

Fullstack Development

Self Evaluation

Roadmap

<https://roadmap.sh/full-stack>

Form

<https://forms.office.com/r/ZGzaZM9yxm>

Preflight Project

Objective

Develop "end-user-ready" full-Stack application in 2 weeks.

- Ideas

Why?

- Give students experience of the overview of the full-stack technology.
 - Before going into more details later.
- Want smaller groups ➡ better learning experience.
- Give students more chance to iterate ➡ better coding decision in the project.

Requirement

- Full stack technology
 - Anything that you can run on local development environment.
 - No cloud service (e.g. Firebase)
 - If using JavaScript frameworks, use TypeScript.
- Automated Testing
 - API testing
 - UI testing

Requirement (cont)

- Deployment
 - Deploy on CPS virtual machine (Ubuntu).
 - No Netlify, Vercel, Railway
- Source code repository
 - GitHub
 - GitLab

Due date

| 28 July 2014

Deliverables

- URL to your deployed application
- URL to your source-code repositories
- URL to your VDO
 - File-downloadable (ผมขอเก็บไฟล์)
 - ความยาวไม่เกิน 5 นาที (Please)

VDO Checklists (1)

- Introduction
 - แนะนำสมาชิกในกลุ่ม
 - ทำแอปอะไร สาธิตการทำงาน (Show me your CRUD!)
 - สรุป Technology Stack

VDO Checklists (2)

- Database
 - ใช้ Database อะไร
 - มีโครงสร้างข้อมูลเป็นอย่างไร (e.g. แสดงหน้าใน Dbeaver)
 - ใช้ ORM อะไร
 - มีอะไรที่อยาก Show เทพ (แตกต่างจากที่สอน)

VDO Checklists (3)

- Backend
 - ใช้ Framework อะไร
 - มี Endpoint อะไรบ้าง (สามารถอธิบายจาก Code หรือเขียน Diagram)
 - มีอะไรที่อยาก Show เทพ (แตกต่างจากที่สอน)
- Frontend
 - ใช้ Framework อะไร
 - อธิบาย Structure ของ Code คร่าวๆ
 - มีอะไรที่อยาก Show เทพ (แตกต่างจากที่สอน)

VDO Checklists (4)

- Testing
 - ใช้ Framework อะไร
 - มี การ Test API อะไรบ้าง
 - มี การ Test UI อะไรบ้าง
 - มีอะไรที่อยาก Show เทพ (แตกต่างจากที่สอน)

VDO Checklists (5)

- Deployment
 - ใช้ Technology อะไร (e.g. Docker)
 - อธิบาย Step ของการ Deployment
 - แสดง `Dockerfile` , `docker-compose.yml`
 - อธิบายการสร้าง Image/Container, etc.
 - มีอะไรที่อยาก Show เทพ (แตกต่างจากที่สอน)

Grade

- 10%
- Score distribution (see image)

Quiz - frontend development	2. สามารถพัฒนาระบบแอปพลิเคชันด้วยเฟรมเวิร์กด้านของฟร... x	x	10	
Quiz - RESTful API development	3. สามารถพัฒนาระบบบริการ RESTful API ด้วยเฟรมเวิร์กด้านของฟ... x	x	10	
Database design lab.	4. สามารถออกแบบโครงสร้างฐานข้อมูลได้อย่างเหมาะสม (AbL... x	x	10	
Testing lab.	5. สามารถทดสอบการทำงานของระบบแอปพลิเคชันด้วยเครื่... x	x	10	
การนำเสนอปลายภาค (final presentation)	1. อธิบายหน้าที่ และหลักการทำงานขององค์ประกอบต่าง ๆ ของ... x	x	10	
การสอบวัดความรู้ภาคทฤษฎี	1. อธิบายหน้าที่ และหลักการทำงานขององค์ประกอบต่าง ๆ ของ... x	x	10	
Full stack application project - frontend	2. สามารถพัฒนาระบบแอปพลิเคชันด้วยเฟรมเวิร์กด้านของฟร... x	x	20	
Full stack application project - RESTful API	3. สามารถพัฒนาระบบบริการ RESTful API ด้วยเฟรมเวิร์กด้านของฟ... x	x	10	
Full stack application project - deployment	6. สามารถติดตั้งแอปพลิเคชันที่พัฒนาโดยใช้เทคโนโลยีที่เหมาะสม... x	x	10	

+ Add Assessment Tool

รวมร้อยละการประเมิน (Total Evaluate) 100 %

Extra credit

- Nice-looking UI design
- More complex data relationship (e.g. join tables)
- Includes Authentication / Authorization
- Use different stacks than the class example.

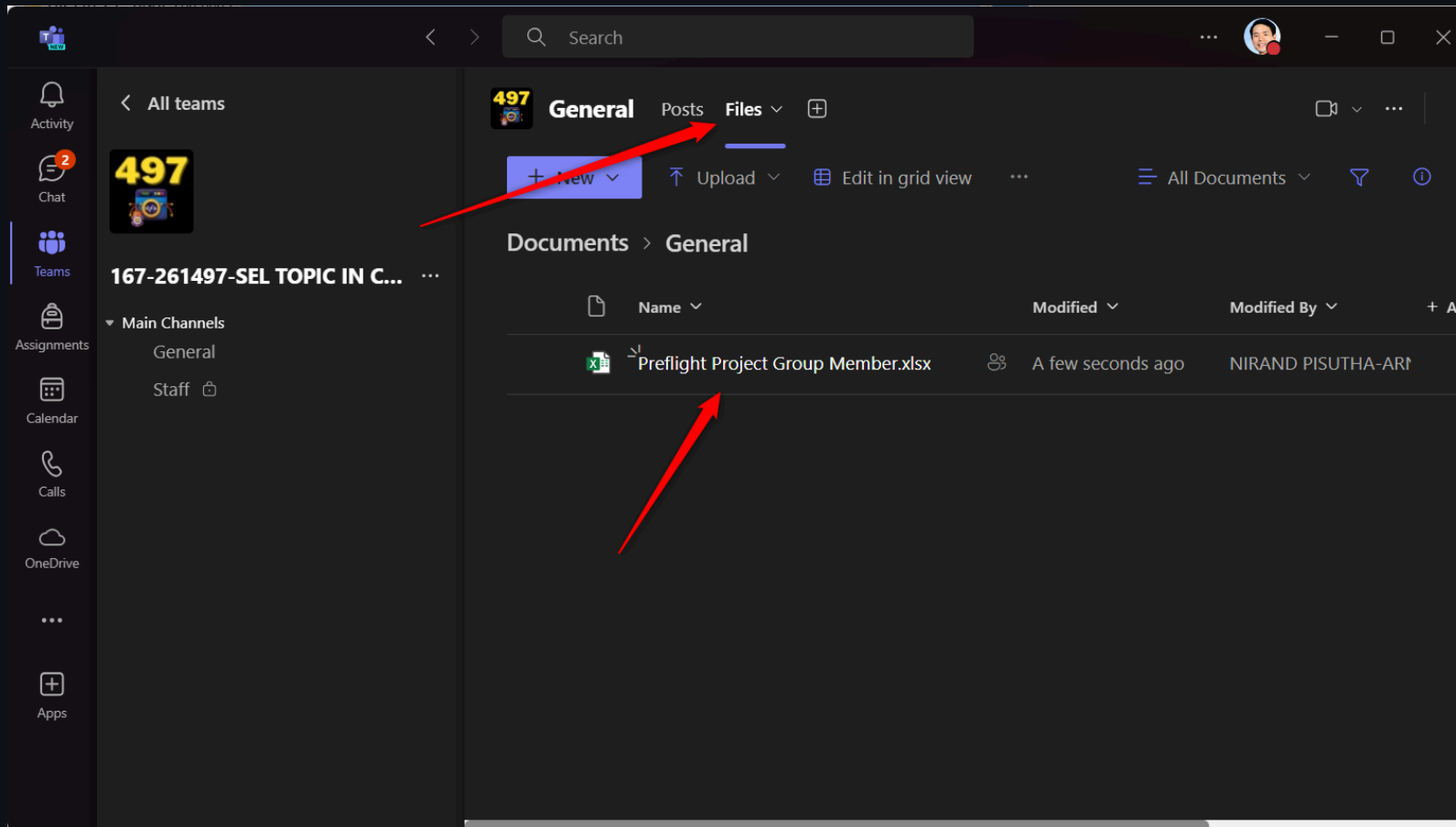
Timeline (1)

Date	Day	Topics
21 Jun	Fri	Modern Javascript / Typescript
25 Jun	Tue	TypeScript
28 Jun	Fri	Database
02 Jul	Tue	Backend
05 Jul	Fri	Frontend

Timeline (2)

Date	Day	Topics
09 Jul	Tue	Testing
12 Jul	Fri	Deployment
16 Jul	Tue	Workshop
28 Jul	Tue	Due date

Group assignment



Tools

- VSCode
 - Prettier
 - ESLint, ES7
 - Quokka.js
- Docker Desktop
- Node.js
- Git

Tools

- Database client
 - Dbeaver
- API Testing tool
 - Insomnia
 - Postman
 - Thunder client (VSCode extension)