

เว็บไซต์ฐานข้อมูล นักศึกษา

Computer Science Research



SE & DB PROJECT

MEMBERS

កម្មករ ពុំពាណិជ្ជ
650510595

Requirement
Manager

កនកលោកមេណី វរុំអ៊ុយ
650510596

Scrum Master

ខ្សែតាំង សរាយុទ្ធនក្រល់
650510605

UX/UI Designer

នុបាល បិនបាន
650510664

Software
Architecture

នីរភ័ព្យ និស្សុមីដូក
650510666

Testing Manager

គ្រារន៍ សែនី
650510681

DevOps

សារិកា ការបិប
650510689

Database Engineer

About the Project

เว็บไซต์ฐานข้อมูลนักศึกษา เป็นเว็บไซต์สำหรับจัดเก็บและสืบค้นผลงานของนักศึกษา เช่น ผลงานค้นคว้าอิสระ, ผลงานวิจัย, โพสต์เรื่องน่าสนใจ ของนักศึกษาระดับ ป.ตรี ให้สามารถเก็บไฟล์ได้หลายรูปแบบ เช่น word, pdf, png หรือ zip

ระบบปัจจุบัน

- ชื่อระบบปัจจุบัน : <https://rsdb.cs.science.cmu.ac.th/>
- ลักษณะของระบบปัจจุบัน : ให้เฉพาะนักศึกษาและบุคลากรภาควิชา
วิทยาการคอมพิวเตอร์มีสิทธิ์เข้าไปสืบค้นหาข้อมูลผลงาน สามารถเปิดอ่าน
ได้อย่างเดียวไม่สามารถดาวน์โหลดไฟล์ โดยต้องลงชื่อเข้าใช้งานด้วย CMU
OAuth รองรับการเปิดจากคอมพิวเตอร์

ปัญหาที่พบในปัจจุบัน

- การอัปโหลดไฟล์จะเป็นการรวบรวมมาให้เจ้าหน้าที่ทำการอัปโหลด โดยอาจจะ 1 ปีครั้งหรือเพอมละครั้ง
- การเลือกหมวดหมู่หรือชนิดของผลงานไม่ตรงตามงานที่นักศึกษาทำ

Business Model

Value Propositions

- ทำให้เว็บไซต์น่าใช้งาน
- สามารถค้นหาผลงานได้ง่าย
- สามารถอัพโหลดผลงานได้สะดวกมากขึ้น

Beneficiaries

- ช่วยประหยัดเวลาในการค้นหาผลงานของนักศึกษาและลดภาระงานของบุคลากรภายในภาควิชา

Functional Requirements

ในฐานะ Admin

- สามารถเข้าใช้ระบบผ่าน CMU OAuth ได้
- สามารถแก้ไข ลบ อัปโหลดผลงานให้นักศึกษาได้
- สามารถเพิ่มบทบาท Admin ให้กับคนอื่นได้
- สามารถคัดกรองนักศึกษาก่อนที่จะอัปโหลดผลงานได้
- สามารถแสดงจำนวนไฟล์ผลงานและจำนวนนักศึกษาได้
- สามารถกำหนดระยะเวลาแก้ไขไฟล์ให้นักศึกษาได้

Functional Requirements

ในฐานะ นักศึกษา

- สามารถเข้าใช้ระบบผ่าน CMU OAuth ได้
- สามารถส่งคำขอเพื่ออัปโหลดไฟล์ผลงานได้
- สามารถอัปโหลดไฟล์ผลงานโดยระบุรายละเอียดต่างๆของผลงานได้
- สามารถค้นหาไฟล์ผลงานผ่าน Keywords ต่างๆได้

Non-Functional Requirements

- **Availability** : สามารถใช้งานระบบได้ 24 ชั่วโมง
- **Scalability** : ระบบรองรับการใช้งานจำนวนมาก เช่น รองรับการใช้งานพร้อมกัน 100 คน
- **Security** : มีความปลอดภัยเนื่องจากการเข้าใช้งานระบบต้อง Sign in ผ่าน CMU OAuth ก่อน
- **Usability** : ระบบสามารถใช้งานได้ง่าย

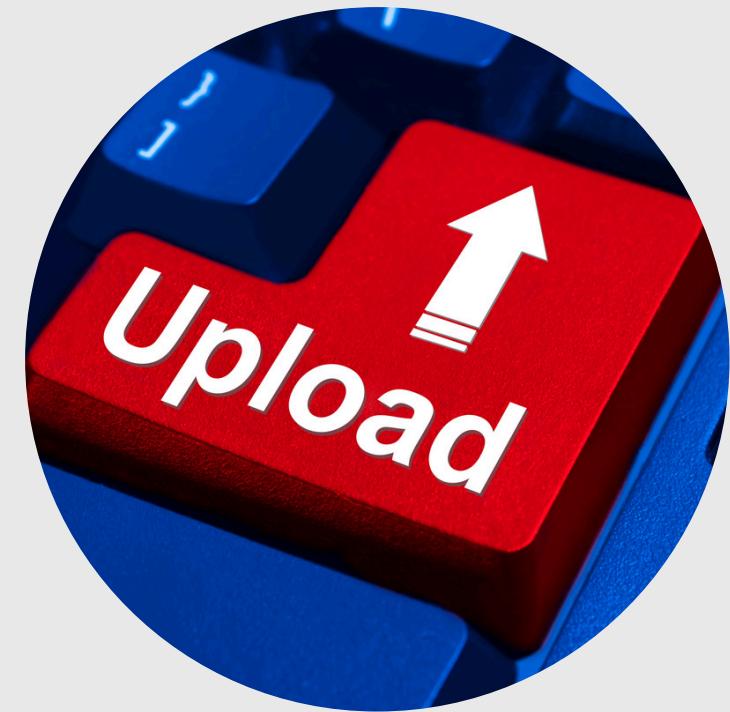
Definition of Done



เข้าใช้งานระบบผ่าน
CMU OAuth ได้



นักศึกษาส่งคำขอเพื่อ
อัปโหลดผลงานได้



นักศึกษาสามารถ
อัปโหลดผลงานได้
ด้วยตัวเอง

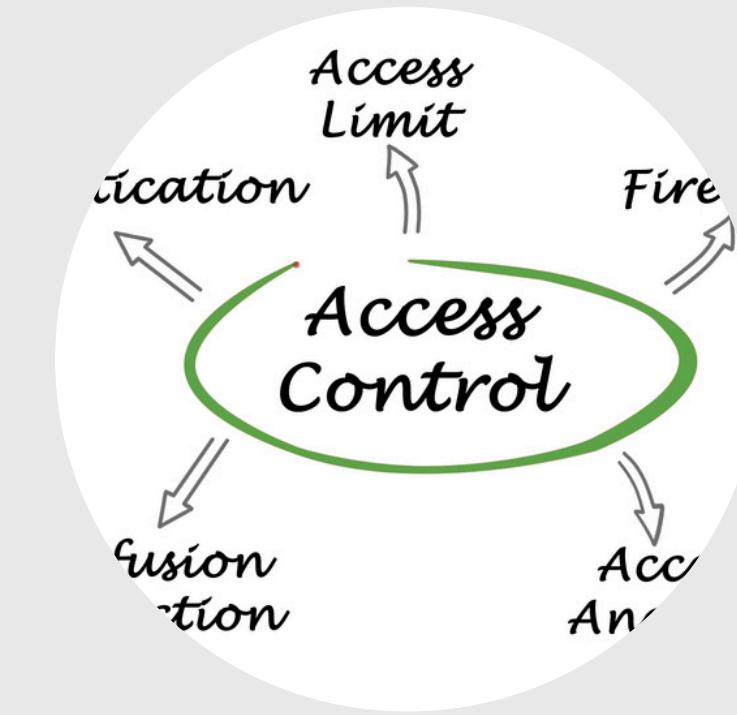
Definition of Done



นักศึกษาสามารถค้นหาผลงานภายในระบบได้



แอดมินสามารถแก้ไข ลบไฟล์ของนักศึกษาได้



แอดมินสามารถให้สิทธิบุคลากรและนักศึกษาได้

Definition of Done



สามารถเพิ่มบทบาท
Admin ให้กับคนอื่นได้



แสดงจำนวนผลงานที่ถูก
อัปโหลดในระบบ

Tools and Technology

- **Flask (Python Web Framework)**

ใช้เป็นเฟรมเวิร์กหลักสำหรับสร้างเว็บแอปพลิเคชันเพื่อจัดการ routing, templating และการเชื่อมต่อกับฐานข้อมูล

- **PostgreSQL (Database)**

ใช้เป็นฐานข้อมูลหลักสำหรับเก็บข้อมูลของโครงการ รวมถึงข้อมูลเกี่ยวกับผู้ใช้

- **SQLAlchemy (ORM)**

ใช้จัดการการเชื่อมต่อและการดำเนินการกับฐานข้อมูล PostgreSQL ผ่าน db

- **Jinja2 (Templating Engine)**

ใช้สำหรับレンเดอร์เทมเพลต HTML ที่มีการเปลี่ยนแปลงข้อมูลแบบไดนามิกจาก Flask

- **Bootstrap 5 (CSS Framework)**

ใช้สำหรับจัดการ responsive และ UI ของเว็บไซต์

Tools and Technology

- **JavaScript (Vanilla JS) และ jQuery**

ใช้สำหรับการจัดการการทำงานที่ฝังผู้ใช้ เช่น การจัดการการส่งฟอร์ม (fetch) และ การแสดงผล UI แบบ interactive

- **Requests (HTTP Library)**

ใช้ในการทำ request ไปยัง API ภายนอก เช่น เพื่อดึงข้อมูลจาก CMU's OAuth Service

- **Functools (wraps)**

ใช้สำหรับสร้าง decorator ที่ใช้ในการตรวจสอบการเข้าถึง เช่น admin_required และ login_required

- **Hashlib (Python Standard Library)**

ใช้ในการแฮชช์ไฟล์เพื่อความปลอดภัย เมื่อมีการอัปโหลดไฟล์

Project Management Tools



Discord

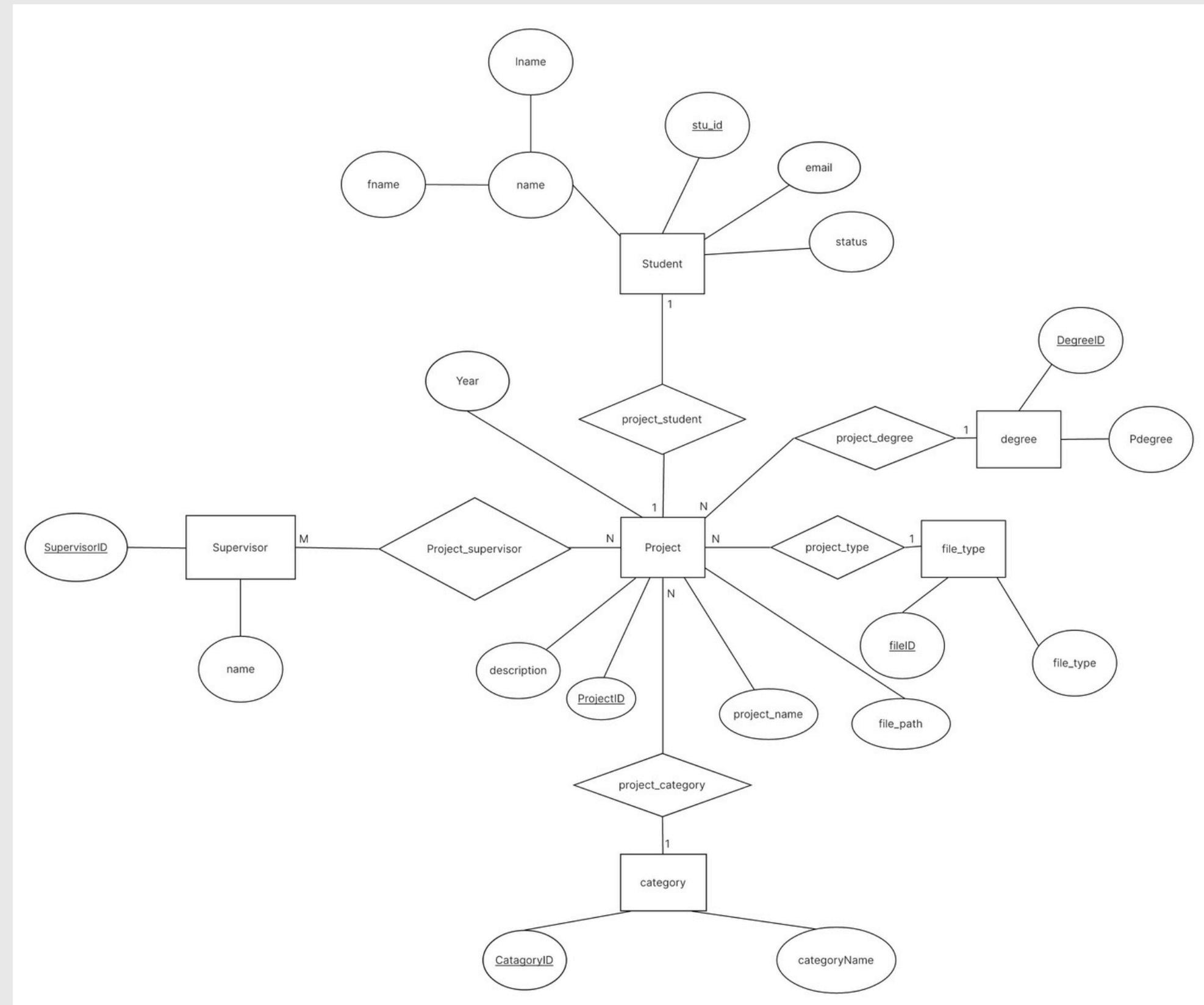


Messenger

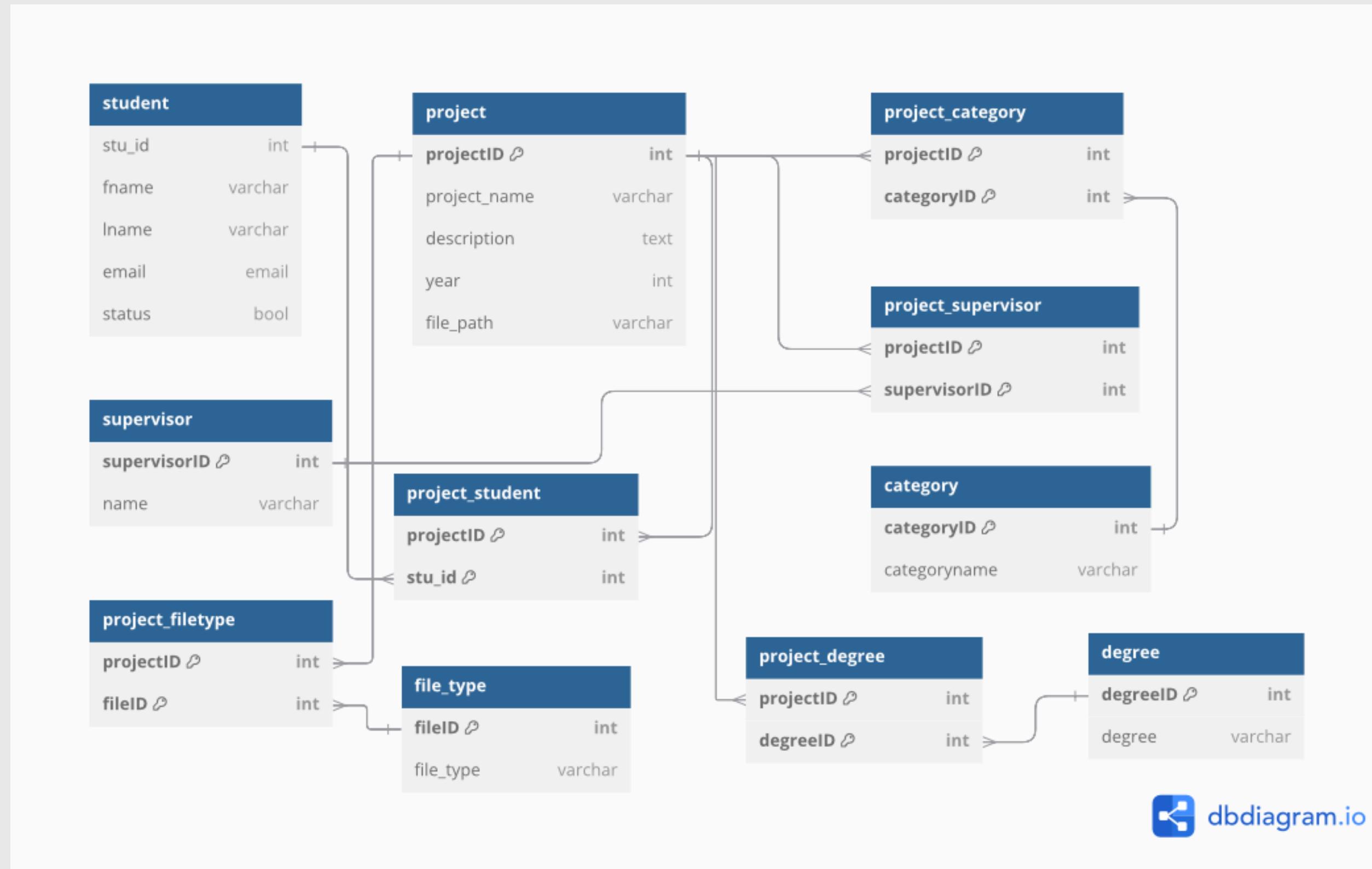


Line

ER Diagram



Database Schema



Step1: Mapping of regular entity types

student

<u>stu_id</u>	fname	lname	email	status

degree

<u>degreeID</u>	degree

file_type

<u>fileID</u>	file_type

Step1: Mapping of regular entity types

category

<u>categoryID</u>	categoryName
-------------------	--------------

supervisor

<u>supervisorID</u>	name
---------------------	------

project

<u>projectID</u>	project_name	description	year	file_path
------------------	--------------	-------------	------	-----------

Step2: Mapping of weak entity types -----> ไม่มี

Step3: Mapping of binary 1:1 relationship types

project_student

<u>projectID</u>	<u>stu_id</u>
------------------	---------------

Step4: Mapping of binary 1:N relationship types

<u>projectID</u>	project_name	year	description	file_path	<u>degreeID</u>	<u>fileID</u>
------------------	--------------	------	-------------	-----------	-----------------	---------------

<u>supervisorID</u>	<u>categoryID</u>
---------------------	-------------------

Step5: Mapping of binary M:N relationship types

project_supervisor

<u>projectID</u>	<u>supervisorID</u>

Step6: Mapping of multivalued attributes -----> ไม่มี

Step7: Mapping of binary N-ary relationship types

<u>projectID</u>	project_name	year	description	file_path	<u>degreeID</u>

<u>supervisorID</u>	<u>categoryID</u>	<u>fileID</u>

1NF ค่าข้อมูลในแต่ละ attribute มีค่าเป็น atomic value แล้ว

2NF จัดการข้อมูลที่เป็น Partial Dependency ซึ่งหมายถึงทุก column ในตารางควรขึ้นกับ primary key ทั้งหมด

<u>projectID</u>	project_name	year	file_path	description
------------------	--------------	------	-----------	-------------

project_degree

<u>projectID</u>	<u>degreeID</u>
------------------	-----------------

project_filetype

<u>projectID</u>	<u>fileID</u>

project_category

<u>projectID</u>	<u>categoryID</u>

3NF ไม่มี transitive dependency เพราะแต่ละตารางอ้างอิงกับ primary key ของตัวเองโดยตรง

PRODUCT SHOWCASE

VERIFICATION

Test Case	Test Input	Expected Output	Status
ตรวจสอบการล็อคอินเข้าระบบของนักศึกษาและ Admin	CMU mail ของนักศึกษา และAdmin	ล็อคอินสำเร็จ และredirectไปถูกหน้า	PASS
ตรวจสอบการอัปโหลดโครงการ	ข้อมูลโครงการที่ต้องการอัปโหลด	โครงการถูกอัปโหลดเข้าสู่ระบบ และแสดงผลบนหน้าเว็บ	PASS
ตรวจสอบการส่งคำขออัปโหลดไฟล์ไปยังAdmin	นักศึกษาส่งคำขออัปโหลดไฟล์	คำขออัปโหลดแสดงผลในหน้าAdmin	PASS
ตรวจสอบการเพิ่มRole Admin	CMU mail ของผู้ที่ต้องการจะเพิ่มเป็นAdmin	CMU mail ถูกเพิ่มลงไว้ใน Role Admin	PASS
ตรวจสอบการค้นหาโครงการ	ค้นหาโครงการตามหมวดหมู่ ประเภทโครงการ ชั้นปี และอาจารย์ที่ปรึกษาโครงการ	ผลการค้นหาอุปกรณ์ตามข้อมูลที่ใส่ไป	PASS

VALIDATION

Requirements	Status
ผู้ใช้และแอดมินสามารถเข้าใช้งานระบบผ่าน CMU OAuth	Pass
แอดมินสามารถแก้ไข ลบ อัปโหลดไฟล์ของนักศึกษาได้	Pass
แสดงจำนวนไฟล์ที่นักศึกษาอัปโหลด	Pass
แอดมินเพิ่มบทบาทแอดมินให้คนอื่นได้	Pass
แอดมินคัดกรองนักศึกษาที่จะอัปโหลดผลงานได้	Pass

VALIDATION

Requirements	Status
ผู้ใช้สามารถส่งคำขอเพื่ออัปโหลดไฟล์ผลงานได้	Pass
ผู้ใช้สามารถค้นหาไฟล์ผลงานผ่าน Keywords ต่างๆได้	Pass
ผู้ใช้สามารถอัปโหลดผลงานได้ด้วยตัวเอง	Pass
แอดมินกำหนดระยะเวลาแก้ไขไฟล์ให้นักศึกษาได้	Fail

Technical Debt

- การสื่อสารไม่ดี : (Poor Communication)
- ทำการทดสอบโค้ดน้อยเกินไป : (Insufficient Code Testing)
- ไม่ได้ Refactory Code : (Lack of Code Refactoring)
- ขาดเอกสารประกอบ : (Lack of Documentation)
- การใช้เทคโนโลยีที่ไม่เหมาะสม : (Using Inappropriate Technology)

Retrospective

ความสำเร็จต่างๆของทีมในทางบุคคล, เทคนิค, หรือ Product ที่จะส่งมอบ

- สมาชิกส่วนใหญ่ทำตามหน้าที่ที่ตนเองรับผิดชอบได้ค่อนข้างดี
- ระบบบอกมาได้ค่อนข้างตรงกับ Requirement ที่ Client ต้องการ

อุปสรรคทางเทคนิคและทางบุคคลที่ทีมต้องเผชิญและวิธีที่ทีม Approach หรือ Solve อุปสรรคเหล่านั้น

- ทางเทคนิค : ไม่ถนัดในการใช้ Tools หรือ Technology บางตัว จนทำให้งานล่าช้า หรือทำให้งานพัง วิธีแก้ไข คือ ใช้บ่อยๆและหัดเรียนรู้จากหลายๆแหล่ง
- ทางบุคคล : เวลาไม่ตรงกัน การสื่อสารที่ไม่ต่อเนื่องหรือขาดการติดต่อในบางครั้ง วิธีแก้ไข คือ พยายามตามและพูดคุยกันให้รู้เรื่องภายในครั้งเดียว

Retrospective

หากมีเวลาเพิ่มอยากรอเพิ่ม Feature หรือปรับปรุงงานให้ดีขึ้น

- ทำให้ไฟล์สามารถดาวน์โหลดได้
- ทำให้ Admin สามารถจัดการเรื่อง Category ได้อย่างชัดเจน
- แสดงยอดการเข้าชมและการดาวน์โหลดไฟล์
- ทำหน้าบ้านให้สวยงามและมีความน่าใช้มากกว่านี้
- ทดสอบระบบทั้งหน้าบ้านและหลังบ้านเพิ่มเติม

คำแนะนำ

เลือกโปรเจคดีๆพยายามเลือกโปรเจคของคนในภาค เพราะจะได้ติดต่อพBg กันได้ง่าย