

05.06.2025



משימת תכנות: פלטפורמת למידה מונעת בינה מלאכותית (Mini MVP)

תפקיד: מפתח Full-stack
פורמט הגשה: מאגר GitHub ציבורי עם תיעוד ושימוש ברורים
הוראות.

מטרה

צור פלטפורמת למידה מיניאטורית המאפשרת למשתמשים לבחור מה הם רוצים ללמוד (על ידי קטגוריה ותת-קטגוריה), לשלוח הנחיות לבינה מלאכותית כדי לקבל שיעורים שנוצרו, ו לצפות בהיסטוריית הלמידה שלהם. הפלטפורמה תכלול קצה אחורי של REST API מסד נתונים, בינה מלאכותית אינטגרציה, וממשק משתמש בסיסי של לוח מחוונים באינטרנט.

משימה זו נועדה להעריך את כישוריך בארכיטקטורת תוכנה, אינטגרציית API, מודולריות ארגון קוד ואיכות אספקה.

דרישות טכניות

קצה אחורי

- שפה: Python (FastAPI), Go, או כל שרת backend אחר
- שפה/מסגרת שאתה מעדיף - כל עוד היא ברמת ייצור.
- מסד נתונים: PostgreSQL, MySQL או MongoDB (השתמשו ב-ORM בבמידת הצורך)
- שרת REST API עם נתיבים מובנים.
- שילוב בינה מלאכותית:
- השתמש ב-API בשל OpenAI GPT (או דמה לתגובה מקומית עם אינטגרציה ברורה)
- ממשק (ממשק)
- שלח את הנושא וההנחיה שבחר המשתמש, וקבל שיעור דמוי שיעור תגובה.
- מודלים / טבלאות:
- משתמשים: מזהה, שם, טלפון
- קטגוריות: תעודת זהות, שם
- תת-קטגוריות: מזהה, שם, מזהה קטגוריה
- הנחיות: id, user_id, category_id, sub_category_id, תגובה, נוצר ב
- משתמשים יכולים לקבל מספר הנחיות (היסטוריית למידה)
- שמירה על קשרי מסד נתונים ואילוצים תקינים
- שפה: React, Vue, או כל מסגרת קצה מודרנית אחרת
- צור לוח מחוונים פשוט באינטרנט התומך ב:
- רישום משתמשים חדשים
- בחירת קטגוריה ותת-קטגוריה
- הגשת בקשה
- הצגת התגובה שנוצרה על ידי בינה מלאכותית
- צפייה בהיסטוריית הלמידה של המשתמש
- לוח בקרה של ניהול לרישום כל המשתמשים וההיסטוריית ההנחיות שלהם

05.06.2025



דרישות חובה

- מבנה פרויקט מאורגן עם שכבות מופרדות בבירור (נתיבים/בקרים/מודלים/שירותים/וכו')
- נקה את קובץ README-העם:
- הוראות התקנה
- טכנולוגיות בשימוש
- הנחות שנעשו
- איך להריץ באופן מקומי (frontend i-backend)
- קובץ דוגמה. vite
- השתמש ב-rekcoD או ב-Compose rekcoD כדי להפעיל את מסד הנתונים במידת הצורך
- מאגר GitHub חייב להיות ציבורי עם היסטוריית commit ברורה
- הקוד צריך להיות נקי, מתועד היטב ולעקוב אחר שיטות העבודה המומלצות
- טיפול באימות קלט בסיסי ובטיפול בשגיאות API
- השתמש ב-dotenv באו בניהול תצורה
- בונס (אופציונלי אך מרשים)
- שימוש TypeScript בבקצה הקדמי או האחורי
- אימות משתמש מבוסס JWT
- עימוד וסינון בלוח המחוונים של המנהל
- בדיקות יחידה/אינטגרציה (באמצעות Jest, Pytest וכו')
- תיעוד Swagger/OpenAPI
- פריסת הדגמה עובדת (למשל, דרך Vercel, Netlify או Heroku)

מקרה שימוש לדוגמה

ישראל נרשם ובוחר ללמוד על מדע חלל. הוא מזין את הנחיה: "למד אותי" "על חורים שחורים." המערכת מאחסנת את הקלט שלו, שולחת אותו למודל בינה מלאכותית ומחזירה שיעור. הוא יוכל לחזור ללוח המחוונים מאוחר יותר כדי לצפות בכל השיעורים שקיבל.

הערכת זמן

ניתן להקדיש עד 2-3 ימים למשימה זו, בהתאם לזמינותכם. איכות ובהירות חשובים יותר משלמות. אנא התמקדו בארכיטקטורה, תחזוקה ובהירות.

קניעה

- העלאה למאגר GitHub ציבורי
 - שתף את הקישור יחד עם הערה קצרה (1-2 פסקאות) המסבירה את שלך גישה
 - לאחר שתסיים, אנא שלח את המשימה שלך באופן הבא:
נושא האימייל: שם פרטי שם משפחה
גוף: קישור https למאגר GitHub (ודא שהוא ציבורי)
קורות חיים: קורות חיים מצורפים בפורמט PDF
- מכתב הערה: כתבו סיפור קצר על עצמכם

05.06.2025



אישורים:

OPENAI_API_KEY=sk-proj-aubFAcg3pLE4sSVBOzbzS1IIiYcNdf1WQSt1cc_MXaBuT1B31CuAdxEGFMu
_V9NVxHy97I8jET3BibkFJJL0V5T5UsVxmrc7SBtMrPwdLk4gbexFaQu_Q40hCjFqHhxx9o_yEgZiYUAFLjFpeX10qvsB8A

אנחנו מחפשים קוד נקי ומודולרי, יכולת פתרון בעיות יצירתית וחשיבה ארכיטקטונית חדה. התייחסו לזה כאל מערכת מיניאטורית ברמת ייצור - סוג העבודה שהייתם עושים עבור הלקוחות שלנו.

לכל שאלה או אי הבנה, אנא שלחו מייל לכתובת shimi@apisoul.com ואנחנו נגיב בהקדם האפשרי.

הערה: אנא אל תשתפו או תעבירו קובץ זה לאף גורם אחר.

בהצלחה!