



รายงาน

## Architectural Drivers Food Delivery System

จัดทำโดย

นายจิรกิตติ์ คำป่าตัน 67543210014-6

นายณัฐกิตต์ แก้วคำยศ 67543210055-9

นางสาวกฤตพร แหลมไทย 67543210051-8

นายนริพัทธ์ สุวรรณ 67543210064-1

เสนอ

อาจารย์ธนิต เกตุแก้ว

ภาคเรียนที่ 2 ปีการศึกษา 2568

วิชา Software Architecture

มหาวิทยาลัย เทคโนโลยีราชมงคลล้านนา

## 1. System Overview

### 1. System Description

QuickBite Delivery System เป็นระบบสั่งอาหารออนไลน์ที่ช่วยให้ผู้ใช้สามารถค้นหาร้านอาหารใกล้เคียง เลือกเมนู สั่งอาหาร ชำระเงินผ่านระบบ และติดตามสถานะคำสั่งซึ่งแบบเรียลไทม์ ระบบรองรับทั้งผู้ใช้งานทั่วไป ร้านอาหาร และไกด์เดอร์ที่ทำการจัดส่ง ช่วยให้กระบวนการสั่งอาหารมีความสะดวก รวดเร็ว และเข้าถึงได้ทุกที่

ระบบเน้นถูกออกแบบมาเพื่อลดความยุ่งยากในการสั่งอาหาร และเพิ่มความสะดวกให้ทั้งลูกค้าและร้านอาหาร โดยจัดการตั้งแต่การเลือกเมนู การชำระเงิน การมอบหมายไกด์เดอร์ ไปจนถึงการจัดส่งอาหารจนถึงมือลูกค้า ระบบยังมีการแจ้งเตือนสถานะตามเวลาจริง พร้อมฟีเจอร์รีวิวร้านอาหารเพื่อเพิ่มความโปร่งใสในการใช้งาน

ระบบเหมาะสมกับผู้ใช้ทั่วไปที่ต้องการสั่งอาหารอย่างรวดเร็ว ร้านอาหารที่ต้องการช่องทางในการขายอาหารออนไลน์ และไกด์เดอร์ที่ต้องการรับงานจัดส่งแบบบีดหยุ่น

### 2. Target Users

- ผู้ใช้งานทั่วไป (Customers)
- ร้านอาหารพันธมิตร (Merchants)
- ไกด์เดอร์ผู้สั่งอาหาร (Delivery Riders)
- ผู้ดูแลระบบ (System Admin)

### 3. Key Features

1. ลงทะเบียนและเข้าสู่ระบบสำหรับผู้ใช้ ร้านอาหาร และไกด์
2. ค้นหาร้านอาหารตามพื้นที่ / ประเภทอาหาร / คะแนนรีวิว
3. ระบบสั่งอาหารพร้อมตักร้านค้า (Cart)
4. ระบบติดตามสถานะคำสั่งชื่อแบบ Real-time
5. ระบบชำระเงินออนไลน์ (PromptPay, Credit/Debit Card)
6. การแจ้งเตือน (Push/In-app) เกี่ยวกับสถานะคำสั่งชื่อ
7. ร้านอาหารเพิ่ม/แก้ไขเมนูและราคาได้
8. ไกด์รับงานและอัปเดตสถานะ (“รับอาหารแล้ว / กำลังจัดส่ง / ส่งสำเร็จ”)
9. ระบบรีวิวและให้คะแนนร้านอาหาร/ไกด์
10. แพลตฟอร์ม Admin สำหรับจัดการผู้ใช้ ร้านค้า และตรวจสอบคำสั่งชื่อ

## 2.Functional Requirements

### 1.User Management

- FR-01: ผู้ใช้สามารถสมัครสมาชิกผ่านอีเมล/เบอร์โทรศัพท์ได้
- FR-02: ผู้ใช้สามารถเข้าสู่ระบบด้วยอีเมลและรหัสผ่านได้
- FR-03: ผู้ใช้สามารถรีเซ็ตรหัสผ่านผ่านอีเมลได้
- FR-04: ผู้ใช้สามารถแก้ไขโปรไฟล์ส่วนตัวได้

### 2.Core Ordering Features

- FR-05: ผู้ใช้สามารถค้นหาร้านอาหารตามตำแหน่งปัจจุบันได้
- FR-06: ผู้ใช้สามารถเพิ่มเมนูลงในตัชกร้าสินค้าได้
- FR-07: ผู้ใช้สามารถสั่งอาหารพร้อมเลือกที่อยู่สำหรับจัดส่งได้
- FR-08: ระบบต้องสามารถคำนวณราคาสุทธิรวมค่าส่งและแสดงยอดรวมได้
- FR-09: ผู้ใช้สามารถติดตามสถานะคำสั่งซื้อแบบเรียลไทม์ได้

### 4.Payment

- FR-10: ผู้ใช้สามารถชำระเงินผ่าน PromptPay / Credit/Debit Card ได้
- FR-11: ระบบต้องยืนยันสถานะการชำระเงินจาก Payment Gateway

### 5.Merchant Features

- FR-12: ร้านอาหารสามารถเพิ่ม/แก้ไขเมนู ซึ่ง ราคา และรูปภาพได้
- FR-13: ร้านอาหารสามารถดำเนินการขายออนไลน์ได้

## 5.Rider Features

FR-14: ไรเดอร์สามารถรับงานที่อยู่ใกล้ตัวแทนปัจจุบันได้

FR-15: ไรเดอร์สามารถอัปเดตสถานะการจัดส่งได้ (“รับอาหารแล้ว / กำลังจัดส่ง / ส่งสำเร็จ”)

## 6.Reporting

FR-16: ร้านอาหารสามารถดูยอดขายรายวัน/รายสัปดาห์ได้

FR-17: Admin สามารถดูสถิติคำสั่งซื้อทั้งหมดในระบบได้

### 3.Quality Attributes & Scenarios

เลือก 6 ตัว: Performance, Scalability, Availability, Security, Usability, Modifiability

#### 1. Performance Scenario

Source	ผู้ใช้ 1,000 คน
Stimulus	กดปุ่ม “สั่งอาหาร” พร้อมกัน
Artifact	Order Service, Database
Environment	ช่วงเที่ยง (peak time)
Response	ระบบประมวลผลคำสั่งซื้อและสร้าง order สำเร็จโดยไม่มี error
Response Measure	Response Time $\leq$ 2 วินาที สำหรับ 95% ของคำขอ

#### 2. Scalability Scenario

Source	ระบบมีจำนวนผู้ใช้เพิ่มจาก 10,000 $\rightarrow$ 100,000 คน
Stimulus	มีการใช้งานพร้อมกันเพิ่มขึ้น 10 เท่า
Artifact	API Gateway, Backend Services
Environment	Normal Operation
Response	ระบบสามารถเพิ่มจำนวน instance อัตโนมัติผ่าน Auto-scaling
Response Measure	Handling Capacity เพิ่ม $\geq$ 10 เท่า โดยไม่ลดคุณภาพบริการ

### 3. Availability Scenario

Source	ผู้ใช้ทั่วไป
Stimulus	พยายามใช้งานระบบในช่วงเวลาทำงาน
Artifact	Frontend, Backend, Database
Environment	24/7
Response	ระบบพร้อมให้บริการต่อเนื่อง
Response Measure	Uptime $\geq$ **99.9% ต่อเดือน

### 4. Security Scenario

Source	แฮกเกอร์
Stimulus	พยายามเข้าถึงข้อมูลผู้ใช้โดยไม่ได้รับอนุญาต
Artifact	Authentication Service, User Database
Environment	Internet
Response	ระบบปฏิเสธคำขอ ผูก Token หมดอายุ และบันทึก log
Response Measure	ไม่มีข้อมูลผู้ใช้รั่วไหล / บล็อกการโจมตีภายใน 1 วินาที

## 5. Usability Scenario

Source	ผู้ใช้ใหม่
Stimulus	เปิดแอปครั้งแรกเพื่อสั่งอาหาร
Artifact	UI/UX
Environment	Mobile Device
Response	ผู้ใช้สามารถสั่งอาหารได้สำเร็จโดยไม่ต้องอ่านคู่มือ
Response Measure	Task Success Rate $\geq$ **90%**, เวลาในการสั่งครั้งแรก $\leq$ 2 นาที

## 6. Modifiability Scenario

Source	ทีมพัฒนา
Stimulus	ต้องเพิ่มบริการใหม่ เช่น ส่วนลดสำหรับสมาชิก VIP
Artifact	Promotion Service
Environment	During Development
Response	เพิ่มฟีเจอร์ใหม่โดยแก้โค้ดเฉพาะส่วน
Response Measure	ไม่มีข้อมูลผู้ใช้ร่วม / บล็อกการโจนตีภายใน 1 วินาที

## 4. Constraints

### 1. Technical Constraints

- ต้องพัฒนาด้วย Web/Mobile-based (React, Flutter หรือเทคโนโลยีตามที่อาจารย์กำหนด)
- Backend ต้องใช้ REST API
- ต้องใช้ฐานข้อมูล SQL (MySQL/PostgreSQL)
- Deployment ใช้บริการ Cloud Free Tier (เช่น Render, Railway, Supabase)

### 2. Time Constraints

- ต้องพัฒนาโครงการให้เสร็จภายใน 6–8 สัปดาห์
- สมาชิกทีมมีเวลาวันละ 1–2 ชั่วโมงเท่านั้นเนื่องจากเรียนหลายวิชา

### 3. Budget Constraints

- ใช้เฉพาะ Cloud แบบ Free Tier
- ไม่มีงบซื้อ Server หรือบริการ Third-party แบบเสียเงิน

### 4. Legal/Policy Constraints

- ต้องปฏิบัติตาม PDPA/Privacy Policy เกี่ยวกับข้อมูลผู้ใช้
- ข้อมูลบัตรเครดิตต้องไม่ถูกเก็บโดยระบบ (ใช้ Payment Gateway เท่านั้น)

## 5.Assumptions

- 1.ผู้ใช้มีสมาร์ตโฟนและอินเทอร์เน็ตความเร็วปานกลางขึ้นไป
- 2.ผู้ใช้ส่วนใหญ่ใช้งานในช่วงเวลา 11:00–13:00 และ 17:00–19:00
- 3.ระบบเวอร์ชันแรกจะรองรับเฉพาะพื้นที่ในเมืองใหญ่
- 4.การจ่ายเงินทั้งหมดผ่าน Payment Gateway ภายนอก
- 5.ไรเดอร์มีแอปมือถือที่สามารถเชื่อมต่อได้

## 6.Priority & Trade-offs

### Quality Attributes Priority

อันดับ	Quality Attribute	เหตุผล
1	Availability	ระบบต้องใช้งานได้ตลอดเวลาในช่วงสั่งอาหาร ถ้าล้ม = เสียโอกาสขายทันที
2	Performance	ต้องตอบสนองเร็วเพื่อไม่ให้ลูกค้าเลิกสั่ง
3	Security	มีข้อมูลส่วนตัวและการชำระเงิน ต้องปลอดภัยสูง
4	Scalability	ต้องรองรับผู้ใช้เพิ่มโดยเฉพาะช่วงเทศกาล
5	Usability	ผู้ใช้ควรสั่งอาหารได้ง่ายที่สุด
6	Modifiability	เพื่อขยายฟีเจอร์ในอนาคต

## Trade-offs Analysis

### Trade-off #1: Performance vs Security

- การเข้ารหัสข้อมูลและ Authentication หลายชั้นช่วยเพิ่มความปลอดภัย แต่ทำให้ Response Time เพิ่มขึ้น เช่น การตรวจสอบ Token ทุก request
- หากลด Security → ระบบเร็วขึ้นแต่เสี่ยงข้อมูลรั่วไหล

### Decision:

เราเลือก “ความปลอดภัยสูงขึ้นแม้จะเพิ่มเวลา response เล็กน้อย (จาก 1.5 → 2 วินาที)”

เพราะระบบเกี่ยวข้องกับข้อมูลสำคัญ เช่น ที่อยู่ผู้ใช้และการชำระเงิน