

## מטלה 4

שם קוד הקבוצה - אריאל נפדנסקי ואבי פרדמן

ת.ז חברי הקבוצה-

1. 313306078 אריאל נפדנסקי

2. 316420132 אבי פרדמן

קישור לפרויקט -

<https://github.com/ArielleNapa/LLMs-and-Multi-Agent-Orchestration---Assignment2.git>

ציון מוערך - 85 😊

### טופס הגשת הערכה עצמית

שם הסטודנט/ים: אריאל נפדנסקי ואבי פרדמן

שם הפרויקט: מורה דרך

תאריך הגשה: 04.12.2025

הציון העצמי שלי: 85/100

הצדקה להערכה העצמית (חובה - 200-500 מילים)

בסעיף זה, הסבירו למה בחרתם בציון זה. כללו:

נקודות חוזק: מה עשיתם במיוחד טוב? אילו מרכיבים הם ברמה גבוהה?

נקודות חולשה: מה חסר או יכול היה להיות טוב יותר? (כנות מוערכת!)

השקעה: כמה זמן ומאמץ השקעתם?

חדשנות: האם יש משהו ייחודי או מיוחד בעבודה?

למידה: מה למדתם מהפרויקט?

### נקודות חוזק

1. תיעוד וארכיטקטורה מצוינים הפרויקט כולל תיעוד רחב היקף ברמה אקדמית גבוהה: PRD מקיף של +500 שורות, תיעוד ארכיטקטורה עם תרשימי C4 מלאים (4 רמות), README ברמת user manual, ו-10+ מסמכי documentation נוספים. כל מסמך מובנה היטב עם diagrams, טבלאות, ודוגמאות קוד.

2. קוד מקצועי עם modularity גבוהה המערכת מממשת ארכיטקטורת multi-agent מתוככמת  
עם 4 סוכנים מתמחים (YouTube, Music, Story, Judge), הפרדת אחריות מלאה, ו-1,247  
שורות קוד מתועדות. הקוד עוקב אחרי עקרונות SOLID, כולל type hints, ומשתמש ב design  
patterns מתקדמים (Factory, Strategy, Singleton).
3. מחקר אמיתי ומדידה מחברת prompt engineering עם +18 איטרציות מתועדות והשפעה  
מדודה (+15% consistency, +20% relevance). ניתוח עלויות מקיף עם חישובים פיננסיים  
מפורטים (\$0.53 לכל journey), תחזיות 3 שנים, וזיהוי אפשרויות אופטימיזציה (72% הפחתת  
עלויות).
4. בדיקות מקיפות 82 unit + integration tests עם 100% pass rate, כיסוי 79.16% (קרוב מאוד  
ל-80%), ומערכת בדיקות מובנית עם fixtures ו-5 mocks. ביצועים concurrency שימוש  
מתוככם ב-ThreadPoolExecutor לביצוע מקבילי של 3 סוכנים לכל waypoint, עם  
thread-safety מלא ו-30x שיפור ביצועים לעומת ביצוע סדרתי.

### נקודות חולשה (כנות)

1. פערי כיסוי בדיקות: Judge Agent רק 55.91% coverage ו-71.43% Story Agent, מה שמוריד  
את הכיסוי הכללי מתחת ל-80%. זה הפער המשמעותי ביותר.
2. העדר UI: אין ממשק משתמש גרפי - רק CLI. Web UI הוסר במהלך ניקיון הפרויקט.
3. סוגיות קטנות בהרחבה: למרות שהמערכת extensible, אין דוגמאות מעשיות להרחבת  
סוכנים חדשים.

### השקעה

השקענו כ-350 שעות בפרויקט (כמתועד בניתוח העלויות), כולל:

- תכנון וארכיטקטורה: 40 שעות
- פיתוח ליבה: 120 שעות
- Prompt engineering: 50 שעות
- בדיקות: 60 שעות
- תיעוד: 80 שעות

### חדשנות

- Multi-agent orchestration ראשון מסוגו המשלב ניווט עם AI-curated content
- Judge algorithm ייחודי עם ציון 4 (multi-criteria קריטריונים משוקללים)
- 3-tier caching system למיטוב עלויות
- Production-grade מההתחלה עם דגש על scalability

## למידה

### למדנו עקרונות קריטיים:

- Multi-agent systems - תיאום סוכנים מקבילים
- Prompt engineering - שיפור בן 18 איטרציות עם מדידות
- Concurrency patterns - ThreadPoolExecutor, thread-safety, queue management
- Cost optimization - ניתוח פיננסי מעמיק והבנת trade-offs
- Production architecture - clean code, testing, documentation, security

המסקנה: פרויקט ברמה אקדמית גבוהה (M.Sc.) עם ביצוע מקצועי, תיעוד יוצא דופן, ומחקר ממשי.

### הצהרת יושר אקדמי (Academic Integrity Declaration)

#### אני מצהיר/ה בזאת ש:

- ☒ ההערכה העצמית שלי היא כנה ואמיתית
- ☒ בדקתי את העבודה מול כל הקריטריונים לפני קביעת הציון
- ☒ אני מודע/ת שציון עצמי גבוה יוביל לבדיקה דקדקנית יותר
- ☒ אני מקבל/ת את העובדה שהציון הסופי עשוי להיות שונה מהציון העצמי
- ☒ העבודה היא פרי עבודתי/נו (של הקבוצה) ואני/ז אחראי/ם לכל תוכנה

חתימה: אריאל נפדנסקי ואבי פרדמן תאריך: 04.12.2025

