



Mahidol University  
Faculty of Information  
and Communication Technology



# GADYPADIESPLIT

ITDS323\_Practical DevOps and Applications

KITSCHYYY

## Core Requirements



### Multi-User Collaboration

- แบ่งค่าใช้จ่ายร่วมกัน: ทริป / กิบข้าว / โครงการกลุ่ม
- รองรับหลายผู้ใช้ (Login หลายคน)
- สร้าง Group (เช่น “Japan Trip”, “Friday Night Gang”) พร้อม history

### Smart Expense Input

- เพิ่มค่าใช้จ่าย: Amount, Payer, Description, Participants
- รองรับวิธีแบ่งเงิน:
  - Equal split
  - Manual (Amount / Percent)
  - Tag-based split (เช่น “Alcohol” คิดเฉพาะคนดื่ม)

### Intelligent Settlement

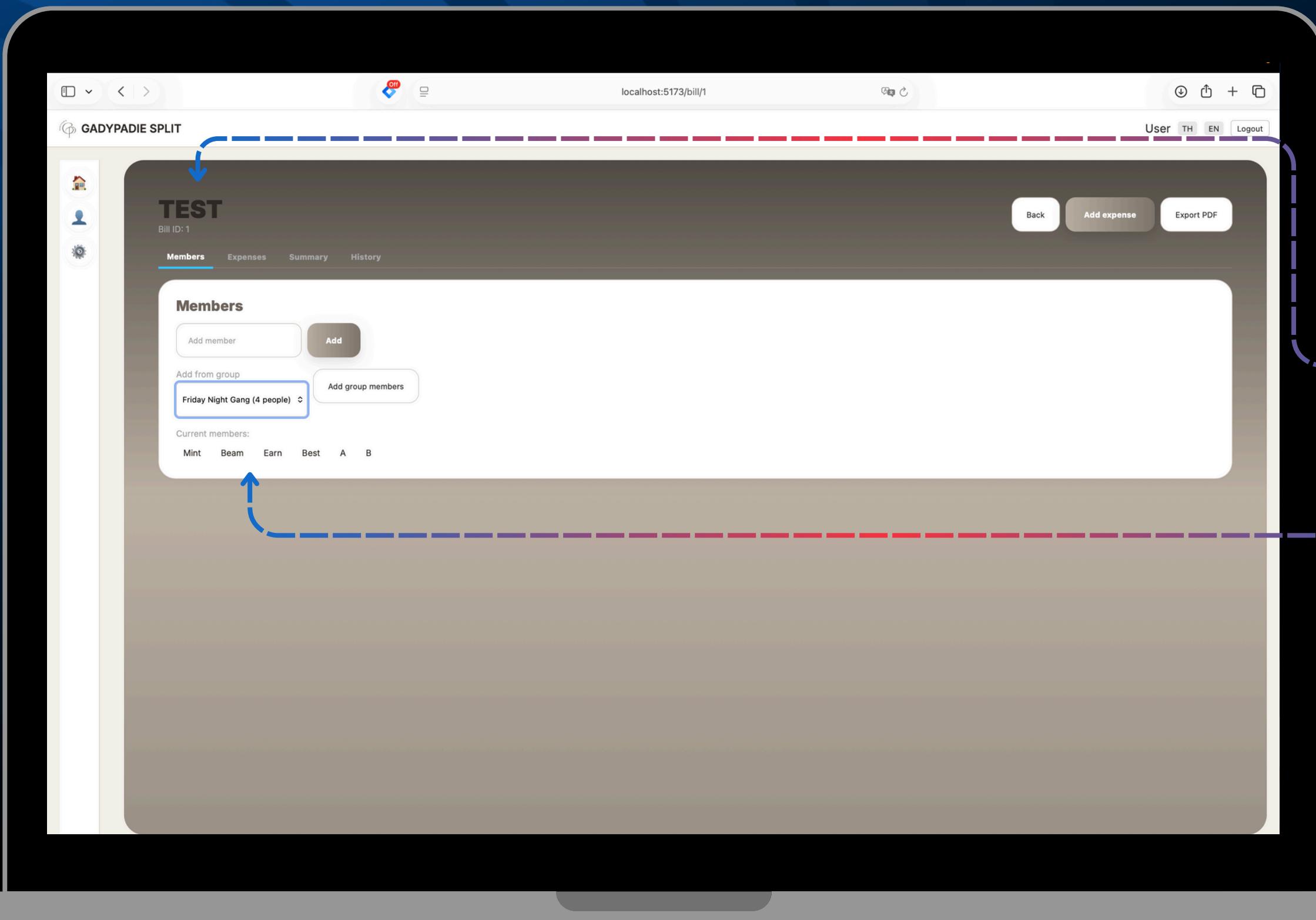
- คำนวณว่า “ใครต้องจ่ายให้ใครเท่าไหร่”
- Mark ว่าชำระแล้ว (settled status)

### Rich Media & Export

- แบบรูปใบเส็จ / voice note
- Export สรุปเป็น PDF

### Currency & Payment Support

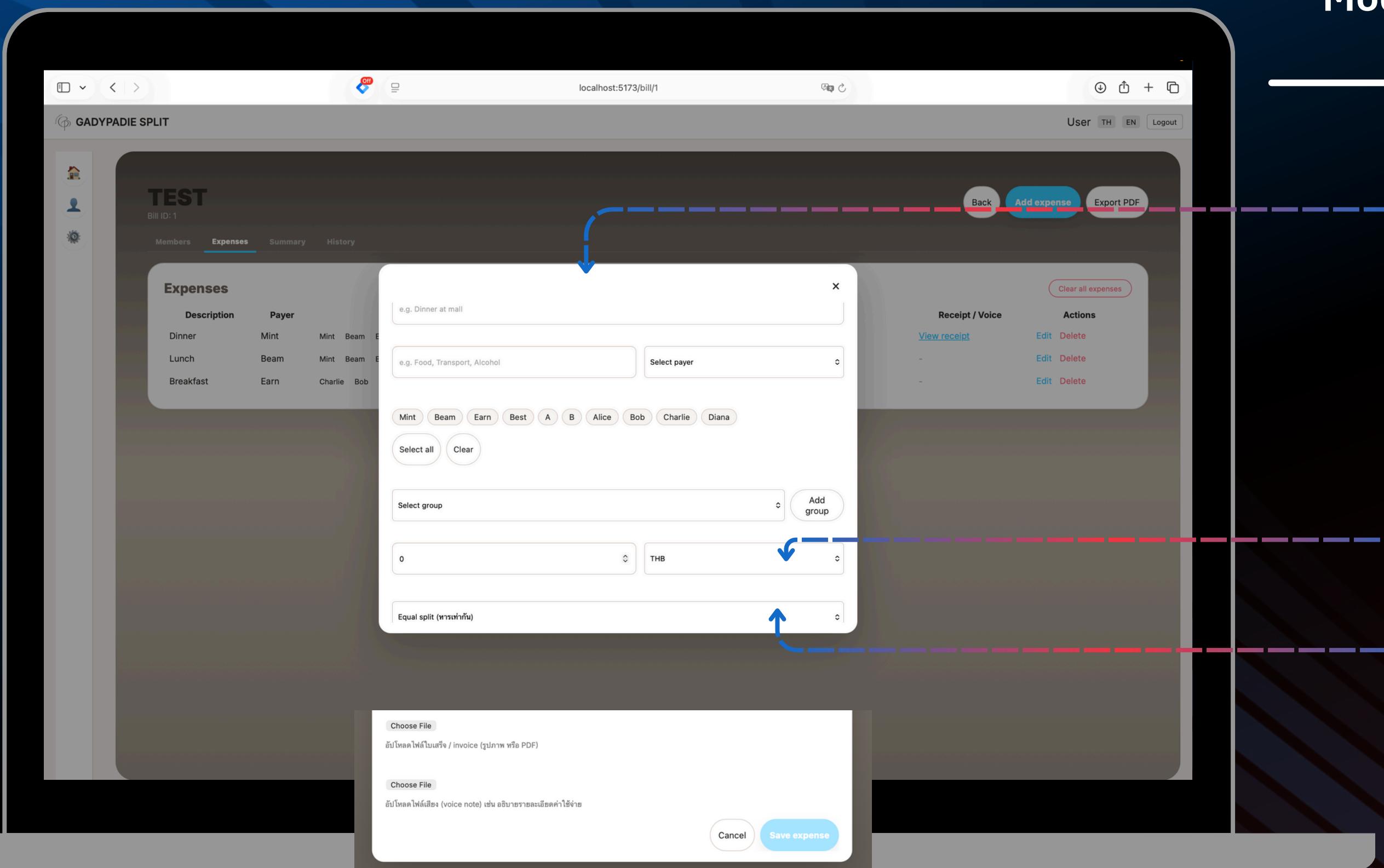
- Multi-currency พร้อมเก็บค่าเดินทาง base
- สร้าง PromptPay QR เพื่อให้โอนเงินง่ายขึ้น



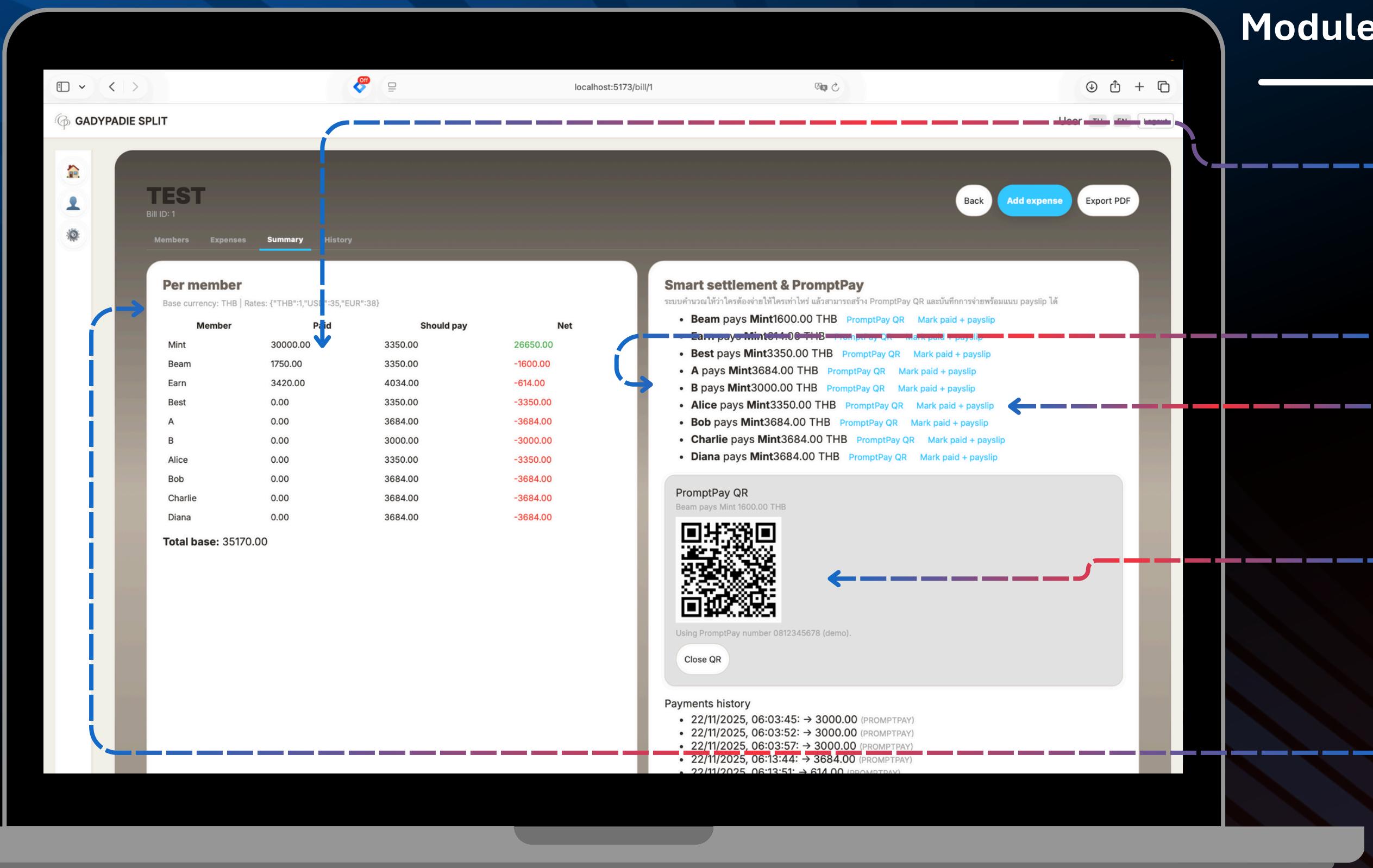
## Module 1: Group & Member Management

- สร้าง Group ใหม่ (Trip, Dinner, Project)
- เพิ่มสมาชิกใน group จาก:
  - รายชื่อเดิมของ user
  - เลือก “Group A” → จะ auto-add คนทั้งกลุ่ม
  - เพิ่ม Individual เพิ่มเติมได้

## Module 2: Expense & Split Engine



- เพิ่ม Expense:
  - Amount, Currency
  - Payer
  - Description + Tag ( เช่น Food, Transport, Alcohol)
  - Participants (เลือกจาก group + individual)
  
- บันทึก original currency + amountBase ( เช่น base เป็น THB )
  
- รองรับ split mode:
  - Equal split → หารเท่ากัน
  - Manual split by amount / percent
  - Tag-based split เช่น tag “Alcohol” คิดเฉพาะคนที่ตีกิว่า drink



**Per member**

Base currency: THB | Rates: {"THB":1,"USD":35,"EUR":38}

Member	Paid	Should pay	Net
Mint	30000.00	3350.00	26650.00
Beam	1750.00	3350.00	-1600.00
Earn	3420.00	4034.00	-614.00
Best	0.00	3350.00	-3350.00
A	0.00	3684.00	-3684.00
B	0.00	3000.00	-3000.00
Alice	0.00	3350.00	-3350.00
Bob	0.00	3684.00	-3684.00
Charlie	0.00	3684.00	-3684.00
Diana	0.00	3684.00	-3684.00

Total base: 35170.00

**Smart settlement & PromptPay**

ระบบคำนวณให้ไว้ต่อจ่ายให้ได้โดยอัตโนมัติ และสามารถสแกน PromptPay QR และมีที่การจ่ายพร้อมแนบ payslip ได้

- Beam pays Mint1600.00 THB [PromptPay QR](#) [Mark paid + payslip](#)
- Beam pays Mint1440.00 THB [PromptPay QR](#) [Mark paid + payslip](#)
- Best pays Mint3350.00 THB [PromptPay QR](#) [Mark paid + payslip](#)
- A pays Mint3684.00 THB [PromptPay QR](#) [Mark paid + payslip](#)
- B pays Mint3000.00 THB [PromptPay QR](#) [Mark paid + payslip](#)
- Alice pays Mint3350.00 THB [PromptPay QR](#) [Mark paid + payslip](#)
- Bob pays Mint3684.00 THB [PromptPay QR](#) [Mark paid + payslip](#)
- Charlie pays Mint3684.00 THB [PromptPay QR](#) [Mark paid + payslip](#)
- Diana pays Mint3684.00 THB [PromptPay QR](#) [Mark paid + payslip](#)

**PromptPay QR**

Beam pays Mint 1600.00 THB



Using PromptPay number 0812345678 (demo).

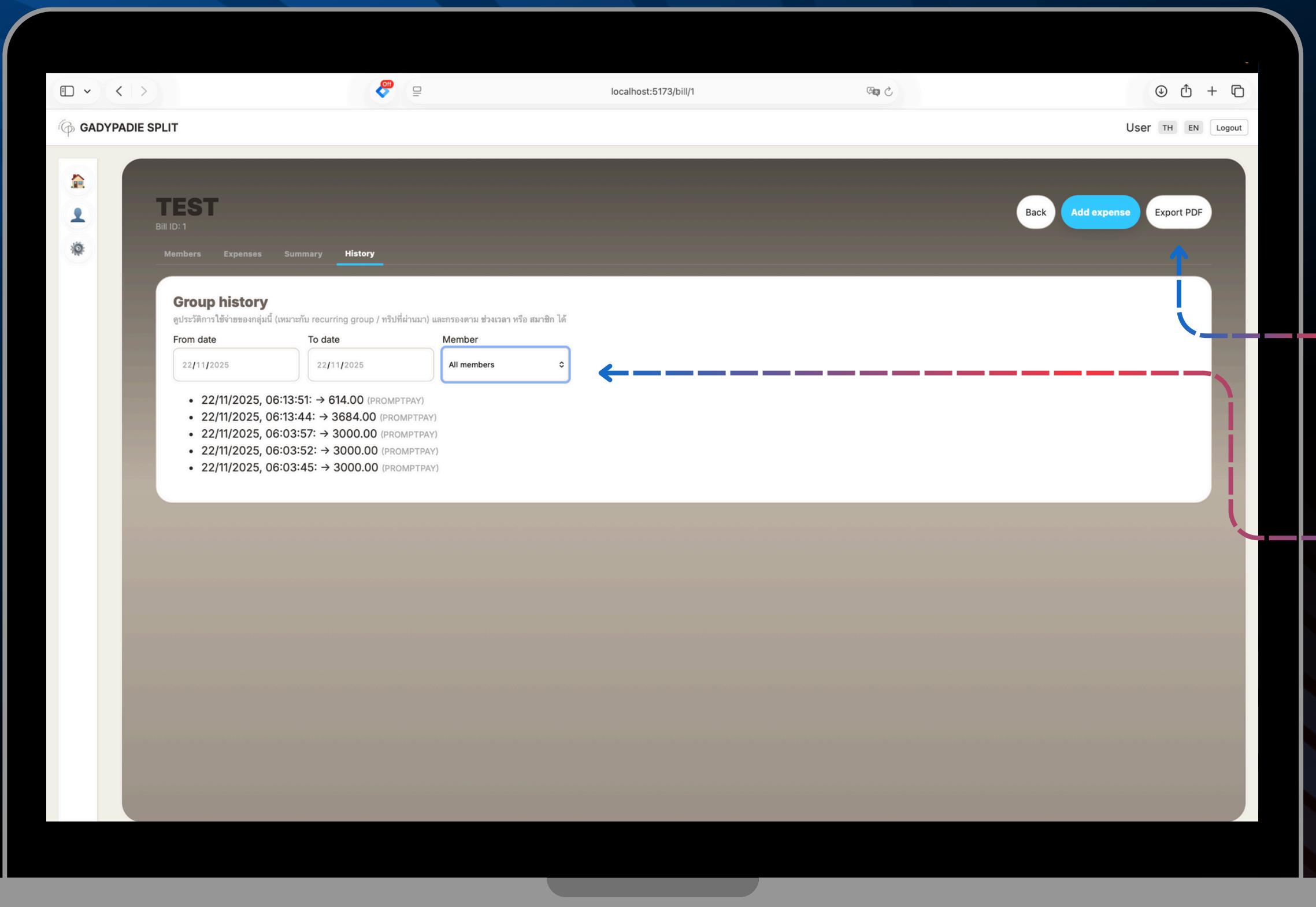
**CLOSE QR**

**Payments history**

- 22/11/2025, 06:03:45: → 3000.00 (PROMPTPAY)
- 22/11/2025, 06:03:52: → 3000.00 (PROMPTPAY)
- 22/11/2025, 06:03:57: → 3000.00 (PROMPTPAY)
- 22/11/2025, 06:13:44: → 3684.00 (PROMPTPAY)
- 22/11/2025, 06:13:51: → 614.00 (PROMPTPAY)

## Module 3: Smart Settlement & PromptPay

- คำนวณว่า แต่ละคนควรจ่าย/รับเท่าไหร่
- Algorithm “Smart Settlement”:
  - ลดจำนวน transaction ให้เหลือน้อยที่สุด (แนวโน้มแบบบุนถอง)
- ให้ผู้ใช้ mark ว่า จ่ายแล้ว พร้อมแนบ payslip
- สร้าง PromptPay QR จากรอดก์ต้องจ่าย:
  - แสดง QR ให้สแกน
  - บันทึก transaction + timestamp เพื่อเป็นประวัติ
- Multi-currency:
  - เก็บ rate ที่ใช้คำนวน ณ ตอนนั้น
  - รองรับการ import rate จากไฟล์ CSV batch



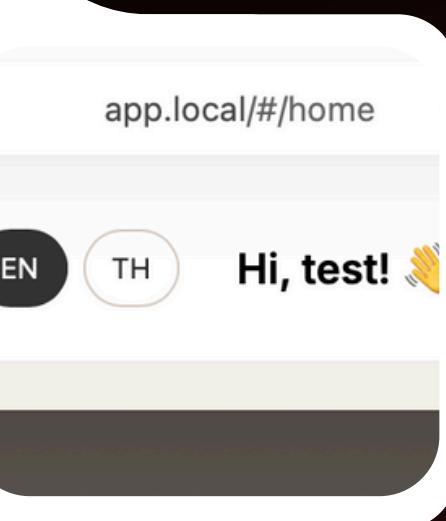
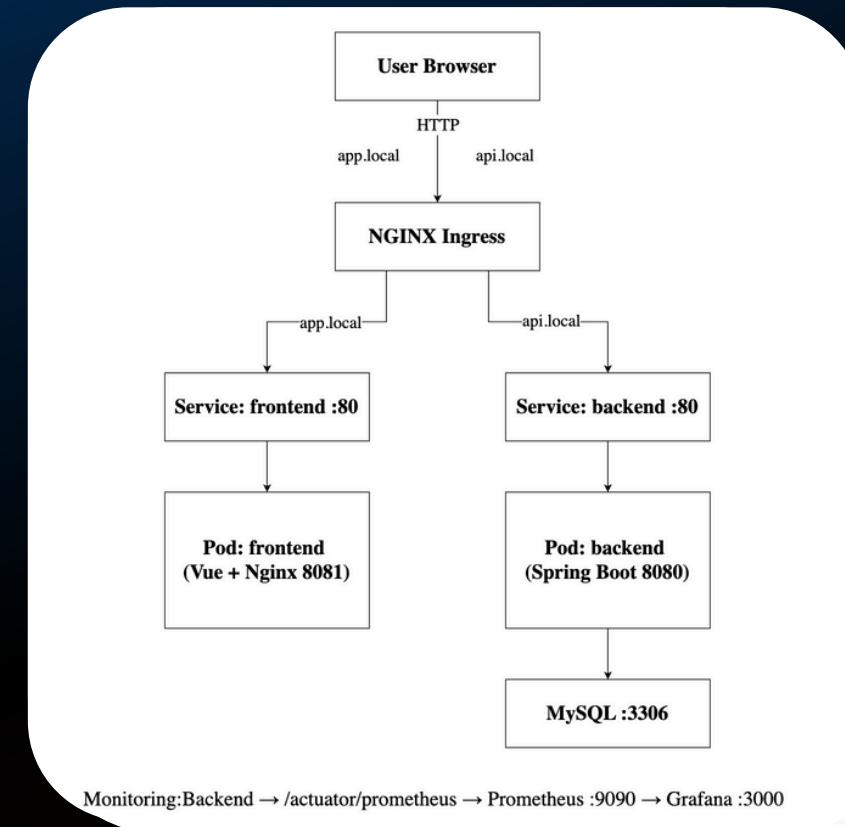
## Module 4: History, Multi-Currency, Export

- Export:
  - สรุประยจ่ายต่อคน
  - สรุป per-expense
  - Export เป็น PDF (ผ่าน jsPDF)
- หน้า Group History:
  - กริปท์ผ่านมา / group ที่ recurring
  - filter ตามเวลา / สมาชิก

## SYSTEM ARCHITECTURE & TECH STACK



- Frontend: Vue + Vite
- Backend: Spring Boot (Java 17)
- Database: MySQL
- API: REST /api/...
- Authentication: Simple token demo + fallback (เพื่อหลีกเลี่ยง external auth)
- Packaging: Docker images (frontend + backend + mysql)
- Orchestration: Kubernetes (Minikube + Ingress)
- Monitoring: Prometheus + Grafana



## Environments & Ports



- Frontend (Vite):  
http://localhost:5173
- Backend:  
http://localhost:8080
- MySQL: localhost:3306

Local Dev

- Frontend:  
http://localhost:8081
- Backend:  
http://localhost:8080
- Grafana:  
http://localhost:3000
- Prometheus:  
http://localhost:9090

Docker Compose

### Kubernetes (Minikube + Ingress)

**http://app.local/ → Frontend**

**http://api.local/ → Backend**

## Result

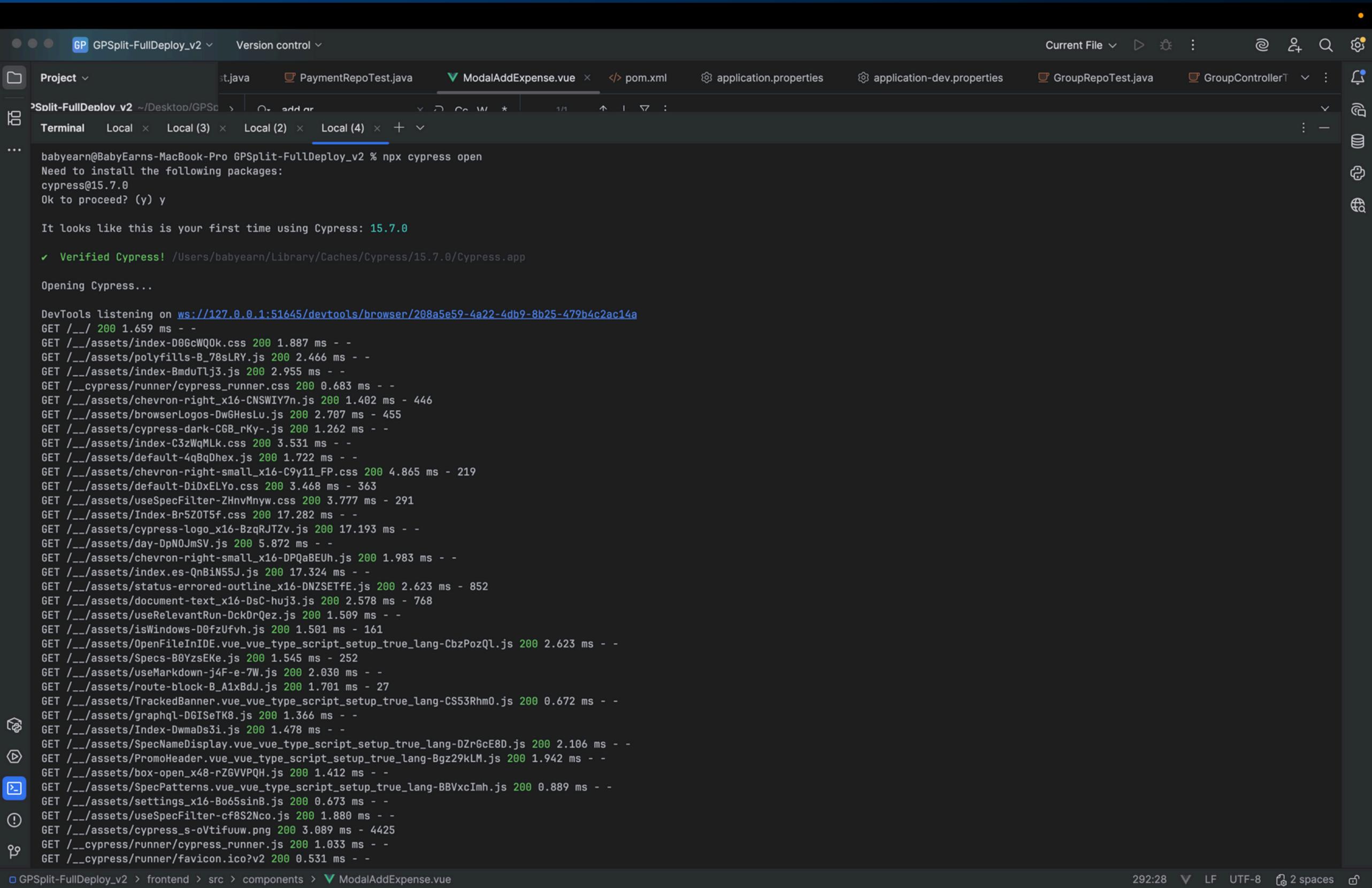


# UNIT TEST

&

# INTEGRATION TEST

## E2E COVERAGE



The screenshot shows a terminal window within a code editor interface. The terminal output is as follows:

```
babyearn@BabyEarns-MacBook-Pro GPSplit-FullDeploy_v2 % npx cypress open
Need to install the following packages:
cypress@15.7.0
Ok to proceed? (y) y

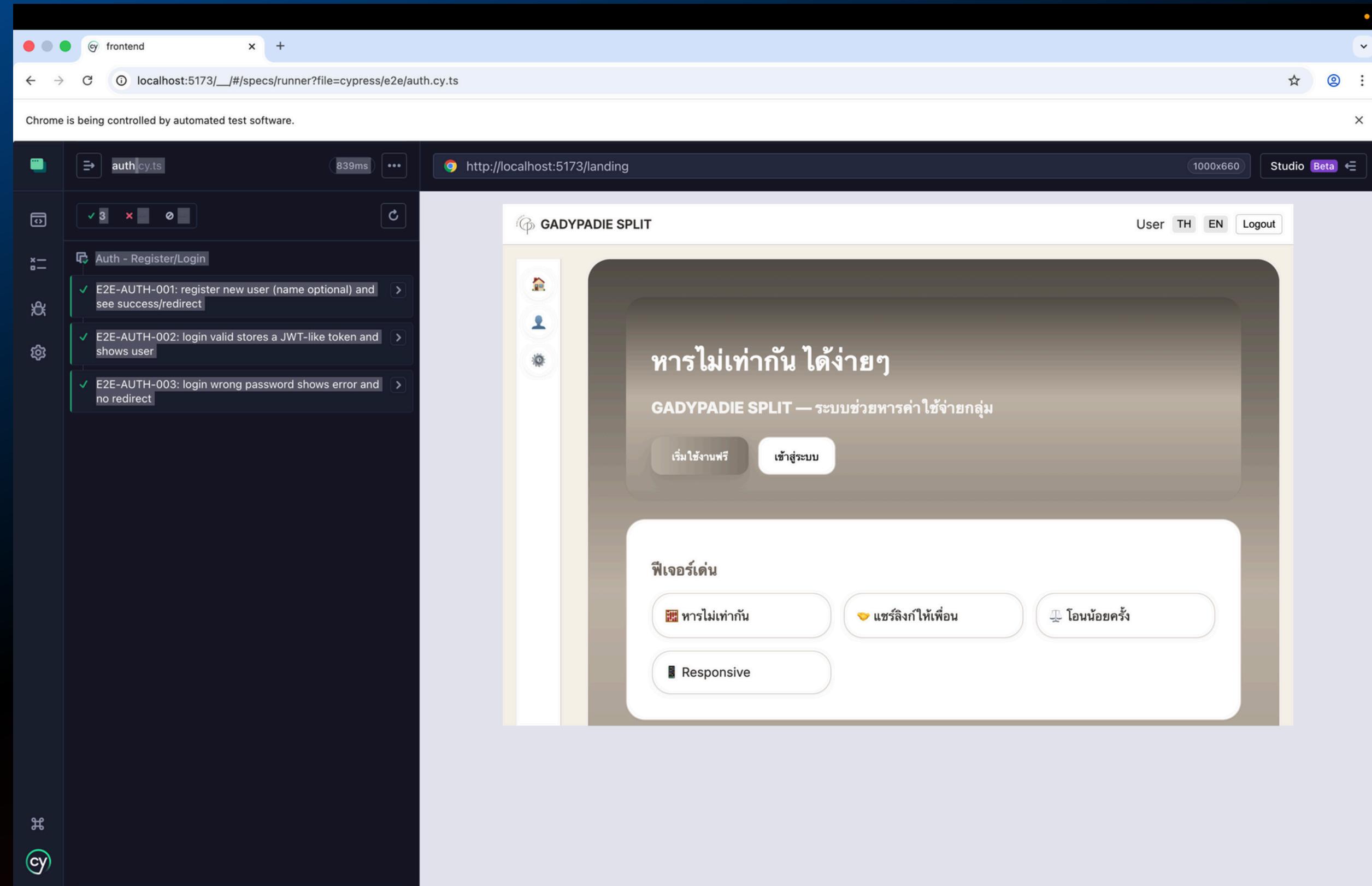
It looks like this is your first time using Cypress: 15.7.0

✓ Verified Cypress! /Users/babyearn/Library/Caches/Cypress/15.7.0/Cypress.app

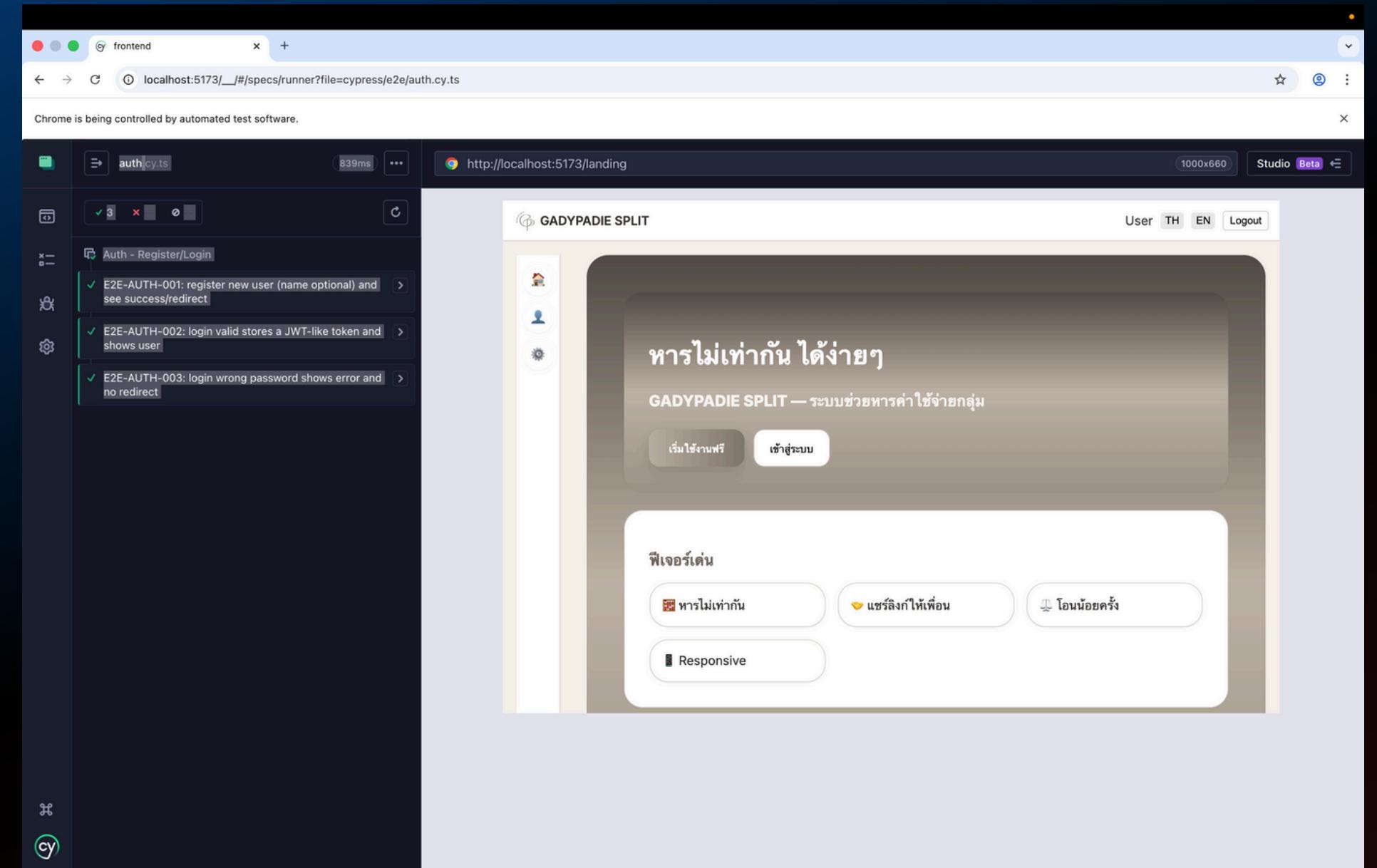
Opening Cypress...

DevTools listening on ws://127.0.0.1:51645/devtools/browser/208a5e59-4a22-4db9-8b25-479b4c2ac14a
GET /__/ 200 1.659 ms - -
GET /__/assets/index-D0GcWQ0k.css 200 1.887 ms - -
GET /__/assets/polyfills-B_78sLRY.js 200 2.466 ms - -
GET /__/assets/index-BmduTlj3.js 200 2.955 ms - -
GET /__cypress/runner/cypress_runner.css 200 0.683 ms - -
GET /__/assets/chevron-right_x16-CNSWIY7n.js 200 1.402 ms - 446
GET /__/assets/browserLogos-DwGHesLu.js 200 2.707 ms - 455
GET /__/assets/cypress-dark-CGB_rKy-.js 200 1.262 ms - -
GET /__/assets/index-C3zWqMLk.css 200 3.531 ms - -
GET /__/assets/default-4qBqDhex.js 200 1.722 ms - -
GET /__/assets/chevron-right-small_x16-C9y11_FP.css 200 4.865 ms - 219
GET /__/assets/default-DiDxELYo.css 200 3.468 ms - 363
GET /__/assets/useSpecFilter-ZHnvMnyw.css 200 3.777 ms - 291
GET /__/assets/Index-Br5ZOT5f.css 200 17.282 ms - -
GET /__/assets/cypress-logo_x16-BzqRJTZv.js 200 17.193 ms - -
GET /__/assets/day-DpNOJmSV.js 200 5.872 ms - -
GET /__/assets/chevron-right-small_x16-DPqaBEUh.js 200 1.983 ms - -
GET /__/assets/index.es-QnBiN55J.js 200 17.324 ms - -
GET /__/assets/status-errorred-outline_x16-DNZSETFE.js 200 2.623 ms - 852
GET /__/assets/document-text_x16-DsC-huj3.js 200 2.578 ms - 768
GET /__/assets/useRelevantRun-DckDrQez.js 200 1.509 ms - -
GET /__/assets/isWindows-D0fzUfvh.js 200 1.501 ms - 161
GET /__/assets/OpenFileInIDE.vue_vue_type_script_setup_true_lang-CbzPozQl.js 200 2.623 ms - -
GET /__/assets/Specs-B0YzsEKe.js 200 1.545 ms - 252
GET /__/assets/useMarkdown-j4F-e-7W.js 200 2.030 ms - -
GET /__/assets/route-block_B_A1xBdJ.js 200 1.701 ms - 27
GET /__/assets/TrackedBanner.vue_vue_type_script_setup_true_lang-CS53Rhm0.js 200 0.672 ms - -
GET /__/assets/graphql-DGISETK8.js 200 1.366 ms - -
GET /__/assets/Index-DwmaDs3i.js 200 1.478 ms - -
GET /__/assets/SpecNameDisplay.vue_vue_type_script_setup_true_lang-DZrGcE8D.js 200 2.106 ms - -
GET /__/assets/PromoHeader.vue_vue_type_script_setup_true_lang-Bgz29kLM.js 200 1.942 ms - -
GET /__/assets/box-open_x48-rZGVVPQH.js 200 1.412 ms - -
GET /__/assets/SpecPatterns.vue_vue_type_script_setup_true_lang-BBVxcImh.js 200 0.889 ms - -
GET /__/assets/settings_x16-Bo6SsinB.js 200 0.673 ms - -
GET /__/assets/useSpecFilter-cf8S2Nco.js 200 1.880 ms - -
GET /__/assets/cypress_s-oVtifuuw.png 200 3.089 ms - 4425
GET /__cypress/runner/cypress_runner.js 200 1.033 ms - -
GET /__cypress/runner/favicon.ico?v2 200 0.531 ms - -
```

## E2E COVERAGE



## E2E COVERAGE



## CI/CD PIPELINE

ระบบ CI/CD ของ GPSplit ใช้ GitHub Runner ในการทำงานอัตโนมัติ โดยเริ่มตั้งแต่การทดสอบ, สร้าง Container Image, จนถึงเตรียม Deploy บน Kubernetes

### Pipeline Automation

- ทำงานอัตโนมัติเมื่อมีการ Push / Pull Request เข้า Branch main
- ดำเนินขั้นตอนต่อเนื่องแบบอัตโนมัติ Test → Build → Push Image → Summary

### Jobs ที่รันอัตโนมัติ

- Backend Build & Unit Test
- E2E Test ด้วย Cypress
- Build & Push Backend Image
- Build & Push Frontend Image (Vue + Vite + Nginx)
- CI/CD Summary

## CI/CD PIPELINE

ระบบ CI/CD ของ GPSplit ใช้ GitHub Runner ในการทำงานอัตโนมัติ โดยเริ่มตั้งแต่การทดสอบ, สร้าง Container Image, จนถึงเตรียม Deploy บน Kubernetes

### Pipeline Automation

- ทำงานอัตโนมัติเมื่อมีการ Push / Pull Request เข้า Branch main
- ดำเนินขั้นตอนต่อเนื่องแบบอัตโนมัติ Test → Build → Push Image → Summary

### Jobs ที่รันอัตโนมัติ

- Backend Build & Unit Test
- E2E Test ด้วย Cypress
- Build & Push Backend Image
- Build & Push Frontend Image (Vue + Vite + Nginx)
- CI/CD Summary

### Backend: Build & Test (Spring Boot)

- ใช้ Gradle + PostgreSQL (service) สำหรับ Unit Test
- ใช้ Gradle Cache ลดเวลา Build มาก
- สร้างรายงานผลการทดสอบแบบ HTML Report
- ตรวจสอบความถูกต้องของ API ก่อนสร้าง Docker Image

### E2E Test (End-to-End)

- ใช้ Docker Compose จำลองทั้งระบบ (Frontend + Backend + Database)
- ใช้ Cypress ทดสอบผ่าน Browser แบบผู้ใช้จริง
- ตรวจสอบการเชื่อมต่อรวม: UI, API และ Database
- เก็บ Screenshot ของผลการทดสอบเป็น Artifacts
- เป้าหมาย: ทดสอบระบบกังระบุเมื่อตอนใช้งานจริง

### Build & Push Frontend (Vue)

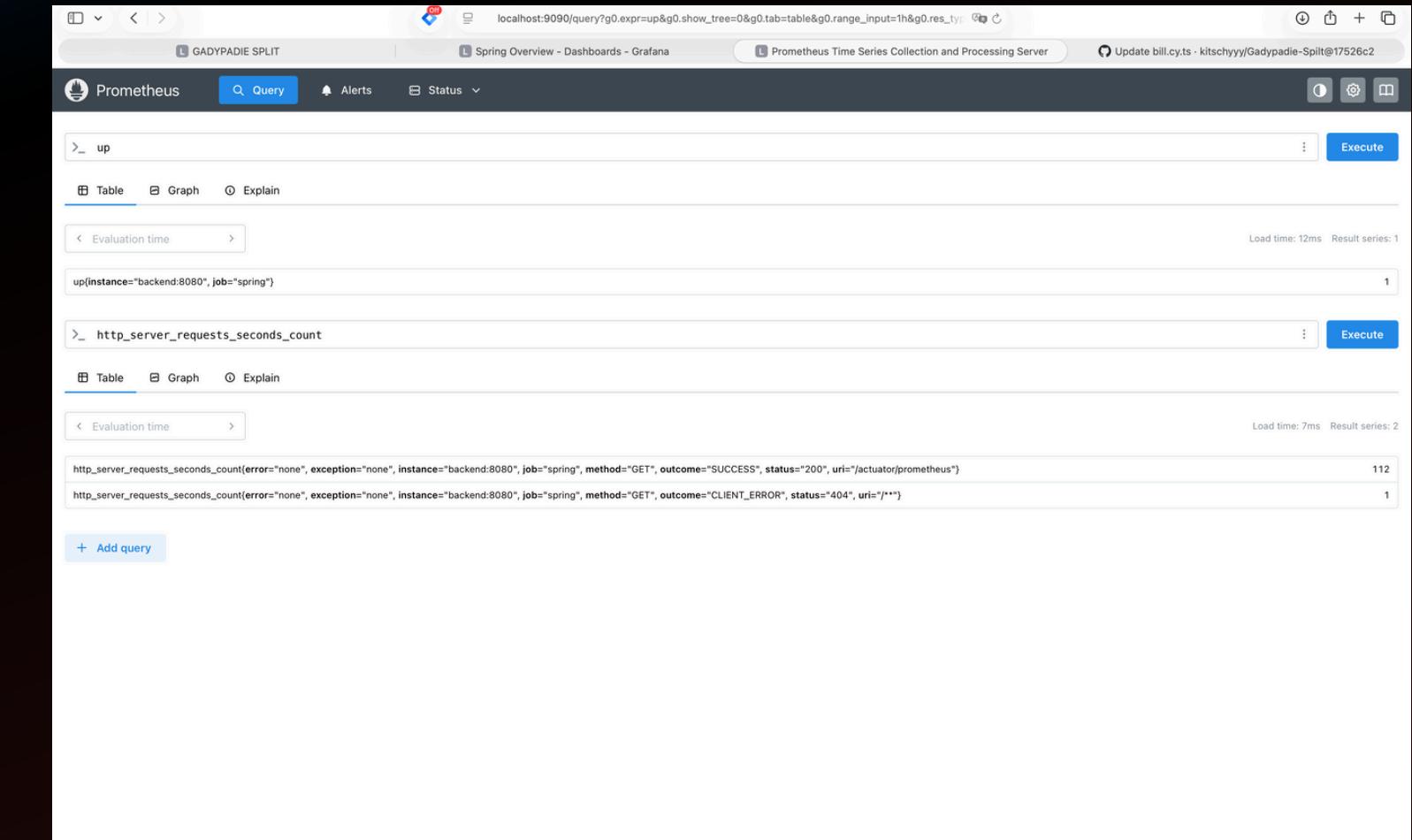
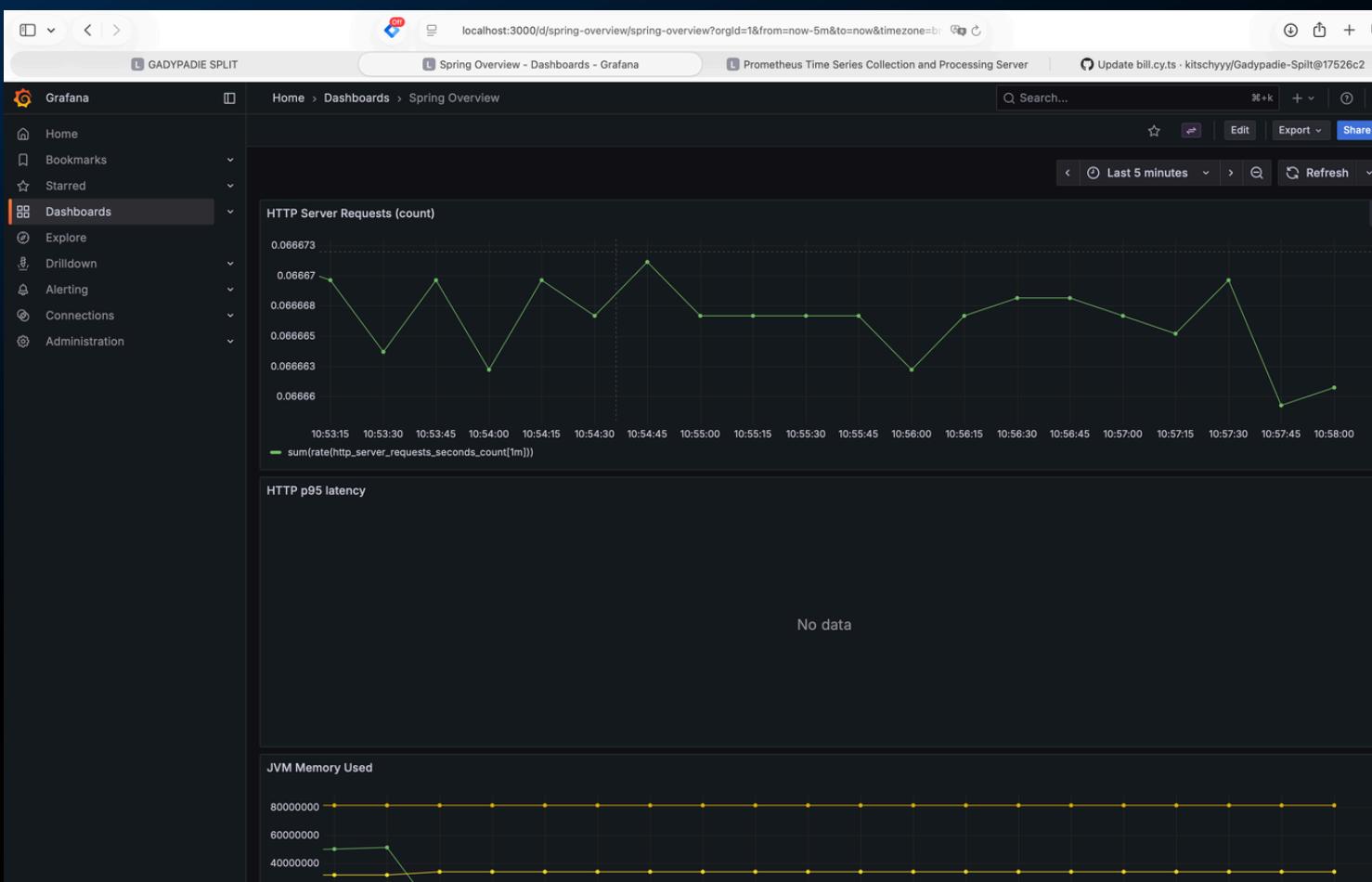
- สร้าง Docker Image ของ Frontend (Vue + Vite + Nginx)
- Push เข้า Registry เดียวกับ Backend
- ใช้ Tag เดียวกับ Backend เพื่อให้ Version สอดคล้องกัน
- Image ที่ Build แล้วพร้อม Deploy ทันที

### Build & Push Backend

- สร้าง Docker Image ของ Backend (Spring Boot API)
- Login เข้า Registry (Docker Hub หรือ GitHub Packages)
- ใช้ Tag จาก Commit เช่น `{{ github.sha }}`
- เก็บ Image ที่ผ่านการทดสอบแล้วสำหรับใช้ใน Kubernetes

## MONITORING &amp; OBSERVABILITY

- ตรวจสอบสถานะ CI/CD Pipeline ผ่าน GitHub Actions Dashboard
- ดู Logs ของ Backend และ Frontend ผ่าน Kubernetes Pods
- ตรวจสอบสถานะของ Service, Pod และ Deployment ผ่าน Dashboard หรือ kubectl CLI
- เฝ้าระวัง Error และ Real-time หลังการ Deploy ระบุ



## K8S DEPLOYMENT FILE

ระบบของเราแบ่งออกเป็น 3 ส่วนหลัก ได้แก่ Frontend (Vue + Vite + Nginx), Backend (Spring Boot + Gradle) และ Database (PostgreSQL)

### backend-deployment.yaml

- สร้างและควบคุมการรัน Pod ของ Backend (Spring Boot API)

### frontend-configmap.yaml

- เก็บค่าคอนฟิกของ Frontend เช่น API Base URL

### frontend-service.yaml

- สร้าง Service เชื่อมระหว่าง Ingress และ Pod ของ Frontend



### backend-ingress.yaml

- กำหนด Ingress สำหรับเปิดทางเข้าถึง API ภายนอก Cluster

### frontend-deployment.yaml

- สร้าง Deployment สำหรับเว็บแอป (Vue + Vite + Nginx)

### frontend-ingress.yaml

- เปิดทางเข้าเว็บหลักของระบบผ่านโดเมน

### postgres-deployment.yaml

- ติดตั้งและจัดการ Pod ของฐานข้อมูล PostgreSQL ภายใน Cluster

## CONCLUSION & FUTURE WORK

### Conclusion

- ระบบ GPSplit รองรับการแบ่งค่าใช้จ่ายหลายรูปแบบ (Equal, Amount, Percent, Tag-Based)
- รองรับ Multi-Currency พร้อมเก็บอัตราเดิม + อัปเดตอัตราผ่าน CSV ได้
- มีการอัปโหลดสลิป/ใบเสร็จ + แสดงตัวอย่าง
- มี Smart Settlement ช่วยคำนวณจำนวนการโอนที่น้อยที่สุด
- ระบบ CI/CD ช่วย Build, Test และ Deploy อัตโนมัติผ่าน Docker + GitHub Runner

### Future Work

- เพิ่ม OCR ดึงข้อมูลจากใบเสร็จเพื่อลดการกรอก (Offline Model)
- เพิ่ม Automatic Suggestion for Split (พกติกรรมการกิน/จ่าย)

# THANK YOU

KITSCHYYY

```
Terminal Local (3) × Local (2) × Local (4) × Local (5) × Local (7) × Local (8) × Local (10) × Local (6) × Local (9) × Local (11) × Local (12) × + ▾

babyearn@BabyEarns-MacBook-Pro gpsplit % kubectl get pods -n gpsplit
NAME           READY   STATUS    RESTARTS   AGE
backend-546f49878f-wcs7w   1/1     Running   0          9m6s
gpsplit-frontend-77cdb54ccc-25ll2  1/1     Running   0          26m
mysql-dc4584fd4-q4phk      1/1     Running   0          145m

babyearn@BabyEarns-MacBook-Pro gpsplit %
```

```
2025-11-22T08:36:47.140+07:00 INFO 61392 --- [gps-backend] [main] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000489: No JTA platform available (set 'hibernate.transaction.jta.platform' to enable JTA platform integration)
2025-11-22T08:36:47.204+07:00 INFO 61392 --- [gps-backend] [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
2025-11-22T08:36:47.644+07:00 INFO 61392 --- [gps-backend] [main] o.s.b.a.e.web.EndpointLinksResolver : Exposing 3 endpoints beneath base path '/actuator'
2025-11-22T08:36:47.674+07:00 INFO 61392 --- [gps-backend] [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8080 (http) with context path '/'
2025-11-22T08:36:47.683+07:00 INFO 61392 --- [gps-backend] [main] c.gpsplit.backend.GpsBackendApplication : Started GpsBackendApplication in 2.024 seconds (process running for 2.15)
^C2025-11-22T08:37:12.755+07:00 INFO 61392 --- [gps-backend] [ionShutdownHook] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistence unit 'default'
2025-11-22T08:37:12.758+07:00 INFO 61392 --- [gps-backend] [ionShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown initiated...
2025-11-22T08:37:12.764+07:00 INFO 61392 --- [gps-backend] [ionShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown completed.

[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] -----
[INFO] Total time: 28.543 s
[INFO] Finished at: 2025-11-22T08:37:12+07:00
[INFO]

babyearn@BabyEarns-MacBook-Pro backend % ./mvnw jacoco:report
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.gpsplit:backend >-----
[INFO] Building backend 1.0.0
[INFO]   from pom.xml
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- jacoco:0.8.11:report (default-cli) @ backend ---
[INFO] Skipping JaCoCo execution due to missing execution data file.
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] -----
[INFO] Total time: 0.235 s
[INFO] Finished at: 2025-11-22T08:44:34+07:00
[INFO]

babyearn@BabyEarns-MacBook-Pro backend %
```