# **טכנולוגיות אינטרנט מתקדמות - 61776**

**משימת בית מס' 1**

**להגשה עד 20.4.25 בשעה 23:59**

**מהנדס מערכת: יהלי רפפורט**

|  |  |  |
| --- | --- | --- |
| **שם חבר הצוות** | **משימות שהוקצו** | **משימות שהושלמו** |
| יהלי רפפורט | |  | | --- | | כתיבת פרסונה ומפת אמפתיה |  |  | | --- | |  | |  |
| לידור בן חמו | |  | | --- | | פיתוח Layout ראשוני ב-Tailwind + HTML |  |  | | --- | |  | |  |
| תומר רול | |  | | --- | | כתיבת הדרישות (פונקציונליות ולא פונקציונליות) |  |  | | --- | |  | |  |
| טום ביטון | |  | | --- | | יצירת תרשים Use Case |  |  | | --- | |  | |  |
| ארד הרוש | |  | | --- | | מימוש לוגיקה של שינוי דרגת קושי ב-JavaScript |  |  | | --- | |  | |  |
|  |  |  |

קישור גיט של הצוות : <https://github.com/tomerroll/WebTeam>

3.

א. האתר שנבחר הוא **MathDuo**  – מערכת אינטראקטיבית ליצירת בעיות מתמטיקה מותאמות אישית. האתר מיועד לתלמידים בבתי ספר יסודיים וחטיבות ביניים, ומטרתו לאפשר תרגול מתמטי מותאם לרמת התלמיד, תוך מתן משוב מיידי ופתרונות מדורגים שלב-אחר-שלב. המערכת מתאימה את רמת הקושי של השאלות לפי הצלחת התלמיד בזמן אמת, במטרה לשפר הבנה ותחושת מסוגלות מתמטית.

**ב. פרסונה + מפת אמפתיה**

**👤 פרסונה: שירה, תלמידת כיתה ז'**

* **שם:** שירה לוי
* **גיל:** 12
* **רקע:** תלמידה חכמה אך מתקשה במתמטיקה. מרגישה לחץ במבחנים ואינה אוהבת לתרגל מהספר. אוהבת טכנולוגיה ומשחקי אונליין.
* **מוטיבציה:** רוצה להשתפר במתמטיקה כדי להרגיש בטוחה יותר בכיתה ולהעלות את הציון.
* **אתגרים:** מאבדת ביטחון עצמי כשהיא לא מצליחה לפתור תרגילים לבד, חוששת לטעות.
* **מטרה:** למצוא דרך לתרגל מתמטיקה שתהיה מעניינת, קלה להבנה, ותיתן לה תחושת הצלחה.

### 🧠 **מפת אמפתיה (Empathy Map)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| FEELS:   |  | | --- | | "אני לא מספיק טובה במתמטיקה" |  |  | | --- | |  | | SAYS   |  | | --- | | "אני לא מבינה איך עושים את זה" |  |  | | --- | |  | | "אני שונאת מבחנים במתמטיקה" | |  |  | | --- | |  | |
| THINKS:  "אם היו מסבירים לי שלב שלב אולי הייתי מבינה" | DOES:   |  | | --- | | מנסה לפתור, אבל מתייאשת מהר |  |  | | --- | |  |   מדלגת על תרגילים קשים |

ג. Divergent Thinking

1. בעיות מתמטיות שמסתגלות אוטומטית לפי הצלחות/כישלונות קודמים
2. פתרון מודרך – הצגה של כל שלב בפתרון בלחיצה
3. מצב תרגול לפי נושאים (שברים, משוואות, חזקות וכו')
4. אפשרות לבקש רמז במקום פתרון
5. תצוגת גרף/תרשים של ההתקדמות לאורך זמן
6. מצב "תרגול חוזר על טעויות"
7. תגמולים/נקודות/כוכבים להצלחה בתרגילים
8. מצב תחרות בין חברים
9. שילוב קול (Text-to-Speech) להסברים
10. שמירת פתרונות קודמים לצפייה חוזרת
11. מצב "משחק מתמטי" עם טיימר וניקוד
12. אפשרות לשלוח שאלה ישירות למורה
13. מערכת המלצות לשאלות על בסיס טעויות נפוצות
14. עיצוב מותאם לילדים – צבעים, דמויות מלוות
15. Dark Mode
16. תמיכה במסכים קטנים (סמארטפון)
17. חיבור לחשבון מורה לצפייה בהתקדמות התלמיד
18. אפשרות להוריד את הפתרון כקובץ PDF
19. קלט קולי – להגיד את התשובה ולא להקליד
20. שילוב וידאו קצר בכל נושא – חיזוק ההבנה

**ד. Convergent Thinking**

1. **בעיות מתמטיקה אדפטיביות** – התאמה לרמת התלמיד בזמן אמת
2. **הצגת פתרון שלב-אחר-שלב** – אפשרות לחשוף שלב בכל פעם
3. **רמזים לפני פתרון** – עידוד חשיבה עצמאית
4. **שמירת טעויות לתרגול חוזר** – חיזוק חולשות באופן ממוקד
5. **סטטיסטיקות וגרף התקדמות אישי** – חיווי ויזואלי להצלחה
6. **מצב תרגול לפי נושא** – גמישות למשתמש לבחור תחום
7. **עיצוב נגיש, צבעוני וידידותי לילדים**
8. **Fake data עבור גרסת הדמו** – לצורך הדגמה גם ללא בסיס נתונים

**ה. דרישות פונקציונליות (Functional Requirements)**

### 1.Adaptive question generation – the system allows solving math problems that adapt to the student’s level based on previous successes and failures.

2. Step-by-step solution display – the system allows viewing a step-by-step solution that can be revealed gradually.

3. Hint system – the system will allow the student to choose between receiving a hint or a full solution.

4. Saving success and error data – the system allows saving information about questions the student answered correctly or incorrectly.

5. Personal progress graph display – the system allows viewing a graph that reflects the student’s progress over time.

### (Non-Functional Requirements) דרישות לא פונקציונליות

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Description | |  |  |  | | --- | --- | --- | | |  | | --- | | **Requirement Type** |  |  | | --- | |  | |  |  | | --- | |  | | |  |  |  | | --- | --- | --- | | |  | | --- | | **Requirement** |  |  | | --- | |  | |  |  | | --- | |  | |
| |  |  |  | | --- | --- | --- | | |  | | --- | | Loading a new question will take less than one econd to prevent user frustration. |  |  | | --- | |  | |  |  | | --- | |  | | |  |  |  | | --- | --- | --- | | |  | | --- | | Performance |  |  | | --- | |  | |  |  | | --- | |  | | |  |  |  | | --- | --- | --- | | |  | | --- | | Fast response time |  |  | | --- | |  | |  |  | | --- | |  | |
| |  |  |  | | --- | --- | --- | | |  | | --- | | The interface will be simple, colorful, and intuitive for middle school children. |  |  | | --- | |  | | | |  | | --- | | Usability |  |  | | --- | |  | | |  |  |  | | --- | --- | --- | | |  | | --- | | User-friendly interface for ages 10–14 |  |  | | --- | |  | |  |  | | --- | |  | |
| |  |  |  | | --- | --- | --- | | |  | | --- | | The system will be able to support at least 1,000 active users at the same time. |  |  | | --- | |  | |  |  | | --- | |  | | Scalability | |  |  |  | | --- | --- | --- | | |  | | --- | | Support for multiple users simultaneously |  |  | | --- | |  | |  |  | | --- | |  | |
| |  |  |  | | --- | --- | --- | | |  | | --- | | Students' personal data will be stored securely and will not be accessible without authorization. |  |  | | --- | |  | |  |  | | --- | |  | | |  | | --- | |  |  |  | | --- | |  |   Security | |  |  |  | | --- | --- | --- | | |  | | --- | | Student data security |  |  | | --- | |  | |  |  | | --- | |  | |
| |  |  | | --- | --- | | |  | | --- | | The code will be modular and structured to allow easy addition of future features. | |  |  | | --- | |  | | |  | | --- | | Maintainability |  |  | | --- | |  | | |  | | --- | | Maintainable and extensible code |  |  | | --- | |  | |

ו. USE CASE

