

6 דרישות תרגיל הבית

6.1 מטרת התרגיל

בתרגיל זה תממשו סוכן שחקן לליגת זוגי/אי-זוגי. בשלב זה, הסוכן שלכם ירוץ בסביבה שלכם בלבד. מומלץ לתאם עם סטודנטים אחרים כדי לוודא תאימות פרוטוקול. **חשוב מאוד:** השתמשו בפרוטוקול המוגדר במסמך זה בדיוק. אחרת הסוכן שלכם לא יוכל לתקשר עם אחרים.

חובה לבנות ולתכנן את הפרויקט בכפוף להנחיות של פרק 9 (פרוטוקול נתוני הליגה), פרק 10 (ערכת כלים בפיתוח), ופרק 11 (מבנה הפרויקט). כמו כן יש לוודא כי הפרויקט רץ ותפקד כמוגדר בפרק 8 (הרצת מערכת הליגה). תרגיל זה מבוסס על הספר:

סוכני בינה מלאכותית עם פרוטוקול הקשר המודל

AI Agents with Model Context Protocol

מאת ד"ר יורם סגל

December 9, 2025

לכן מאוד מומלץ לקרוא וללמוד את הנושא לעומק.

6.2 משימות חובה

6.2.1 משימה 1: מימוש סוכן שחקן

ממשו שרת MCP שמאזין על פורט ב-localhost. השרת חייב לתמוך בכלים הבאים:

1. `handle_game_invitation` – קבלת הזמנה למשחק והחזרת `GAME_JOIN_ACK`.
2. `choose_parity` – בחירת "זוגי" או "אי-זוגי" והחזרת `CHOOSE_PARITY_RESPONSE`.
3. `notify_match_result` – קבלת תוצאת משחק ועדכון מצב פנימי.

6.2.2 משימה 2: רישום לליגה

הסוכן חייב לשלוח בקשת רישום למנהל הליגה. הבקשה תכלול:

- שם תצוגה ייחודי (השם שלכם או כינוי).
- גרסת הסוכן.
- כתובת ה-endpoint של השרת.

6.2.3 משימה 3: בדיקה עצמית

לפני ההגשה, בדקו את הסוכן שלכם:

1. הריצו ליגה מקומית עם 4 שחקנים.

2. ודאו שהסוכן מגיב לכל סוג הודעה.
3. ודאו שתבניות ה-JSON תואמות לפרוטוקול.

6.3 דרישות טכניות

6.3.1 שפת תכנות

אתם יכולים לבחור כל שפה שתמצאו. העיקר שהסוכן:

- מממש שרת HTTP.
- מגיב לבקשות POST בנתיב /mcp.
- מחזיר JSON בפורמט JSON-RPC 2.0.

6.3.2 זמני תגובה

- GAME_JOIN_ACK – תוך 5 שניות.
- CHOOSE_PARITY_RESPONSE – תוך 30 שניות.
- כל תגובה אחרת – תוך 10 שניות.

6.3.3 יציבות

הסוכן חייב:

- לפעול ללא קריסות.
- לטפל בשגיאות קלט.
- לא להפסיק לפעול באמצע ליגה.

6.4 תהליך העבודה

6.4.1 שלב 1: פיתוח מקומי

1. ממשו את הסוכן.
2. בדקו מקומית עם הקוד שלכם.
3. תקנו באגים.

6.4.2 שלב 2: ליגה פרטית

1. הריצו ליגה מקומית עם 4 עותקים של הסוכן.
2. בדקו שכל התקשורת עובדת.
3. שפרו את האסטרטגיה (אופציונלי).

6.4.3 שלב 3: בדיקת תאימות עם סטודנטים אחרים

1. תאמו עם סטודנט אחר להחלפת סוכנים.
2. בדקו שהסוכנים מתקשרים ביניהם כראוי.
3. ודאו שתבניות ה-JSON תואמות לפרוטוקול.

6.4.4 מבט לעתיד: ליגת כיתה

הערה חשובה

בעתיד, ייתכן שתדרשו:

- ליצור משחקים חדשים (לא רק זוגי/אי-זוגי).
- להתחרות בליגת כיתה כחלק מהפרויקט המסכם של הקורס.

נושא זה טרם נסגר וייתכנו שינויים. עליכם להיערך לכך ולבנות את הסוכן בצורה גמישה שתאפשר הרחבה עתידית.

6.5 הגשה

6.5.1 קבצים להגשה

1. קוד מקור של הסוכן.
2. קובץ README עם הוראות הרצה.
3. דוח מפורט הכולל:
 - תיאור מלא של הארכיטקטורה והמימוש.
 - תיאור האסטרטגיה שנבחרה והסיבות לבחירה.
 - קשיים שנתקלתם בהם והפתרונות שמצאתם.
 - תיעוד של תהליך הפיתוח והבדיקות.
 - מסקנות מהתרגיל והמלצות לשיפור.

6.5.2 פורמט הגשה

יש להגיש קישור לריפוזיטורי כשהוא ציבורי. ויש להגיש בנוהל ההגשה הרגיל כפי שהוגשו התרגילים הקודמים.

6.6 דגשים כלליים לבדיקת העבודה

מעבר לדרישות הרגילות, הקריטריונים הבאים ייבדקו:

טבלה 13: קריטריונים לבדיקה

קריטריון	תיאור
תפקוד בסיסי	הסוכן עובד, עונה להודעות, משחק במשחקים
תאימות פרוטוקול	תבניות JSON תואמות בדיוק לפרוטוקול
יציבות	הסוכן יציב, לא קורס, מטפל בשגיאות
איכות קוד	קוד נקי, מתועד, מאורגן
תיעוד	הוראות הרצה ברורות, תיאור מפורט
אסטרטגיה	מימוש אסטרטגיה מעניינת (לא רק אקראי)

6.7 שאלות נפוצות

6.7.1 האם אפשר להשתמש בספריות חיצוניות?

כן. אתם יכולים להשתמש בכל ספרייה שתמצאו. ודאו שאתם מספקים הוראות התקנה.

6.7.2 האם חייבים להשתמש ב-Python?

לא. השתמשו בכל שפה שמתאימה לכם. העיקר שהסוכן עומד בדרישות הפרוטוקול.

6.7.3 מה קורה אם הסוכן שלי קורס?

הסוכן יספוג הפסד טכני במשחק הנוכחי. אם הוא לא חוזר לפעולה – הוא יוצא מהליגה.

6.7.4 האם אפשר לעדכן את הסוכן אחרי ההגשה?

לא. ההגשה סופית. בדקו היטב לפני שאתם מגישים.

6.7.5 איך אדע מה הדירוג שלי?

טבלת הדירוג תפורסם לאחר כל מחזור. תוכלו לראות את המיקום של הסוכן שלכם.

6.8 סיכום

1. ממשו סוכן שחקן שעומד בפרוטוקול.
2. בדקו מקומית לפני הגשה.
3. הגישו את הקוד והדוח.
4. הסוכן שלכם ישחק בליגת הכיתה.

בהצלחה!

מידע נוסף:

- לשאלות ובירורים פנו לד"ר יורם סגל.
- מומלץ לקרוא את הספר "סוכני AI עם MCP" [1].
- לפרטים נוספים על פרוטוקול MCP ראו את התיעוד הרשמי [2].