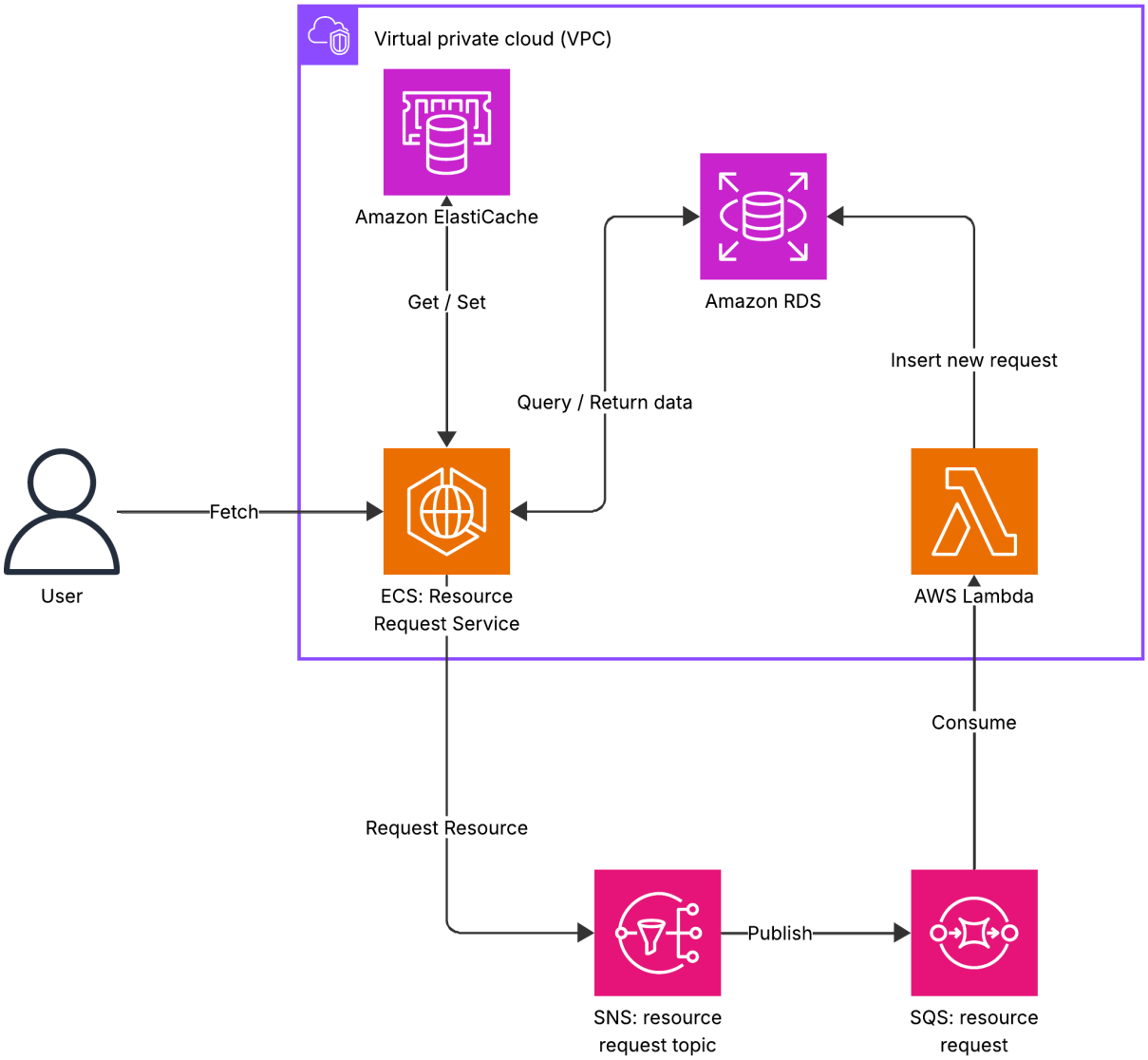


# Service Architecture

## Components

- **User (Client):** ผู้ใช้งานระบบที่ส่งคำขอ เพื่อดึงข้อมูลหรือสร้างคำร้องขอใหม่
- **ECS: Resource Request Service (Backend):** บริการประมวลผลหลักที่อยู่ภายใน VPC ทำหน้าที่รับคำขอจาก User หรือ Service อื่น ๆ จัดการ Business logic, Cache, Query database และส่งต่อ Event คำร้องขอใหม่ไปยัง Queue
- **Amazon ElastiCache:** ระบบหน่วยความจำแคช (In-memory Cache) ที่ใช้สำหรับบันทึกและดึงข้อมูลที่ถูกเรียกใช้บ่อย (Get / Set) เพื่อเพิ่มความเร็วในการตอบสนองแก่ User และลดปริมาณการคิวรีฐานข้อมูลโดยตรง โดยตอนนี้จะใช้เฉพาะกับ idempotency key

- **Amazon RDS:** ฐานข้อมูลเชิงสัมพันธ์ (Relational Database) หลักของระบบ ทำหน้าที่จัดเก็บข้อมูลทรัพยากรและคำร้องขอต่างๆ รองรับการดึงข้อมูล (Query / Return data) จาก ECS และการเพิ่มข้อมูลใหม่ (Insert new request) จาก AWS Lambda
- **SNS: resource request topic:** บริการกระจายข้อความแบบ Pub/Sub ทำหน้าที่รับ Event "Request Resource" จากบริการหลัก (ECS) และนำไป Publish กระจายต่อให้ระบบที่เกี่ยวข้อง
- **SQS: resource request:** คิวสำหรับพักข้อความ (Message Queue) ที่รับ Event มาจาก SNS เพื่อจัดลำดับและเก็บข้อความไว้รอการประมวลผล ป้องกันไม่ให้ระบบหลังบ้านทำงานหนักเกินไปเมื่อมีคำขอเข้ามาพร้อมกันจำนวนมาก
- **AWS Lambda (Async Worker):** ฟังก์ชันที่ทำงานแบบ Serverless คอยทำงานอยู่เบื้องหลัง ทำหน้าที่ดึงข้อความคำร้องขอ (Consume) จาก SQS มาประมวลผล และทำการบันทึกข้อมูลคำร้องขอใหม่ (Insert new request) ลงในฐานข้อมูล Amazon RDS
- **Virtual Private Cloud (VPC):** เครือข่ายส่วนตัวเสมือนที่ครอบคลุมบริการหลัก ได้แก่ ECS, ElastiCache และ Amazon RDS เพื่อจำกัดขอบเขตการเข้าถึงและเพิ่มความปลอดภัยให้กับข้อมูลสำคัญ

---

## Explanation

สถาปัตยกรรมของ **Resource Request Service** นี้ ถูกออกแบบมาเพื่อเพิ่มประสิทธิภาพในการรองรับโหลดข้อมูลระดับหนึ่ง มีการนำระบบ Cache มาช่วยลดภาระของฐานข้อมูลหลัก การทำงานแบ่งออกเป็น 2 ส่วนหลัก ดังนี้:

- **Synchronous Flow (สำหรับการเรียกดูและค้นหาข้อมูล):** เมื่อ User ส่งคำสั่ง "Fetch" เข้ามายัง ECS ระบบจะทำงานแบบตอบกลับทันที ติดต่อกับฐานข้อมูลโดยตรง ในอนาคตวางแผนที่จะให้ ECS จะไปตรวจสอบข้อมูลที่ Amazon ElastiCache ก่อน แล้วค่อยไปค้นหาใน RDS แล้วนำผลลัพธ์มาบันทึกเก็บไว้ในแคชก่อนที่จะตอบกลับไปยัง User
- **Asynchronous Flow (สำหรับการสร้างคำร้องขอทรัพยากรใหม่):** เมื่อระบบได้รับคำสั่งสร้างคำร้องขอใหม่ (Request Resource) แทนที่ ECS จะเขียนข้อมูลลงฐานข้อมูลโดยตรง (ซึ่งอาจทำให้ระบบเกิดการล็อกหรือล่าช้า) ECS จะส่งข้อมูลนั้นเป็น Event ไปยัง SNS (resource request topic) แทน จากนั้น SNS จะทำการกระจายข้อความ (Publish) ไปพักไว้ที่คิว Amazon SQS เมื่อข้อความอยู่ในคิวแล้ว AWS Lambda จะคอยดึงข้อความเหล่านั้น (Consume) ไปประมวลผลตามลำดับแบบเบื้องหลัง แล้วจึงนำข้อมูลที่ประมวลผลเสร็จแล้วไปบันทึกลงฐานข้อมูล Amazon RDS (Insert new request)