

Full-stack developer assignment (Next.js + NestJS)

ข้อแนะนำ:

- งานชิ้นนี้ควรทำเสร็จได้ภายใน 1.5 - 2 ชั่วโมง เราไม่แนะนำให้ใช้เวลาเกินกว่านี้
- กรุณาอ่านเกณฑ์การให้คะแนนด้านล่างก่อน

งานที่ต้องทำ

งานที่ 1: พื้นฐาน

- เราต้องการจะสร้างเว็บบอร์ดสำหรับให้ผู้ใช้งานสามารถตั้งคำถาม แสดงความคิดเห็นได้
- สร้าง Project ใหม่แยก Frontend และ Backend อย่างชัดเจน โดยใช้ NextJS สำหรับหน้าบ้าน และใช้ NestJS สำหรับ API Server (หลังบ้าน)

งานที่ 2: การคำนึงถึงผู้ใช้งาน

- ออกแบบงานให้สอดคล้องกับการออกแบบใน Figma ตามไฟล์ที่ให้โดยคำนึงถึงการแสดงผลบนอุปกรณ์ต่างๆ ไม่ว่าจะเป็นในโทรศัพท์ แท็บเล็ต หรือหน้าจอคอมพิวเตอร์
- ใช้ CSS/HTML สำหรับการแสดงผลทางหน้าบ้าน โดยสามารถใช้ CSS frameworks ช่วยได้ เช่น Tailwind, Bootstrap และสามารถนำมาปรับใช้เป็นการแสดงผลที่ต้องการได้

งานที่ 3: การบริหารหรือใน board - CRUD

Backend

สร้าง RESTful API โดยใช้ NestJS เพื่อทำ CRUD operations:

- User
 - สามารถกรอก username เพื่อเข้าสู่ระบบ
 - ไม่จำเป็นต้องมีระบบยืนยันอีเมล และระบบลืมรหัสผ่าน
- Posts
 - ดูรายการคำถามที่คนอื่นสร้างได้
 - สามารถสร้าง topic ใหม่ เพื่อพูดคุยหรือตั้งคำถามใน community
 - สามารถ comment ใน topic ต่างๆ ได้
 - Comment ได้เพียง 1 Level (ไม่ต้องทำ Comment Nested)
 - สามารถแก้ไขหรือลบ topic และ comment เฉพาะของ user เท่านั้น
- กรณีคุณใส่ Features อื่นๆ นอกจากนี้ให้ระบุใน README เพื่อให้ทางบริษัทฯ ตรวจสอบเพิ่มเติมได้ แต่คะแนนส่วนหลักจะอยู่ที่ความถูกต้องของส่วน Requirement หลัก

Frontend

- หน้าแสดงกระดาน Board
- หน้าสำหรับสร้างกระดานใหม่ (ต้อง Login ก่อน)
- หน้าแสดงรายละเอียดของกระดาน และแสดงรายการความคิดเห็น
- สามารถสร้างความคิดเห็นใหม่ได้ (ต้อง Login ก่อน)

งานที่ 4: การจัดการข้อผิดพลาด (Error) ในส่วนของ Server

- พัฒนาการตรวจสอบฝั่ง Server หาก HTTP Request ไม่ถูกต้อง เช่น Missing title or content และฝั่ง server ควรจะส่งข้อความเพื่อบ่งบอกว่ามี error เกิดขึ้น
- Error ที่เกิดขึ้นต้องนำมาแสดงผลให้กับ User

งานที่ 5: Unit test

- ต้องมี unit test โดยเฉพาะตัวจัดการการดำเนินการ CRUD
- ในส่วนของหน้าบ้านนั้นไม่จำเป็น แต่ถ้ามีจะถือเป็นคะแนนพิเศษ

การส่งงาน

ส่งลิงค์ GitHub/Bitbucket repository ที่เข้าถึงแบบสาธารณะได้ มี code สำหรับการบ้าน และในการทำการบ้านนั้นไม่ควรมี commit ที่ใหญ่เกินไป และ commit message ควรสอดคล้องกับสิ่งที่ได้ commit ไป โปรดหลีกเลี่ยงการทำ commit ขนาดใหญ่

ในไฟล์ README file

- บอกขั้นตอนในการติดตั้งเพื่อทดสอบให้ชัดเจน
- ภาพรวมการออกแบบของ application architecture.
- พวก libraries หรือ packages ต่างๆ ที่ใช้ พร้อมคำอธิบาย
- วิธีการ run unit test

เกณฑ์การพิจารณา

- (30%) ความถูกต้องและสมบูรณ์แบบ: สามารถตอบโจทย์ requirements ต่างๆ ได้ไหม
- (25%) การเขียน code และโครงสร้าง: code ถูกออกแบบมาดีไหม อ่านง่ายหรือไม่ สามารถให้คนมาดูแลต่อได้ไหม
- (20%) การออกแบบโดยคำนึงถึงผู้ใช้งาน: การแสดงผลตอบโจทย์สำหรับหลายๆ อุปกรณ์หรือไม่
- (5%) การจัดการ error: สามารถจัดการ error ได้ดีแค่ไหน ทั้งหน้าบ้านและหลังบ้าน
- (15%) การ test: test case ครอบคลุมหรือไม่ และผ่านไหม
- (5%) เอกสาร: ไฟล์ README ครบถ้วนหรือไม่
- (0-5%) คะแนนพิเศษ กรณีทำเกิน Requirement และอธิบายได้ดีใน README และสมเหตุสมผล

หมายเหตุ: การบ้านนี้ควรใช้เวลาไม่เกิน 2-3 ชั่วโมง หากคุณติดอยู่กับงานหนึ่งนานเกินไป พยายามทำงานอื่นๆ ให้ได้มากที่สุด การจัดสรรเวลากับงานเป็นอีกงานที่เราให้ความสำคัญไม่แพ้กัน ทั้งนี้เราอาจจะไม่สามารถให้ Feedback ให้กับผู้สมัครทุกท่าน กรณีต้องการ Feedback สามารถแจ้งใน Readme ไปได้ หากทีมงานที่ตรวจของเราสามารถให้ได้ จะดำเนินการส่ง Feedback ให้กับท่าน