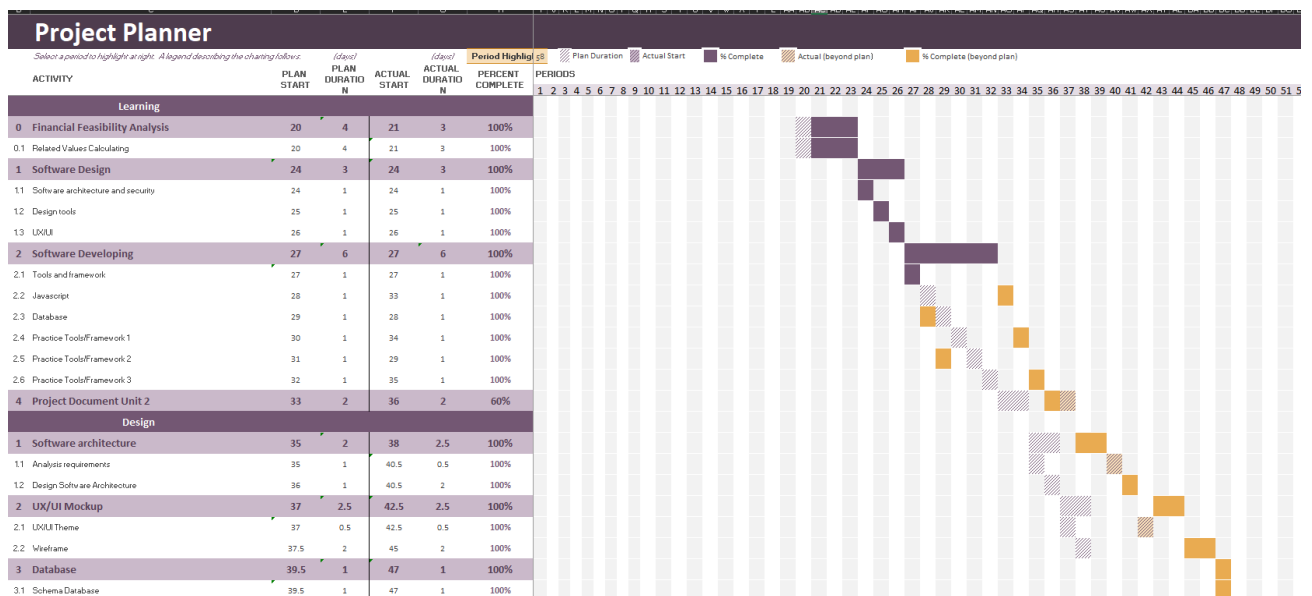


รายงานความก้าวหน้าวิชา Computer Engineering Project Preparation

ครั้งที่ 3

1. ชื่อโครงการ (อังกฤษ) Analytics system for starting business
2. การดำเนินงานมีความก้าวหน้า 82%
3. ความก้าวหน้าระหว่างวันที่ 01 เม.ย. 65 ถึงวันที่ 22 เม.ย. 65
4. รายละเอียดความก้าวหน้า

แผนการพัฒนาซอฟต์แวร์ Phase 3 ได้ดำเนินการตามกำหนดการ Gantt Chart ดังนี้



รูปที่ 1 ผลการทำงานของทีมงานตาม Gantt Chart ปัจจุบัน

จากรายงานความคืบหน้าครั้งที่แล้วที่มีการเก็บ user requirements และทำ functional requirement ตามกำหนดการตามแผนใน Gantt Chart ส่วนรายงานความคืบหน้าในครั้งนี้จะกล่าวถึงการพัฒนาใน phase 3 ของโปรเจก Financial Analysis คือ การศึกษาค้นคว้าและทำการออกแบบส่วนต่างๆตามแผนงานที่กำหนดไว้

ตามแผนการพัฒนา ทีมได้ทำการศึกษาค้นคว้าเพื่อให้เกิดความเข้าใจ ก่อนทำการออกแบบส่วนต่างๆในขั้นตอนต่อไป ดังนี้

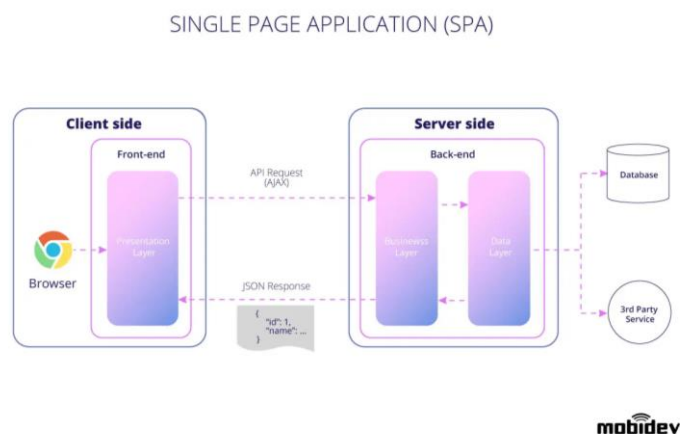
1. Software Design

1.1 Software Architecture and security

ทางทีมได้ทำการศึกษาค้นคว้าข้อมูลเพื่อที่จะออกแบบ Software Architecture ตามขอบเขตดังนี้

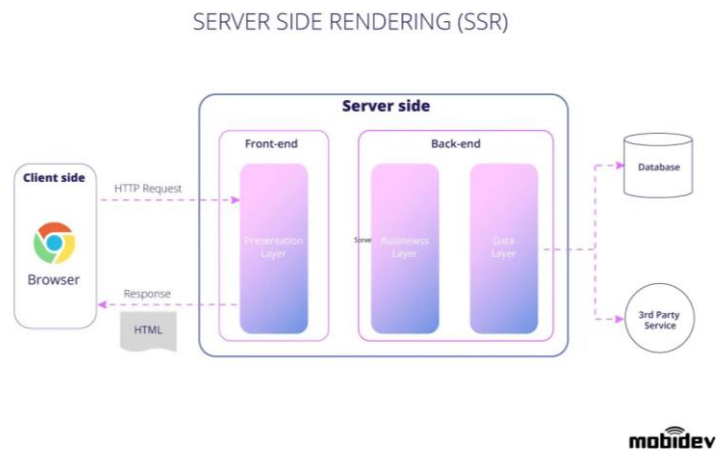
Web Application Architecture

จากที่ทีมได้เลือกใช้ React framework ในการพัฒนา Web Application ด้วยภาษา JavaScript จึงได้ทำการศึกษาค้นคว้า Web Application Architecture พบว่ามีโมเดลที่ทางทีมเห็นว่าเหมาะสม ดังนี้



รูปที่ 2 Single Page Application Architecture (SPA)

Single Page Application Architecture (SPA) นั้นง่ายต่อการพัฒนา มีความ responsive สูงมากๆ ลดช่องว่างของการรอระหว่างเปลี่ยนหน้า ซึ่งเกิดจากการรอ response จาก server หรือการเก็บ Cache ข้อมูลให้โหลดข้อมูลได้เร็ว แต่มีข้อเสียคือ ในการเปิดหน้าเว็บครั้งแรกจะใช้เวลาในการให้ Client Browser render ขึ้นอยู่กับขนาด HTML ของเรา และทำ SEO ไม่ค่อยมีประสิทธิภาพ



รูปที่ 3 Server side Rendering Architecture (SSR)

Server side Rendering Architecture (SSR) นั้นเหมาะกับการทำ SEO และใช้เวลาในแสดงผลครั้งแรกไม่นาน เพราะให้ server เป็นคน render แทนและใช้ development cost ในการพัฒนาต่ำ ข้อเสียคือ การพัฒนานั้นมีความซับซ้อนและการเกิดการ Coupling กันได้ และในบางเฟรมเวิร์ค จำเป็นต้องมีการ config ผ่าน framework อื่นอีกที เช่น React ใช้ ASP.NET CORE ในการทำ SSR ได้ เป็นต้น

แหล่งข้อมูล:

<https://mobidev.biz/blog/web-application-architecture-types>

<https://stackify.com/web-application-architecture/>

<https://litslink.com/blog/web-application-architecture>

1.2 Design Tools

ทางทีมได้ทำการเลือกใช้โปรแกรม Figma ในส่วนของการออกแบบ UX/UI เนื่องจากสามารถแชร์ และ แก้ไขไฟล์ร่วมกันได้แบบเรียลไทม์เหมาะแก่การทำงานเป็นทีม



รูปที่ 4 เครื่องมือช่วยออกแบบ UX/UI Figma

1.3 UX/UI

UX/UI นั้นเป็นส่วนประกอบหนึ่งที่สำคัญในการพัฒนาซอฟต์แวร์ เนื่องจากมีผลโดยตรงต่อความรู้สึกของผู้ใช้ หากเราออกแบบได้ไม่ดีพอ ทำให้ใช้งานได้ยากอาจส่งผลทำให้จำนวนผู้ใช้งานลดลงได้ ในทางกลับกันหากเราดีไซน์ออกมาได้ดีก็จะสามารถดึงดูดผู้ใช้ ทั้งนี้เราจึงต้องออกแบบโดยคำนึงถึงกลุ่มเป้าหมายเป็นหลัก และยังต้องอัปเดตเทรนในปัจจุบันด้วย

UX หรือ User Experience คือ ประสบการณ์ของผู้ใช้งานในด้านความรู้สึกที่ตอบสนองต่อการใช้งานผลิตภัณฑ์ หรือระบบต่าง ๆ ยกตัวอย่างเช่น ความสะดวกสบาย ใช้งานง่าย ความสนุกสนาน จนเกิดเป็นความพึงพอใจสูงสุดหรือเกิดประสบการณ์ที่ดีของผู้ใช้งานนั่นเอง

UI หรือ User Interface คือ ส่วนต่อประสานกับผู้ใช้งาน หรือ ส่วนที่ใช้ในการเชื่อมต่อกับผู้ใช้งาน กล่าวคือ ส่วนที่ให้ผู้ใช้งานสามารถโต้ตอบกับการใช้งานผลิตภัณฑ์ได้ ซึ่งจะมุ่งเน้นไปที่เรื่องของหน้าตา การออกแบบ และการดีไซน์ ยกตัวอย่างเช่น หน้าจอ แพลตฟอร์ม เมนู ฟอรัมต่าง ๆ การวางภาพ ขนาด ตัวอักษร ปุ่ม แป้นพิมพ์ เสียง หรือแม้แต่แสงไฟ เป็นต้น

แหล่งข้อมูล:

<https://maze.co/collections/ux-ui-design/ux-examples/>

2. Software Development

2.1 Tools and frameworks

จากที่ทีมได้เลือกใช้ React framework ในการพัฒนาซอฟต์แวร์ด้วยภาษา JavaScript จึงได้ทำการศึกษาค้นคว้า Tech Stack สำหรับ AI Based React Web Application ดังนี้

- React Framework สำหรับการพัฒนาเว็บแอปพลิเคชัน
- MongoDB สำหรับ NoSQL Database
- Docker สำหรับ Microservice infrastructure ช่วยในการสร้าง แชนจ์ และติดตั้งโปรแกรม
- Node.js/express สำหรับเป็น Backend
- NextJS ในการทำ SSR Server

แหล่งข้อมูล:

<https://medium.com/i-gear-geek/docker-คือ-เริ่มต้นกับ-docker-56d0ba499ae8>

<https://stackshare.io/stackups/nestjs-vs-nextjs>

<https://blog.jakoblind.no/getting-started-react-ssr/>

<https://blog.saeloun.com/2022/01/20/new-suspense-ssr-architecture-in-react-18>

2.2 Javascript

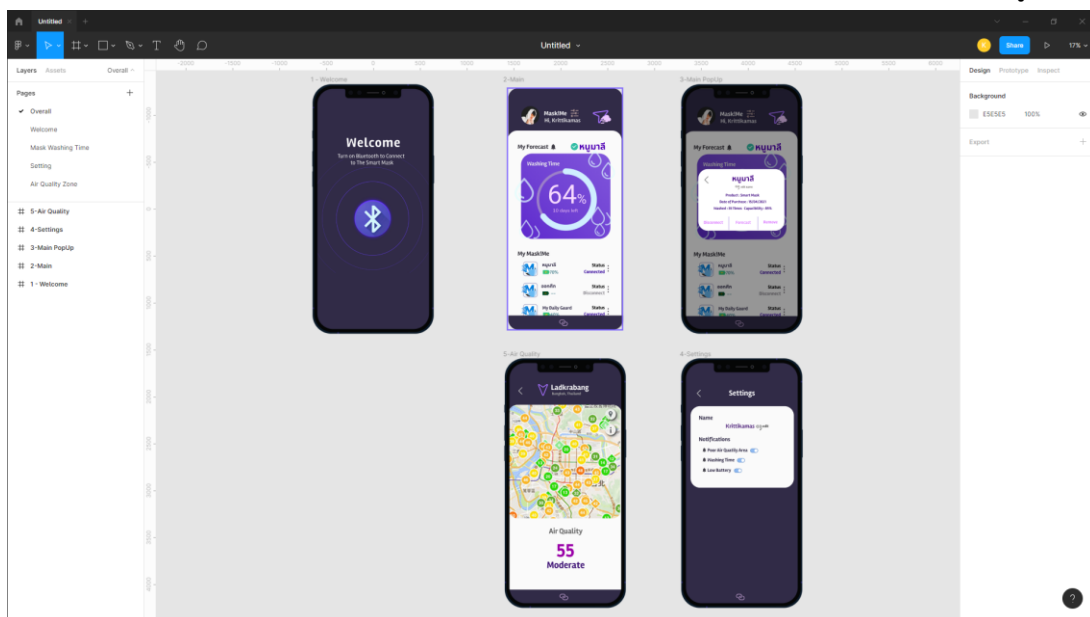
เนื่องจากทางเราต้องการใช้ React framework ซึ่งเป็น JavaScript Library ที่พัฒนาขึ้นมา และได้รับความนิยมอย่างมากมาย รูปแบบการใช้ก็ยังคงใช้ HTML CSS และ JavaScript ส่วนที่เพิ่มเติมคือสามารถแบ่งเว็บให้เป็นส่วนๆ เรียกกันว่า Component โดยจะเน้นรูปแบบที่เป็น Dynamic มากขึ้น คือมีการนำรูปแบบของ JSX มาใช้งานที่ฝั่งหน้าเว็บเลย ทำให้สามารถแทนค่าตัวแปรต่างๆ บนหน้าเว็บได้เลย

2.3 Database

ในส่วนของ database ทางผู้พัฒนาได้เลือกใช้ MongoDB ซึ่งเป็น NoSQL Database เนื่องจากต้องการความรวดเร็วในการส่งข้อมูลและเพื่อการเปลี่ยนโครงสร้าง ขยับขยายได้ในอนาคต จากวัตถุประสงค์ของโครงการ ซึ่งมีโอกาสที่จะเปลี่ยนแปลงข้อมูลในแต่ละส่วนสูง โดยเฉพาะการปรับข้อมูลธุรกิจให้สอดคล้อง นำใช้ตามยุคสมัย

2.4 Practice Tools/Framework 1 : Figma Design Tools

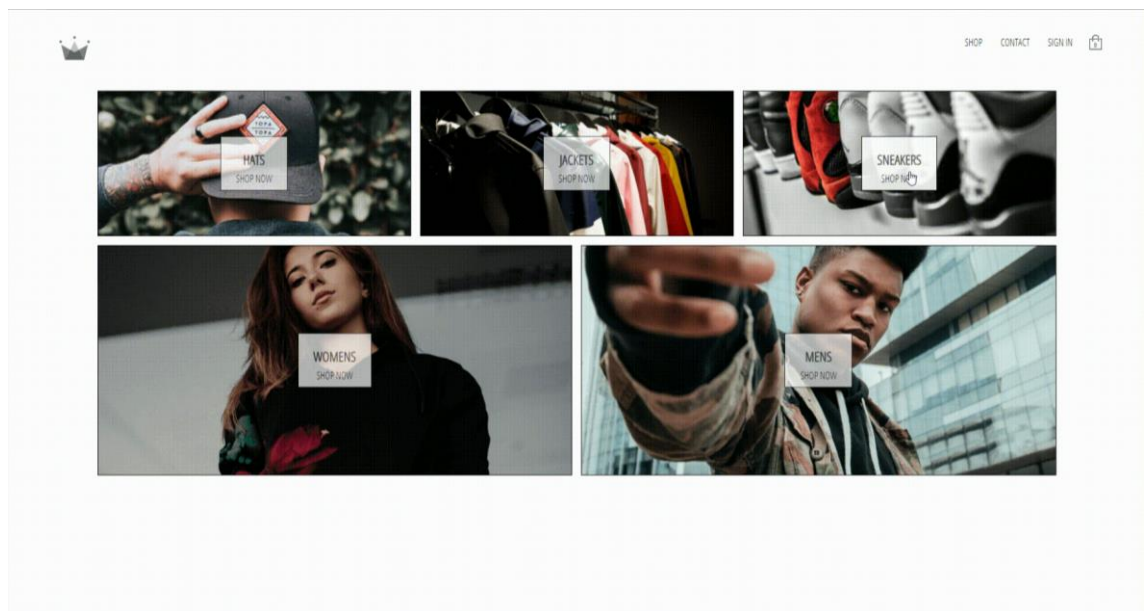
ทางทีมได้ลองใช้ Figma ในการออกแบบและทำ Page Navigation Prototype ดังรูป



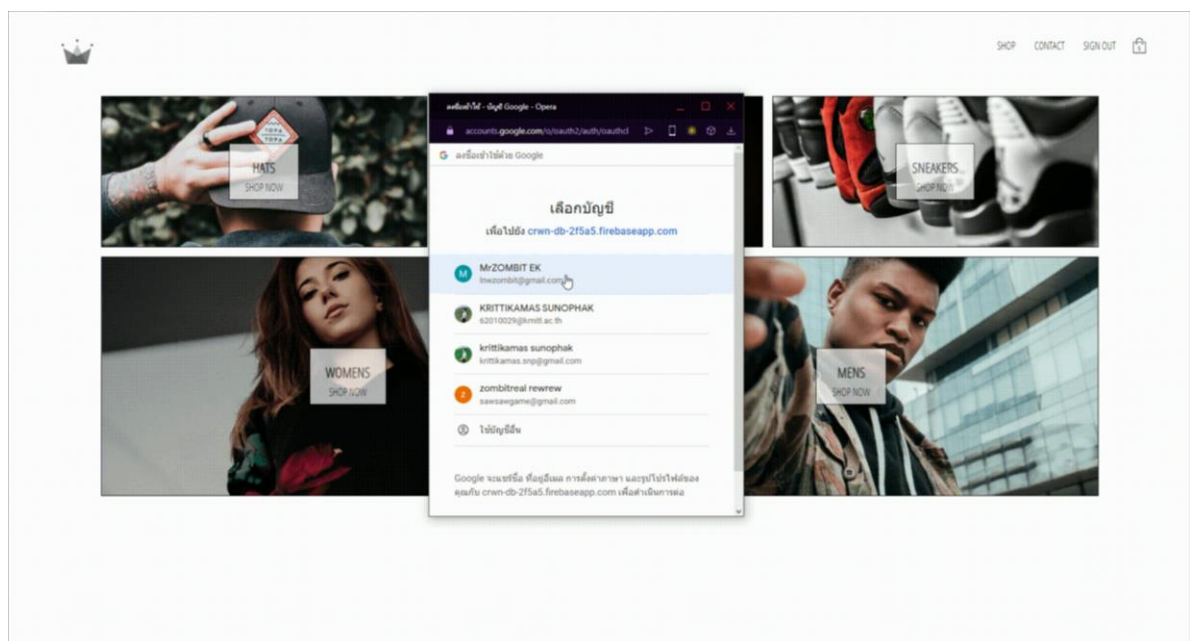
รูปที่ 5 ภาพโปรเจกต์ที่ฝึกใช้ Figma

2.5 Practice Tools/Framework 2 : React

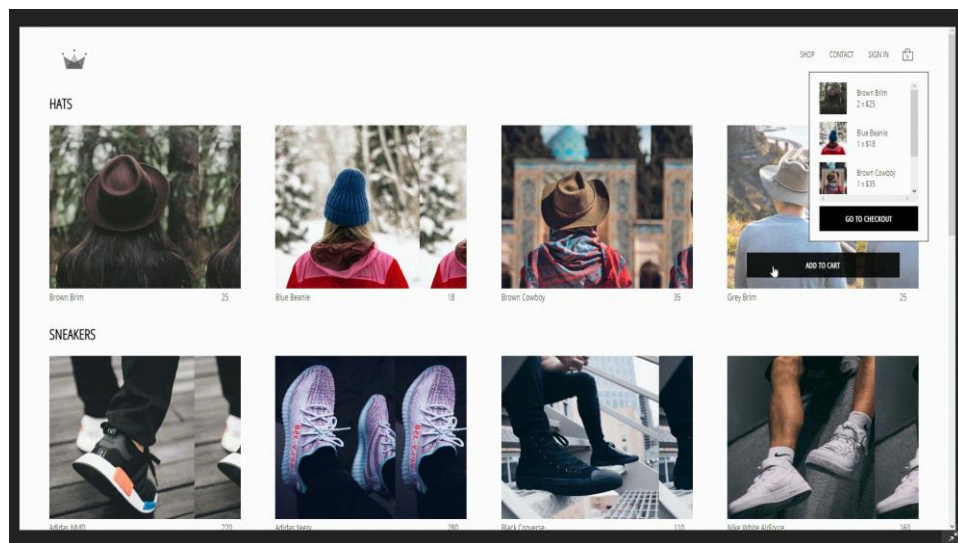
ในส่วนของการศึกษาก่อนมีการเปลี่ยนแปลง technology ที่ใช้ใน project ได้มีการศึกษาการใช้ React framework ด้วยการสร้างมินิโปรเจกต์ Ecommerce ดังนี้



รูปที่ 6 ภาพหน้า Home ของร้าน Ecommerce



รูปที่ 7 ภาพการ Sign in ด้วย Google account



รูปที่ 8 ภาพในส่วนของตะกร้า

แหล่งศึกษา :

<https://www.udemy.com/course/complete-react-developer-zero-to-mastery/>

3. Project Document Unit 2

ทางทีมได้จัดทำรูปเล่มโครงงานบทที่ 2 ไปแล้วบางส่วน

3. ทฤษฎีที่เกี่ยวข้อง (Theoretical Background)

3.1 React



React คือ JavaScript Library อีกตัวหนึ่ง ซึ่งมีหน้าที่สร้างงานทางด้าน Front-end และ Back-end โดยลักษณะของ React จะแบ่งส่วนของการแสดงผลออกเป็นหลายๆส่วน หรือหลายๆไฟล์ก็พูดได้ เช่น ส่วนของ header, footer เป็นต้น โดยจะเรียกส่วนต่างๆที่แยกย่อยนี้กันว่า Component (คอมโพเนนต์) และเมื่อจะใช้งานก็จะนำ Component ต่างๆมาประกอบร่างกัน และแสดงผลออกมา

React มี 3 Concept คือ

Component — ส่วนต่าง ๆ ในเว็บเราจะมองเป็น Component

รูปที่ 9 ภาพตัวอย่างรูปเล่มบทที่ 2

Design

ตามแผนการพัฒนา หลังจากทีทีมได้ทำการศึกษาค้นคว้าเพื่อให้เกิดความเข้าใจแล้ว จึงทำการออกแบบส่วนต่างๆของซอฟต์แวร์ ดังนี้

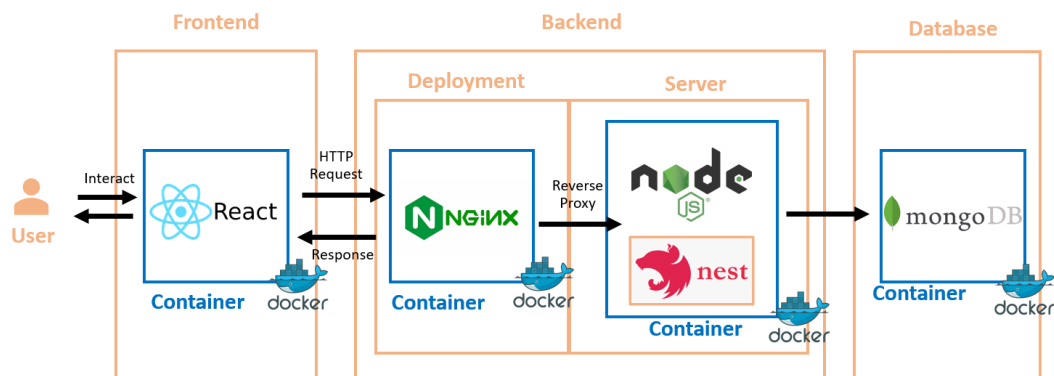
1. Software Architecture

1.1 Analysis Requirements

จากการวิเคราะห์ requirement ของระบบทั้งหมดระบบของเรา ซึ่งตามวัตถุประสงค์ที่จะนำไปใช้ในเชิงธุรกิจ ทางทีมจึงให้ความสำคัญกับการทำ SEO ดังนั้น Software Architecture ที่เลือกคือ SSR

1.2 Design Software Architecture

โครงสร้างสถาปัตยกรรมซอฟต์แวร์ประเมินความเสี่ยงของธุรกิจ เป็นดังนี้



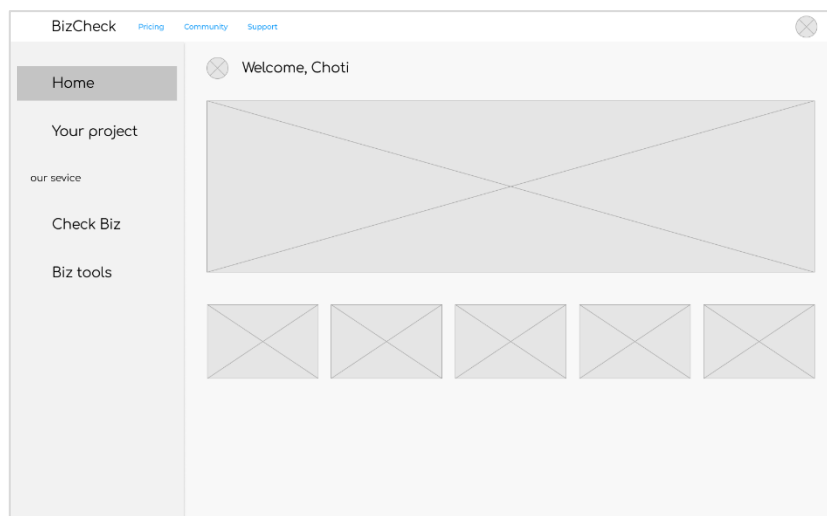
รูปที่ 10 Software Architecture

2. UX/UI

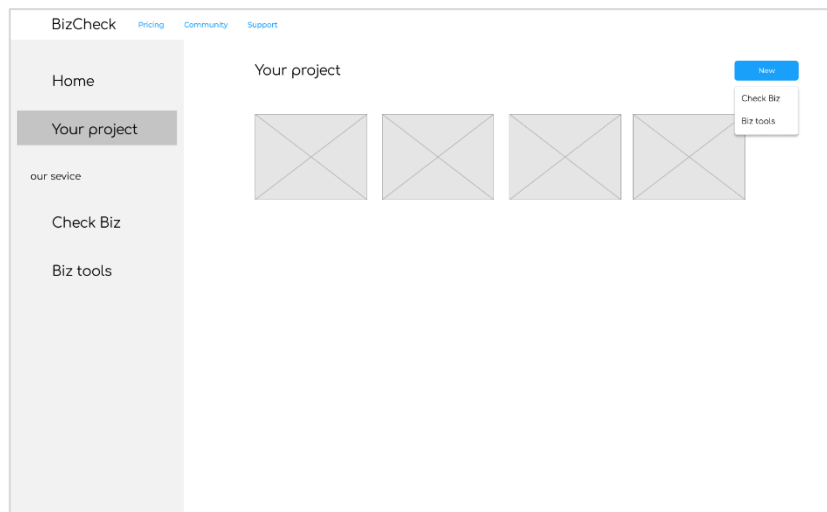
2.1 UX/UI Theme

ธีมสีขาวเป็นหลักเนื่องจากเหมาะกับผู้ใช้หลายประเภทและกลุ่มเป้าหมาย

2.2 Wireframe



รูปที่ 11 home page



รูปที่ 12 your project page

BizCheck Pricing Community Support Log in Register

Ready to take a free trial?

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore

Sign up for a free account

First name Last name

Email address

Create password

Register

รูปที่ 13 check bit page

BizCheck Pricing Community Support New

Home

Your project

our service

Check Biz

Biz tools

Check Biz

รูปที่ 14 check bit – in progress

BizCheck Pricing Community Support

Home

Your project

our service

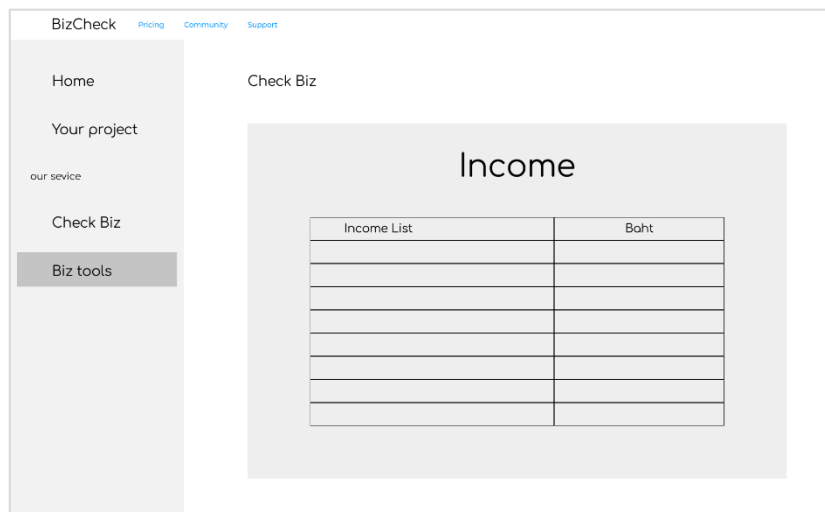
Check Biz

Biz tools

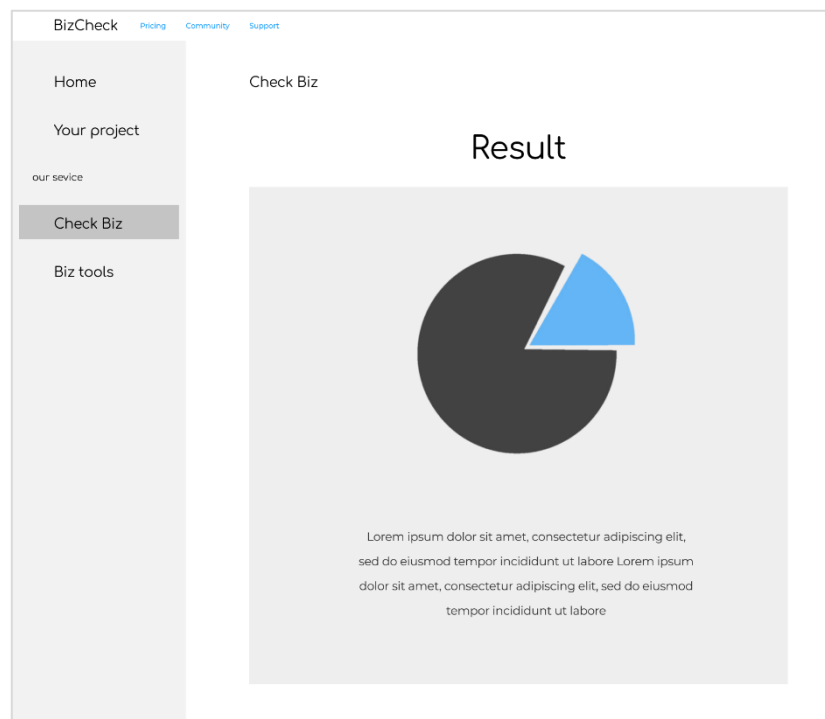
Check Biz

Question 1

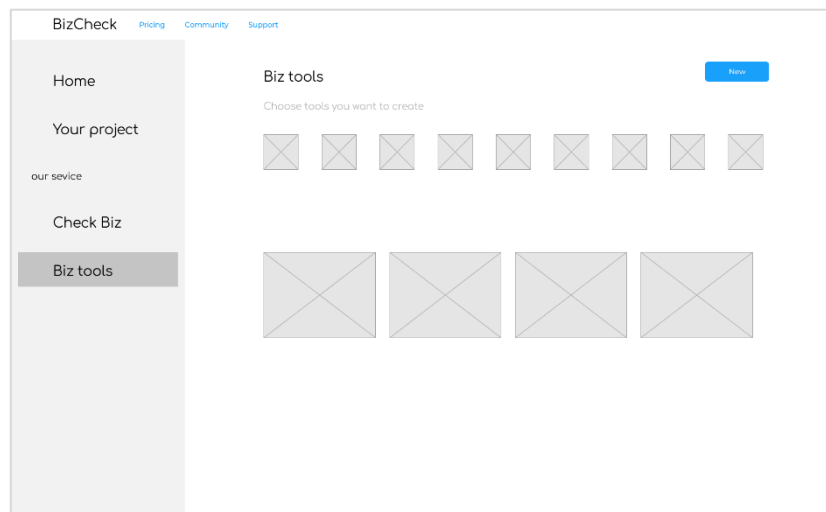
รูปที่ 15 check bit – in progress



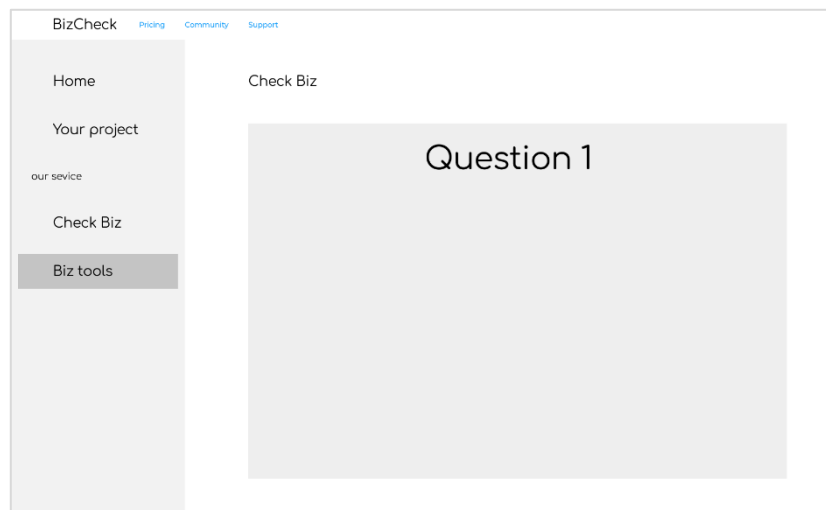
รูปที่ 16 check bit – in progress



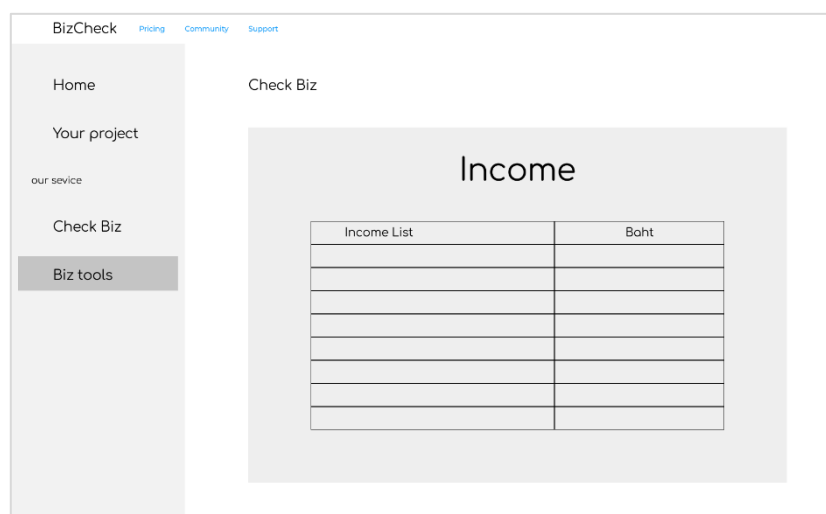
รูปที่ 17 check bit – result



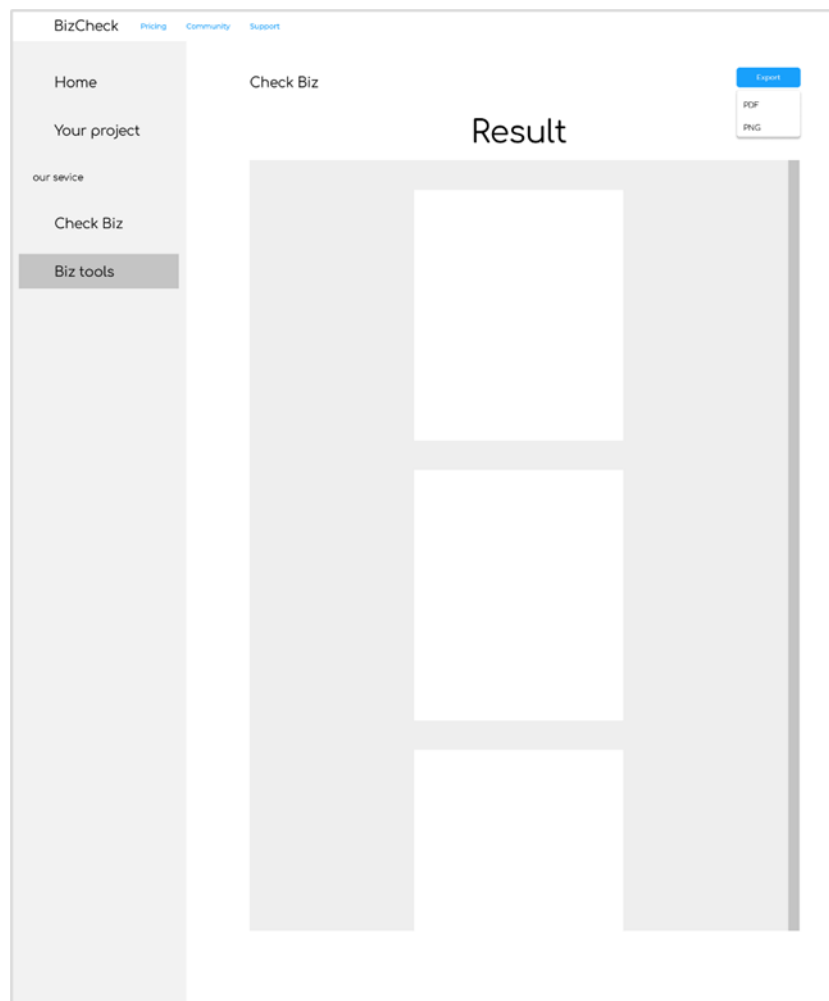
รูปที่ 17 bit tools



รูปที่ 18 bit tools – in progress



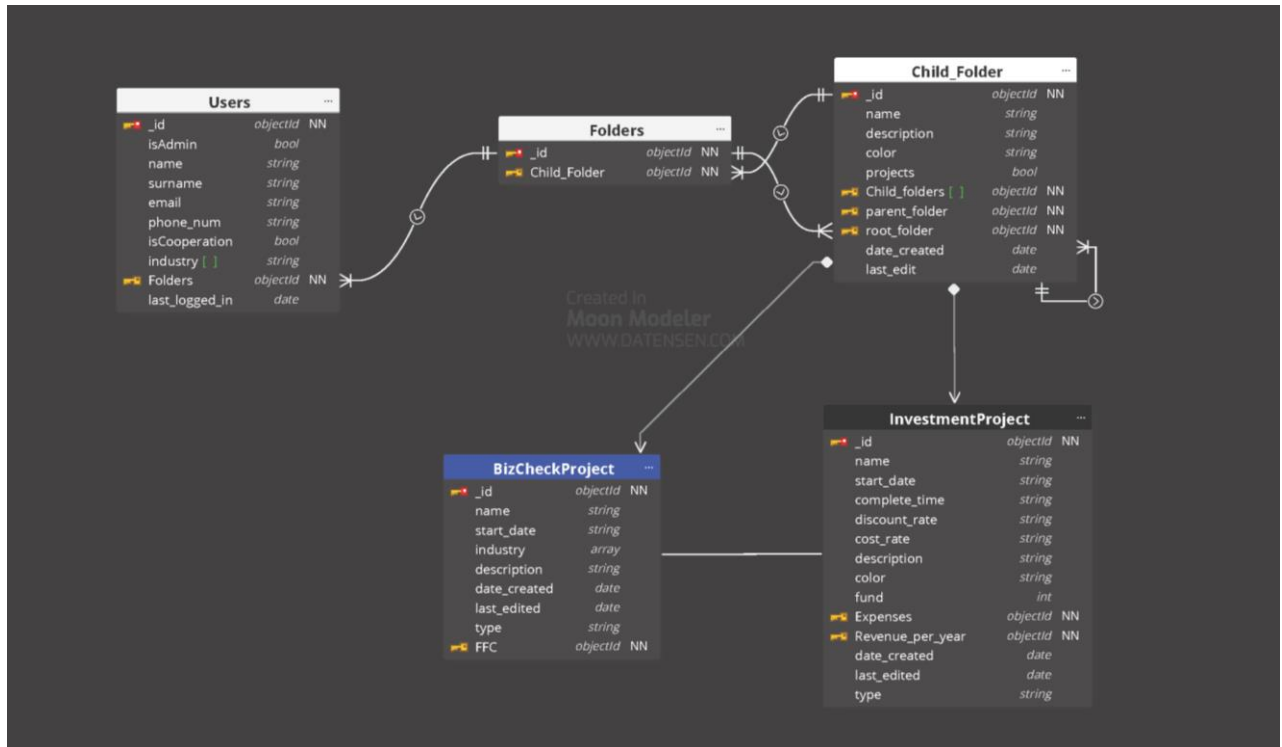
รูปที่ 19 bit tools – in progress



รูปที่ 20 bit tools – result

3. Database

3.1 Schema Database

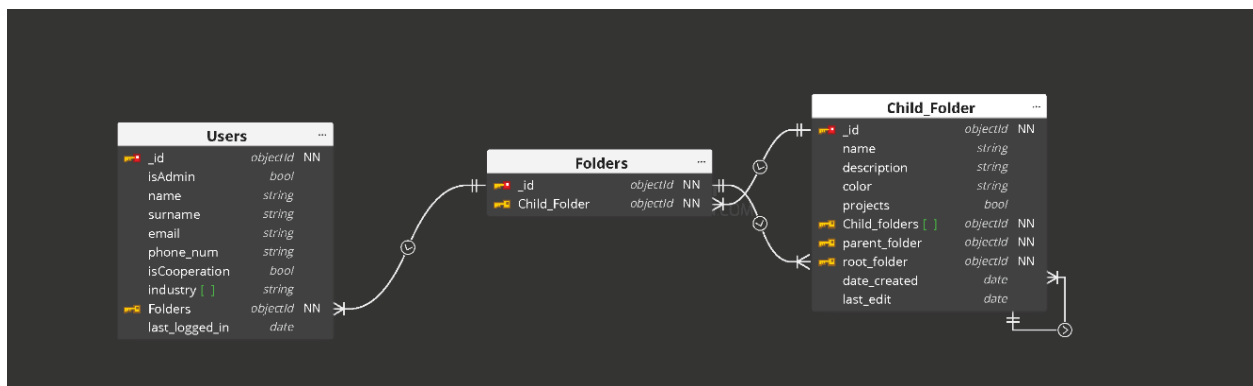


รูปที่ 21 ภาพรวมของ Database Schema

จากการวิเคราะห์การทำงานของซอฟต์แวร์ ทางทีมได้ออกแบบ Schema หลักสำหรับ Database แบ่งออกเป็น 4 Collections หลักได้แก่ Users ที่จะเป็นคนสร้างโปรเจก InvestmentProject และนำเข้า InvestmentProject มาใช้ฟังก์ชันประเมินความเสี่ยงของธุรกิจสร้างโปรเจก BizCheckProject โดยมีรายละเอียดดังนี้

1. Users เป็น collections ที่เก็บข้อมูลของผู้ใช้แต่ละคนและโฟลเดอร์เก็บโปรเจกที่สร้างขึ้น

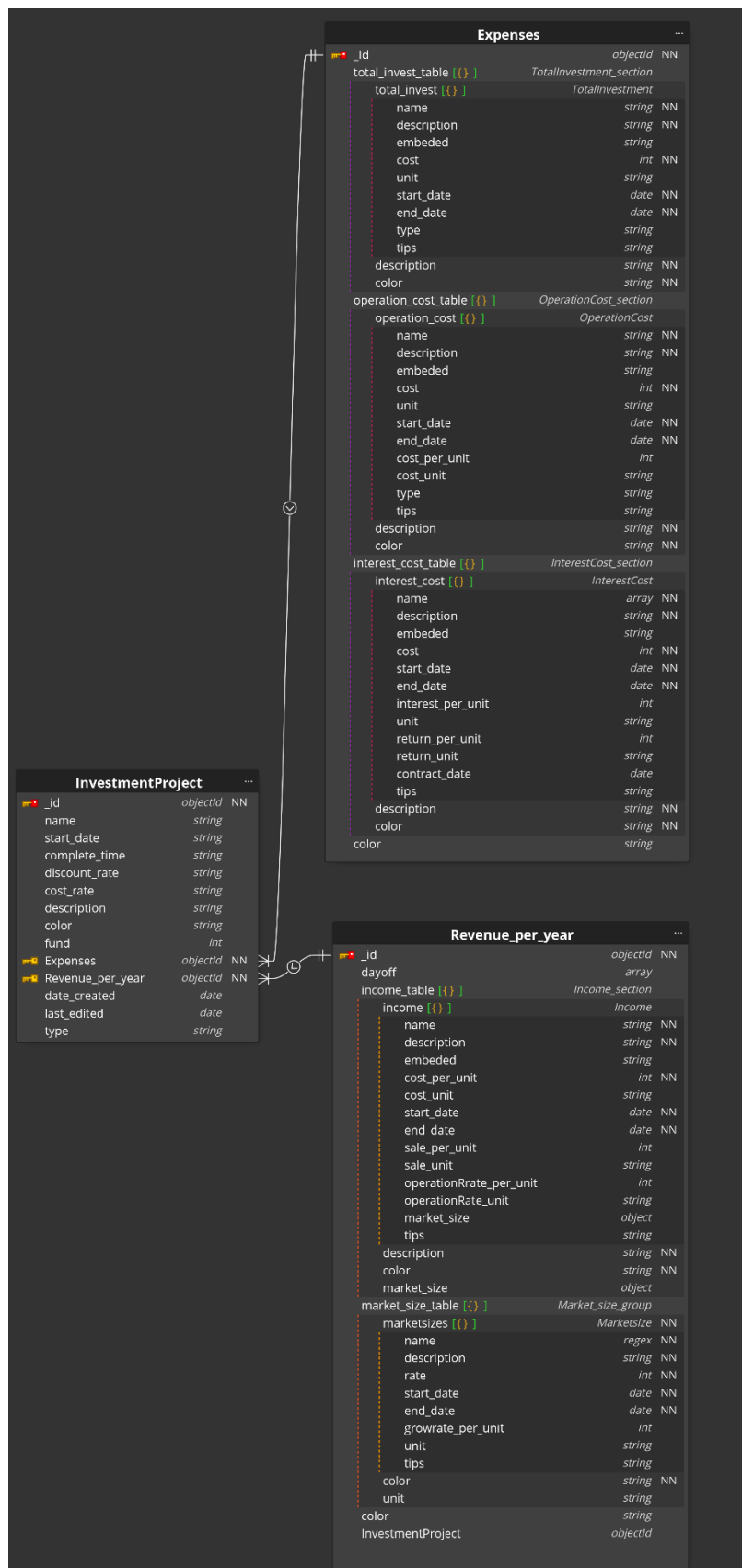
- Users ประกอบด้วยข้อมูลผู้ใช้หรือข้อมูลบริษัท (ถ้า isCooperation เป็น true) โดยแต่ละ account มีโฟลเดอร์สำหรับเก็บโปรเจกที่สร้างขึ้น
- Folders เป็น root โฟลเดอร์ จะคอยเก็บ child folder ที่สืบทอดเป็น node tree
- Child_folder เป็นโฟลเดอร์เก็บโปรเจกที่สร้างขึ้น



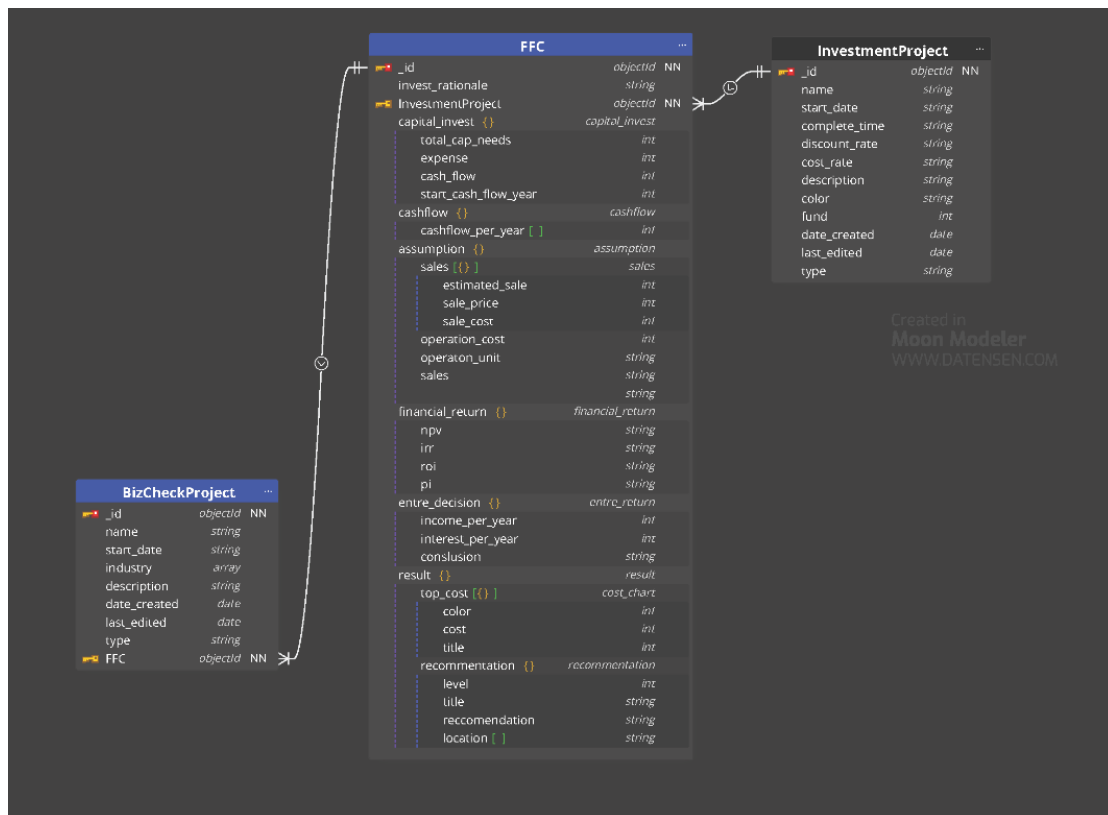
รูปที่ 22 User Collection ของ Database Schema

2. InvestmentProject เป็น collection แทนโปรเจกต์คำนวณเอกสารทางการเงินที่สร้างขึ้นผ่านฟังก์ชัน Biz tool

- InvestmentProject เป็นโปรเจกต์ที่สร้างขึ้นมาเพื่อคำนวณเอกสารทางการเงิน โดยจะมี collection อีก 2 ส่วนคือ Expenses (รายจ่าย) และ Revenue_per_year(รายได้ต่อปี)
- Expenses รายจ่ายที่เกิดขึ้นในการทำธุรกิจทั้งหมดตลอดระยะเวลาโครงการ ประกอบไปด้วย Total invest (เงินลงทุน), Operation cost (ค่าใช้จ่ายในการดำเนินการ) และ Interest cost (ค่าดอกเบี้ยและเงินกู้)
- Revenue_per_year เป็นรายได้ต่อปีตลอดระยะเวลาโครงการ จะมีทั้งส่วนรายได้(Income) และ ขนาดตลาด (Marketsize)



รูปที่ 23 InvestmentProject Collection ของ Database Schema



รูปที่ 24 BizCheck Collection ของ Database Schema

3. BizCheckProject เป็น collection แทนโปรเจกต์สำหรับประเมินความเป็นไปได้ทางการเงินของธุรกิจตามหลัก Financial Feasibility Canvas

- BizCheckProject เป็นโปรเจกต์ที่สร้างขึ้นสำหรับประเมินความเป็นไปได้ทางการเงินของธุรกิจตามหลัก Financial Canvas โดยจะมี FFC หรือ Financial Feasibility Canvas ของตัวโปรเจกต์หรือธุรกิจที่นำมาประเมิน
- FFC เป็น Financial Feasibility Canvas ประกอบไปด้วย 6 ส่วนคือ Capital Invest, Cashflow, Assumption, Financial return, Entrepreneurial Decision และ Result ผลลัพธ์ของการประเมิน โดยในการคำนวณจะ import InvestmentProject จากฟังก์ชัน Biz tool เพื่อมาใช้ในการคำนวณในการประเมินความเป็นไปได้ของธุรกิจ

5. ปัญหาที่เกิดขึ้นและแนวทางการแก้ไข

Problem No. 1

พบปัญหาในการรายงานความก้าวหน้า ครั้งที่ 3

สถานะ ☐ กำลังดำเนินการ ☒ แก้ไขสำเร็จ

รายละเอียดปัญหา

ในส่วนของฟังก์ชันประเมินความเสี่ยง เนื่องจาก Dataset ที่จำเป็นต้องใช้ในการประเมินนั้นมีน้อยและหายากเนื่องจากเป็นเอกสารลับทางการค้าของแต่ละธุรกิจ รวมไปถึงการรับข้อมูลเข้าที่คลุมเครือ ซึ่งมีผลต่อความน่าเชื่อถือของผลลัพธ์

แนวทางแก้ไข/การแก้ไข

ทางทีมได้ตัดสินใจกับอาจารย์ที่ปรึกษาว่าควรยกเลิกการใช้ AI ในฟังก์ชันประเมินความเสี่ยงในการทำธุรกิจ

Problem No. 2

พบปัญหาในการรายงานความก้าวหน้า ครั้งที่ 3

สถานะ ☐ กำลังดำเนินการ ☒ แก้ไขสำเร็จ

รายละเอียดปัญหา

เนื่องจากตามแบบแผนเดิมใน Phase ถัดไปคือ ทางทีมจะทำการออกแบบเพิ่มเติมจาก wire frame และเริ่มพัฒนาซอฟต์แวร์ ตามฟังก์ชันต่างๆ ดังนี้ Web App Navigating Page, Financial Calculating และ Business Financial Analytics และทำรูปเล่มโครงงานบทที่ 4 อิงตามรูป Gantt Chart เดิม

Project Planner					
Select a period to highlight at right. A legend describing the charting follows.					
ACTIVITY	PLAN START	(days) PLAN DURATION	ACTUAL START	(days) ACTUAL DURATION	PERCENT COMPLETE
4 Project Document Unit 3	97	4	0	0	0%
Development					
1 Web Application	97	13	0	0	0%
1.1 Design UX/UI	97	4	0	0	0%
1.2 Main Page	101	1	0	0	0%
1.3 Login and Register Page	102	1	0	0	0%
1.4 Edit Profile Page	103	1	0	0	0%
1.5 Biz Tool Page	104	2	0	0	0%
1.6 Check Biz Page	106	2	0	0	0%
1.7 Admin Page	108	1	0	0	0%
1.8 Software Testing	109	1	0	0	0%
2 User	110	4	0	0	0%
2.1 Create User and Authentication	110	1	0	0	0%
2.2 Edit User Data	111	1	0	0	0%
2.3 Recovery Account Handle	112	1	0	0	0%
2.4 Test Software	113	1	0	0	0%
3 Biz Tools - Financial Calculating Tools	114	24	0	0	0%
3.1 Design 'Financial Calculating Tools'	114	6	0	0	0%
3.2 Code 'Financial Calculating Tools'	120	14	0	0	0%
3.3 Code 'History'	134	2	0	0	0%
3.4 Test Software	136	2	0	0	0%
4 Check Biz	138	26	0	0	0%
4.1 Design 'Check Biz'	138	2	0	0	0%
4.2 Code Financial Feasibility Canvas	140	20	0	0	0%
4.3 Test Software	160	4	0	0	0%
5 Admin	164	3	0	0	0%
5.1 Coding 'System management'	164	2	0	0	0%
5.2 Test Software	166	1	0	0	0%
4 Project Document Unit 4	167	4	0	0	0%

รูปที่ 25 แผนการนำเงินงาน Gantt chart รูปแบบเดิม

ซึ่งจากการที่ระบบนี้ต้องการความรู้และความเข้าใจในการคำนวณและผลลัพธ์การประเมินที่แม่นยำสูง จึงจำเป็นต้องใช้เวลาในการออกแบบและ validate กับ stakeholders ให้ถูกต้องมากและมีประสิทธิภาพมากที่สุด และระยะเวลาในการออกแบบและทดสอบฟังก์ชันของระบบนั้นไม่เพียงพอ

แนวทางแก้ไข/การแก้ไข

เนื่องจากการเปลี่ยนแปลงของแผนการดำเนินงานที่มีความละเอียดและรอบคอบมากขึ้น จึงได้มีการปรึกษอาจารย์ที่ปรึกษาและวางแผน Gantt chart ใหม่ โดยใน Phase ถัดไป คือ ตัดส่วนของการพัฒนาซอฟต์แวร์ ออก ทางทีมพัฒนาจะให้ความสำคัญกับการออกแบบและที่มาของปัญหาแทน

6. สิ่งที่จะดำเนินการต่อไป

แผนการพัฒนาซอฟต์แวร์ถัดไปใน Phase 4 ที่มจะดำเนินการตามกำหนดการ Gantt Chart ดังนี้

Project Planner					
<i>Select a period to highlight at night. A legend describing the charting follows.</i>					
ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
2.1 UX/UI Theme	37.5	0.5	41	0.5	100%
2.2 Wireframe	38	2	43.5	2	100%
3 Database	40	1	45.5	1	100%
3.1 Schema Database	40	1	45.5	1	100%
4 UX/UI Design	41	6.5	46.5	0	0%
4.1 Main Page	41	1	46.5	0	0%
4.2 Login and Register Page	42	0.5	46.5	0	0%
4.3 Edit Profile Page	42.5	0.5	46.5	0	0%
4.4 Biz Tool Page	43	2	46.5	0	0%
4.5 Check Biz Page	45	2	46.5	0	0%
4.6 Admin Page	47	0.5	46.5	0	0%
Research and Validation					
1 Research & Market segmentation	47.5	6	0	0	0%
1.1 Research problems	47.5	2	0	0	0%
1.2 Survey for market testing	49.5	2	0	0	0%
1.3 Interview target audience	51.5	2	0	0	0%
2 Solution Validation	53.5	3.5	0	0	0%
2.1 Create prototype	53.5	0.5	0	0	0%
2.2 Create Hypothesis assumption	54	1	0	0	0%
2.3 Interview target audience	55	2	0	0	0%
3 Improve from feedback	57	2	0	0	0%
3.1 Improve prototype and UX/UI	57	1	0	0	0%
3.2 Improve software design	58	1	0	0	0%
4 API Design	59	2	0	0	0%
4.1 Design API	59	2	0	0	0%
5 Project Document Unit 3	61	2	0	0	0%

รูปที่ 26 แผนการนำเงินงาน Gantt chart phase 4

ใน Phase 4 ทางทีมจะมีดำเนินงานแบ่งเป็นสามส่วน คือ 1. ทำการออกแบบ UX/UI เพิ่มเติมจาก wire frame และทำ prototype เพื่อนำไป validate กับ stakeholders รวมไปถึงทำการปรับปรุงในส่วน Schema ของระบบหลังบ้าน 2. ทีมจะทำการสำรวจตลาดและทำการค้นคว้าเกี่ยวกับที่มาและความสำคัญของปัญหาเพิ่มเติม และ 3.ทำการออกแบบ API และเลือก dependencies ที่ใช้ในการพัฒนาระบบ

