

ระบบค้นหากองทุนที่เหมาะสมกับผู้ลงทุน

Mutual Fund Searching System

คณะผู้จัดทำ

นาย ฐปนพงศ์ จันทะมะ 600510542

นาย วชิระ นรสิงห์ 600510576

อาจารย์ที่ปรึกษา

ผู้ช่วยศาสตราจารย์ ดร.อารีรัตน์ ตรงรัมย์ทอง

รายงานนี้เป็นส่วนหนึ่งของวิชา การออกแบบและพัฒนาออนไลน์ (204424)

ภาคการศึกษาที่ 2 ปีการศึกษา 2563

## คำนำ

รายงานเล่มนี้จัดทำขึ้นเพื่อเป็นส่วนหนึ่งของวิชา การออกแบบและพัฒนาออนโทโลยี (204424) โดยคณะผู้จัดทำได้สร้างระบบสำหรับค้นหากองทุนให้เหมาะสมกับผู้ลงทุน โดยใช้ฐานข้อมูลในรูปแบบ ออนโทโลยีซึ่งมีความสามารถสำหรับจัดเก็บฐานข้อมูลในรูปแบบฐานข้อมูลแบบความรู้

โดยคณะผู้จัดทำหวังว่าการจัดทำเอกสารฉบับนี้จะช่วยให้ข้อมูลที่เป็นประโยชน์ต่อผู้สนใจในการออกแบบฐานข้อมูลแบบ ออนโทโลยี และ การใช้งานเครื่องมือสำหรับใช้ฐานข้อมูลแบบ ออนโทโลยี

นาย ฐปนพงศ์ จันทะมะ

นาย วชิระ นรสิงห์

## สารบัญ

เรื่อง	หน้า
คำนำ .....	a
สารบัญ .....	b
บทที่ 1 บทนำ.....	1
1.1 แนวคิดหลัก (Concept).....	1
1.2 วัตถุประสงค์.....	1
1.3 ข้อความที่เกี่ยวข้อง .....	1
บทที่ 2 เครื่องมือ และ ทฤษฎีที่เกี่ยวข้อง .....	3
2.1 เครื่องมือที่ใช้ในการพัฒนาระบบ.....	3
2.1.1 Related Technologies.....	3
2.2 ภาษาที่ใช้พัฒนาระบบและภาษาที่ใช้ในการ Query Ontology.....	3
บทที่ 3 การออกแบบ.....	5
3.1 Ontology Schema.....	5
3.2 Class Hierarchy.....	7
3.3 Object Properties.....	11
3.4 Data Properties.....	11
3.5 ตัวอย่างของ Instance.....	12
3.6 Relation between objects.....	14
3.7 ส่วนก่อบริการ (User Interface).....	15
3.8 สถาปัตยกรรมของระบบ (Software Architecture) .....	16

## บทที่ 1

### บทนำ

#### 1.1 แนวคิดหลัก (Concept)

ระบบค้นหากองทุนที่เหมาะสมกับนักลงทุน เนื่องจากการลงทุนเป็นเรื่องที่จำเป็นต้องใช้ความรู้ในการพิจารณา ทำให้นักลงทุนจะต้องศึกษาข้อมูลจำนวนมากก่อนตัดสินใจ แต่ระบบที่จัดทำจะนำ Knowledge Base มาช่วยในการตัดสินใจและเลือกกองทุนที่เหมาะสมกับนักลงทุน และ แสดงข้อมูลการคาดการณ์ ผลตอบแทนที่ได้ผ่านระบบภายนอก โดยการใช้การเขียนโปรแกรมสำหรับแสดงผลคาดการณ์ในอนาคตทั้งนี้ผลคาดการณ์เป็นเพียงการนำข้อมูลจากอดีตมาประมวลผล

#### 1.2 วัตถุประสงค์

เพื่อแสดงกองทุนที่เหมาะสมกับ เป้าหมาย, เงินทุน และ ระยะเวลาการลงทุน ของผู้ลงทุน

#### 1.3 ข้อความที่เกี่ยวข้อง

- บริษัทหลักทรัพย์จัดการกองทุน
- นักลงทุน
- ราคา
  - NAV
  - ราคาซื้อ
  - ราคาขาย
- กองทุนรวม
  - กองทุนเปิด
  - กองทุนปิด
  - กองทุนรวมตลาดเงิน
  - กองทุนรวมตราสารหนี้
  - กองทุนรวมตราสารทุน
  - กองทุนรวมผสม

- กองทุนรวมทรัพย์สินทางเลือก
- กองทุนรวมต่างประเทศ
- ความเสี่ยง
  - เสี่ยงน้อย (1)
  - เสี่ยงมาก (8)
- สินทรัพย์
  - กองทุน
  - หุ้น
  - ตราสารหนี้ภาครัฐ
  - หุ้นกู้
  - ทองคำ
  - น้ำมัน
  - อสังหาริมทรัพย์
- ผลตอบแทน
  - ปันผล
  - สะสมมูลค่า

## บทที่ 2

### เครื่องมือ และ ทฤษฎีที่เกี่ยวข้อง

#### 2.1 เครื่องมือที่ใช้ในการพัฒนาระบบ

##### 2.1.1 Related Technologies

- React Framework สำหรับสร้าง Client-application
- Express.js (a server framework)
- Prisma (an ORM for creating Object-oriented model for Relational Database)
- Fast API (a server framework for python)
- Axios (Library for creating Http Request to RESTful API)
- PostgreSQL สำหรับเก็บข้อมูลที่จำเป็นต้อง Update อยู่ตลอด

ยกตัวอย่างเช่น NAV, ราคาซื้อ, ราคาขาย, รายละเอียดกองทุน

- Protégé (a knowledge base administration tool) สำหรับสร้าง Ontology Schema
- Apache Jena Fuseki สำหรับ Ontology Server เพื่อรับ Query และ Update Command
- fbProphet สำหรับทำนายอัตราการเติบโตของกองทุน

##### 2.1.2 Outer API

- SEC API (กลต.) สำหรับเรียก Fact Sheet และ NAV ของกองทุนรวมทั้งหมด

#### 2.2 ภาษาที่ใช้พัฒนาระบบและภาษาที่ใช้ในการ Query Ontology

##### 2.2.1 Client App และ Server

- TypeScript
- Python
- SQL Language

### 2.2.2 Ontology Server (Jena Fuseki)

- SPARQL Command
- Turtle Type extension

## บทที่ 3

### การออกแบบ

#### 3.1 Ontology Schema

การออกแบบ Ontology Schema ในระบบค้นหากองทุนนั้นเริ่มจากการออกแบบที่ตัวกองทุนซึ่งสำคัญที่สุด โดยกองทุนมี Sub-Class ที่ได้มาจากการจำแนกของ กสท. จากเอกสาร นิยามประกาศ สน.87/2558 ภาคผนวก 2 โดยออกแบบไว้ดังนี้

- กองทุน
  - กองทุนที่จ่ายปันผล
  - กองทุนตราสารหนี้
  - กองทุนตราสารทุน
  - กองทุนผสม
  - กองทุนทางเลือก
  - อื่น

นอกจากนี้กองทุนยังมี Data-Properties หรือ ข้อมูลภายใน Object โดยการเลือก Data-Properties ทำได้จากการวิเคราะห์การเลือกกองทุนเบื้องต้นซึ่งประกอบไปด้วย

- ชื่อรหัสกองทุน
- ผลตอบแทนที่ทำได้ ตั้งแต่เริ่มจัดตั้ง
- ระดับความเสี่ยง
- การขาดทุนมากที่สุด ตั้งแต่จัดตั้ง
- ค่าความคลาดเคลื่อน

เนื่องจากข้อมูลเหล่านี้มีการเปลี่ยนแปลงเพียงปีละครั้งจึงเหมาะสมที่จะนำเข้าไปไว้ใน Knowledge Base เพื่อทำการวิเคราะห์หากองทุนที่เหมาะสม



## ตัวอย่าง Individual ของ กองทุนรวม

Description: BCHINEEQ		Property assertions: BCHINEEQ	
Types +		Object property assertions +	
● asset	? @ x o	■ invest BABAUS	? @ x o
● dividend	? @ x o	■ invest ALCPTUA	? @ x o
● equity_fund	? @ x o	■ invest ALACEPT	? @ x o
● fund	? @ x o	■ invest 700HK	? @ x o
		■ invest 2020HK	? @ x o
Same Individual As +		Data property assertions +	
Different Individuals +		■ asset_symbol "B-CHINE-EQ"	? @ x o
		■ project_sd 21.78	? @ x o
		■ project_id "M0651_2560"	? @ x o
		■ project_name "BCHINEEQ"	? @ x o
		■ risk_rate 6	? @ x o
		■ asset_name "BBLAM"	? @ x o
		■ asset_id "BCHINEEQ"	? @ x o
		■ project_loss -29.68	? @ x o
		Negative object property assertions +	
		Negative data property assertions +	

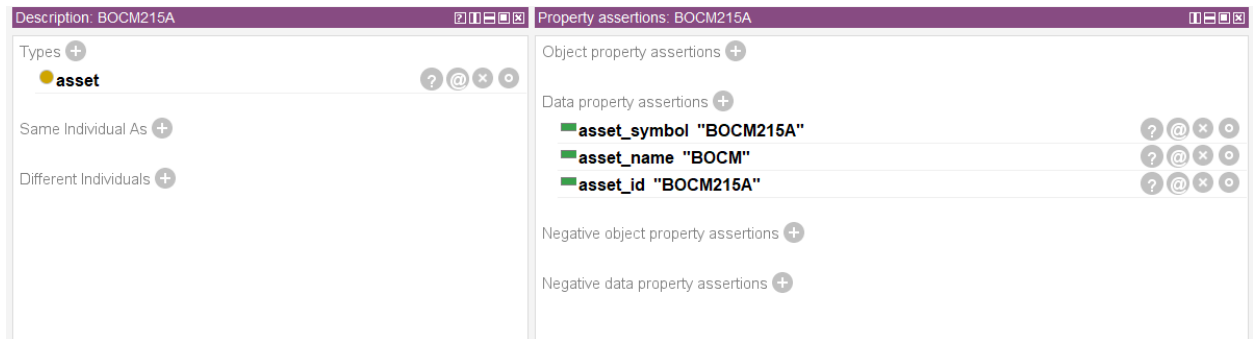
ในส่วนต่อไปคือส่วนของ สินทรัพย์ที่กองทุนลงทุนโดยจะถูกเชื่อมความสัมพันธ์กับกองทุนโดย Object Properties ที่ชื่อว่า Invest โดยลักษณะของ Class สินทรัพย์มีดังนี้

- สินทรัพย์
  - เงินฝาก
  - หน่วยลงทุน
  - หุ้น
  - ตราสารหนี้
  - หุ้นกู้
  - ทองคำ
  - ใบสำแดงสิทธิ

โดย สินทรัพย์จะมี Data Properties เพียง 3 เท่านั้นซึ่งประกอบไปด้วย

- รหัสของสินทรัพย์
- ชื่อของสินทรัพย์
- สัญลักษณ์ของสินทรัพย์

## ตัวอย่าง Individual ของ สินทรัพย์



## ตัวอย่างการเรียก Query Command และ ผลลัพธ์สำหรับเรียกกองทุนใน Ontology

Parameter: dividend: false, loss: 5

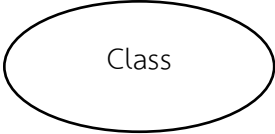
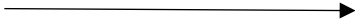

```
const command = `
SELECT ?project_id ?name ?draw_down ?profit ?sd ?risk (?profit+?sd as ?max_profit)
      (?profit-?sd as ?min_profit)
WHERE{
  ?fund mat:project_id ?project_id;
    mat:project_name ?name;
    mat:project_loss ?draw_down;
    mat:project_profit ?profit;
    mat:project_sd ?sd;
    mat:risk_rate ?risk;
    ${dividend ? 'rdf:type mat:dividend;' : ''}
  FILTER ( ?draw_down >= ${loss} && (?sd + ?profit >= ${profit} || ?profit - ?sd >= 0))
  ${risk ? `FILTER (?risk >= ${riskValue[risk].start}
    && ?risk <= ${riskValue[risk].end})` : ''}
}
ORDER BY DESC(?profit+?sd) DESC(?profit-?sd) DESC(?draw_down)
${limit ? `LIMIT ${limit}` : ''}
`
```

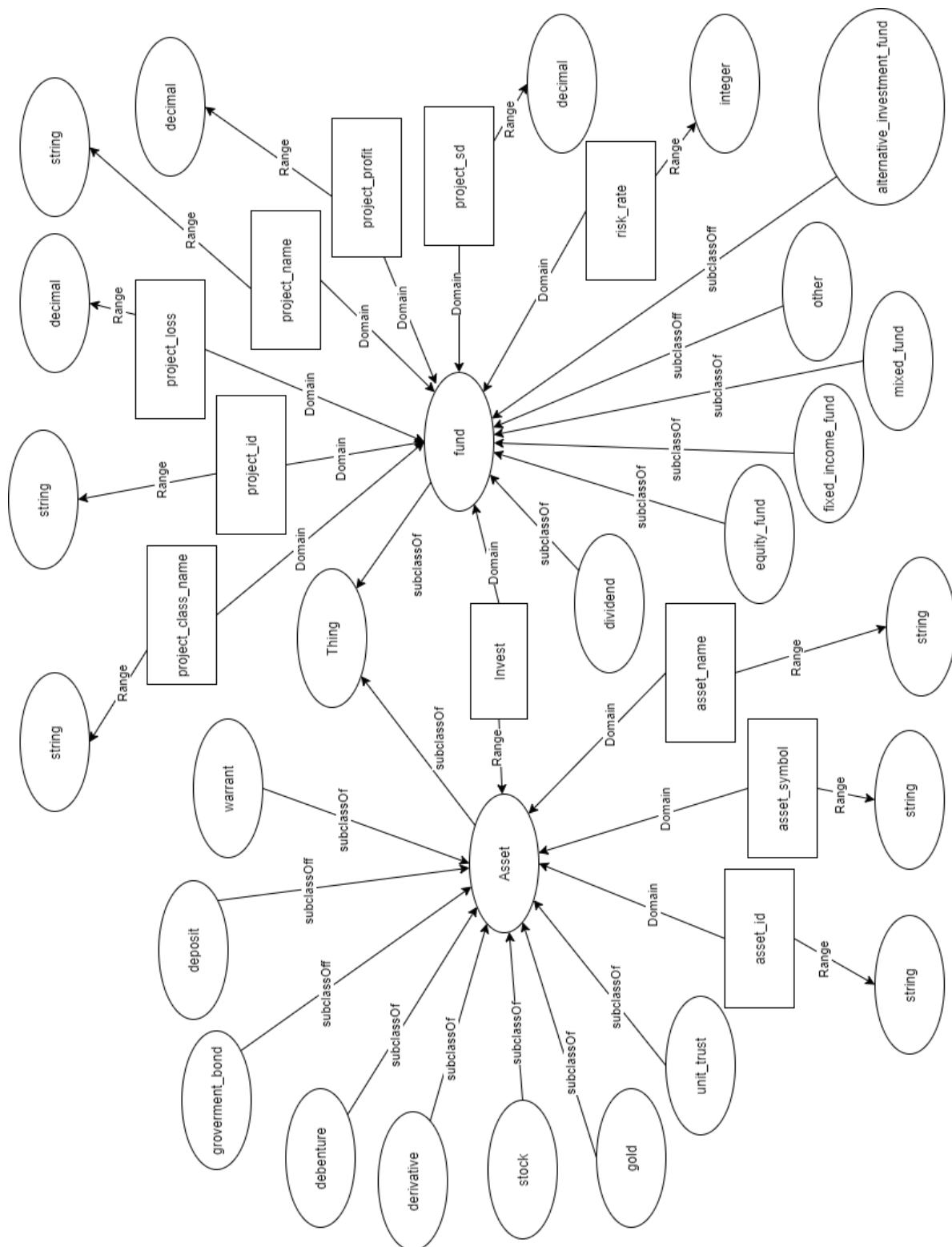
## ผลลัพธ์จาก Query Command

```
[
  {
    "project_id": "M0512_2563",
    "name": "KCHINARMF",
    "draw_down": "-2.16",
    "profit": "8.02",
    "sd": "17.19",
    "risk": "6",
    "max_profit": "25.21",
    "min_profit": "-9.17"
  },
  {
    "project_id": "M0110_2563",
    "name": "BCHINAARMF",
    "draw_down": "-2.48",
    "profit": "8.05",
    "sd": "14.21",
    "risk": "6",
    "max_profit": "22.26",
    "min_profit": "-6.16"
  },
  {
    "project_id": "M0470_2563",
    "name": "KUSARMF",
    "draw_down": "-2.73",
    "profit": "5.24",
    "sd": "16.15",
    "risk": "6",
    "max_profit": "21.39",
    "min_profit": "-10.91"
  },
  {
    "project_id": "M0108_2563",
    "name": "BMAPS100",
    "draw_down": "-1.81",
    "profit": "4.89",
    "sd": "6.4",
    "risk": "6",
    "max_profit": "11.29",
    "min_profit": "-1.51"
  },
  {
    "project_id": "M0534_2563",
    "name": "SCBGEESG",
    "draw_down": "-1.08",
    "profit": "5.37",
    "sd": "2.29",
    "risk": "6",
    "max_profit": "7.66",
    "min_profit": "3.08"
  }
]
```

### 3.2 Class Hierarchy

Class Hierarchy คือ แผนภาพแสดงความสัมพันธ์ระหว่าง Class และ Property ซึ่งมีความสัมพันธ์ในลักษณะ Subject => Predicate => Object โดยใช้สัญลักษณ์ในการแสดงแผนภาพดังนี้

Symbol	Meaning
	แสดงถึง Class ที่เป็น Subject
	แสดงถึง ความสัมพันธ์ระหว่าง Class ถึง Class หรือ Class ถึง Property
	แสดงถึงความหมายของ ความสัมพันธ์ที่เกิดขึ้นระหว่าง Class โดยผ่าน Relation



### 3.3 Object Properties

Name	Domain	Range
invest	Fund	Asset

### 3.4 Data Properties

Name	Domain	Range
asset_id	asset	string
asset_name	asset	string
asset_symbol	asset	string
project_class_name	fund	string
project_id	fund	string
project_loss	fund	decimal
project_name	fund	string
project_profit	fund	decimal
project_sd	fund	decimal
risk_rate	fund	integer

### 3.5 ตัวอย่างของ Instance

#### 1. Class: fund

Sub-Class: equity-fund

Instance Name: SCBSET50

#### Data Property

Subject	Type	Value
project_id	string	M0415_2562
loss	float	36.5%
risk_rate	int	6
sd	float	18.76%
profit	float	5.04%

#### Object Property

Subject	Predicate	Object
SCBSET50	invest	PTT
SCBSET50	invest	AOT
SCBSET50	invest	CPALL
SCBSET50	invest	DELTA
SCBSET50	invest	AIS

2. **Class:** asset

**Sub-class:** stock

**Instance Name:** AOT

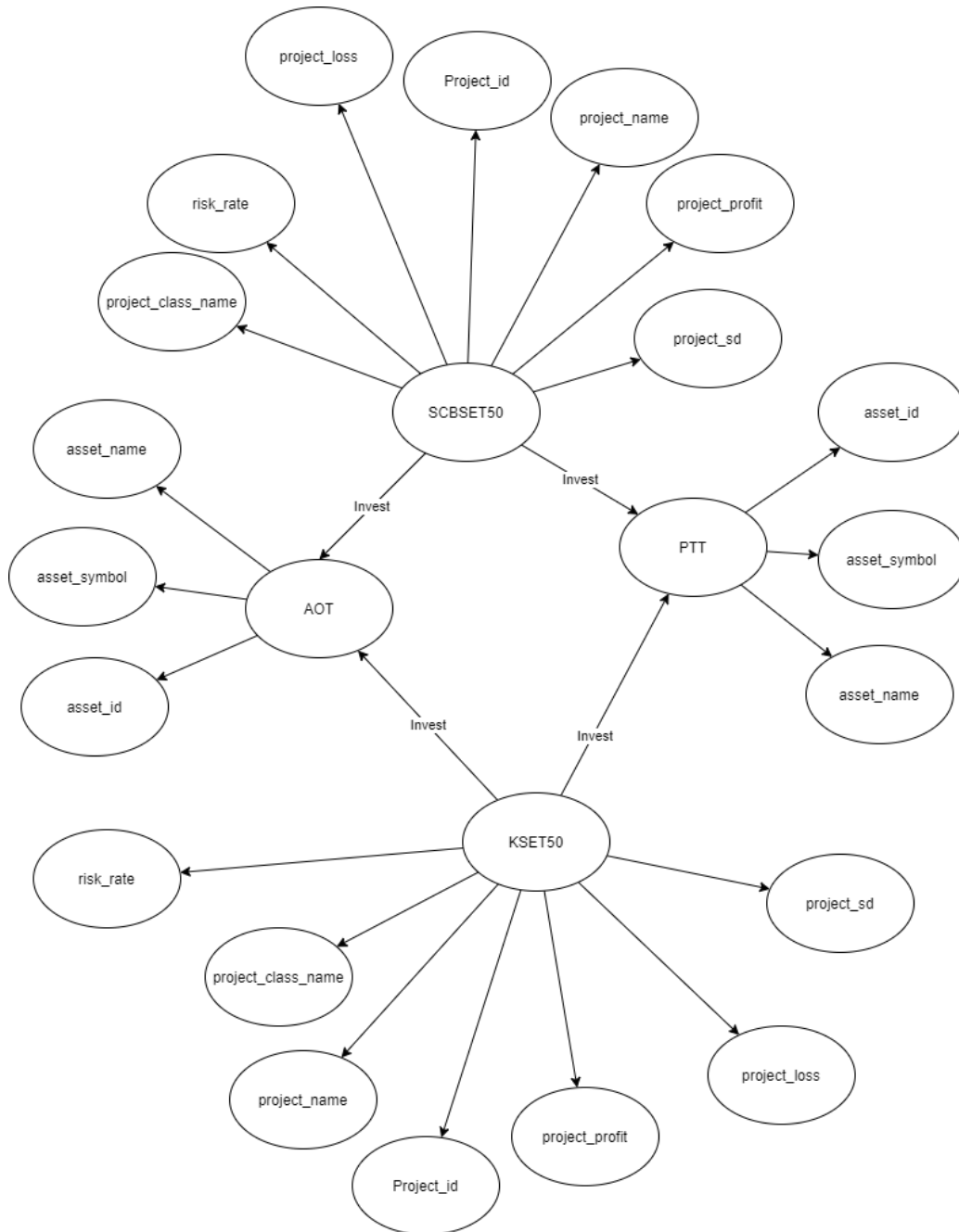
**Data Property**

Property	Type	Value
asset_id	string	AOT
asset_name	string	AIRPORTS OF THAILAND PUBLIC COMPANY LIMITED
Asset_symbol	string	AOT



### 3.6 Relation between objects

ตัวอย่างของ Individual ที่ถูกเชื่อมความสัมพันธ์เข้ากับทั้ง Object Property และ Data Property



### 3.7 ส่วนก่อนประสาน (User Interface)

การออกแบบส่วนก่อนประสานประกอบไปด้วย 5 ส่วนโดยประกอบไปด้วย

#### A. ส่วนรับเข้าข้อมูล

1. เงินทุนเริ่มต้น
2. เงินทุนต่อเดือน
3. ความเสี่ยงที่รับได้
4. ระยะเวลาการลงทุน
5. จำนวนเงินเป้าหมาย
6. การจ่ายปันผล
7. ปุ่มค้นหา

#### B. ส่วนแสดงอัตราส่วนการลงทุน

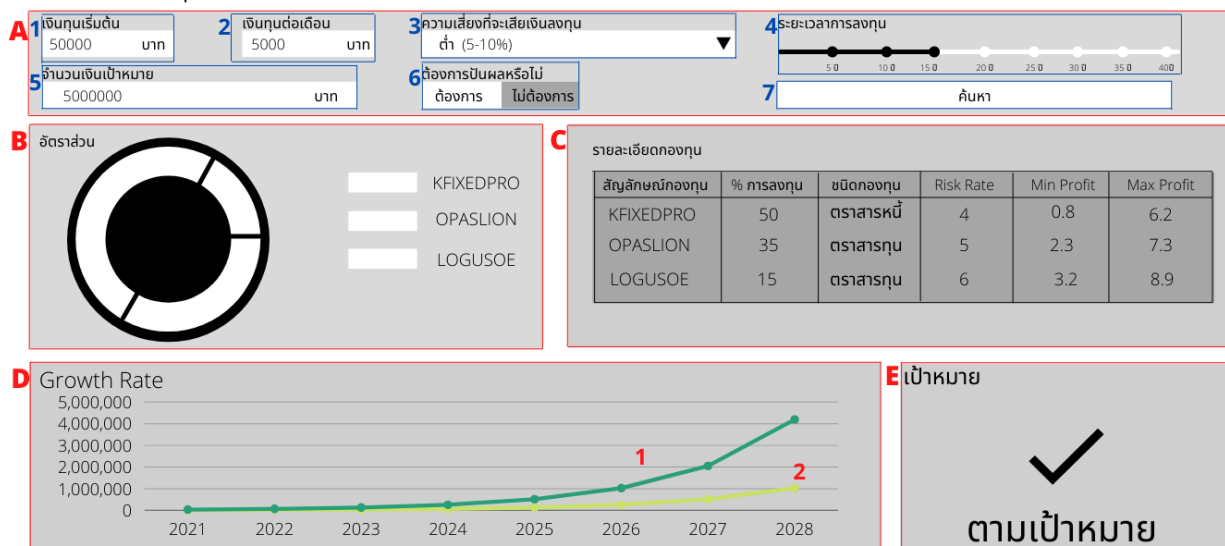
#### C. ส่วนแสดงรายละเอียดของกองทุนที่ถูกเลือก

#### D. ส่วนแสดงการคาดการณ์มูลค่ากองทุนทั้งหมดในอนาคต

1. เส้นที่ 1 แสดงมูลค่ากองทุนทั้งหมด
2. เส้นที่ 2 แสดงจำนวนเงินต้นทั้งหมด

#### E. ส่วนแสดงผลลัพธ์ความสามารถในการทำตามเป้าหมาย

#### ระบบค้นหากองทุน

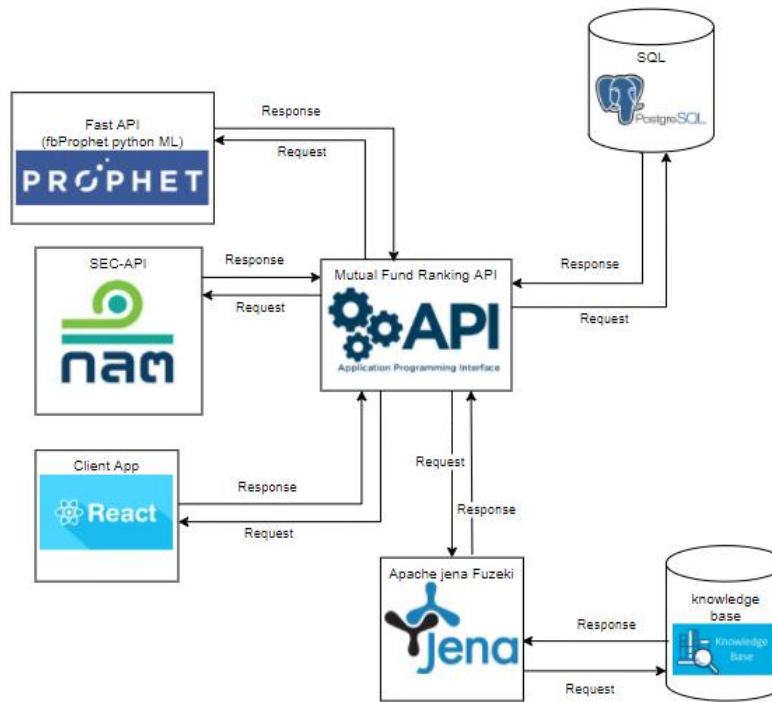


### 3.8 สถาปัตยกรรมของระบบ (Software Architecture)

การออกแบบสถาปัตยกรรมของระบบโดยออกแบบในลักษณะ Three-Tier Architecture โดยมี Application Layer แยกจาก Database Layer และมี Presentation Layer โดยจำแนกได้ดังนี้

1. Presentation Layer
  - a. Client-App
2. Application Layer
  - a. Mutual-Fund-Ranking API: เป็น Entry Point สำหรับ Client-App เพื่อเรียกใช้ API อื่นๆ
  - b. SEC-API: สำหรับ ข้อมูลของกองทุนรวม จาก กลต.
  - c. Predication API: สำหรับ ประมวลผล และ สร้างการคาดการณ์กองทุนผ่าน fbPhophet
  - d. Ontology API: สำหรับ ประมวลและเรียกข้อมูลภายใน Knowledge Base
3. Database Layer
  - a. Turtle Extension: Ontology Schema
  - b. PostgreSQL: Relational Database

โดยภาพรวมของสถาปัตยกรรมทั้งหมดเป็นตาม รูปภาพด้านล่าง



### 3.9 ทฤษฎีที่เกี่ยวข้อง (Related Theory)

ในหัวข้อนี้กล่าวถึงกระบวนการ และ ทฤษฎีทั้งหมดที่ใช้ภายในระบบจัดการกองทุน

#### 1. หลักการคำนวณผลตอบแทนสูงสุดและมากที่สุดของ กองทุนรวม ในระยะเวลา 1 ปี

ในการหาผลตอบแทนสูงสุดของกองทุนรวมนั้นจำเป็นต้องมีข้อมูลประกอบดังนี้

- ผลตอบแทนตั้งแต่จัดตั้งกองทุน (initProfit)
- ค่าความผันผวนตั้งแต่จัดตั้งกองทุน (S.D.)

โดยสามารถคำนวณจากสมการดังนี้

$$\text{estimateMaxProfit} = \text{initProfit} + s.d.$$

$$\text{estimateMinProfit} = \text{initProfit} - s.d.$$

#### 2. ความเสี่ยงและผลตอบแทน (Risk And Reward)

Risk and Reward คือ การนำความเสี่ยงมาเทียบกับผลตอบแทนที่เราจะได้รับนั้นคุ้มค่าหรือไม่ โดยใช้วิธีง่ายๆ โดยนำความเสี่ยงหรือการขาดทุนมานับให้คุ้มค่าต่อการลงทุนยกตัวอย่างเช่นหากขาดทุนที่ 5 % ของเงินทุน ควรที่จะได้ผลตอบแทนมากกว่า 5 % เพื่อที่จะได้รับผลตอบแทนที่คุ้มค่ากับความเสี่ยงที่ให้การลงทุนแต่ละครั้ง โดยใน Project ให้ Risk: Reward อยู่ที่ 1:1.25