חוזר מנהל מה"ט 51-4-11 – נספח מס '1 )הצעה לפרויקט גמר(

תאריך:\_\_\_\_\_\_\_\_\_\_\_

לכבוד

יחידת הפרויקטים מה"ט

**הצעה לפרויקט גמר**

1. **פרטי הסטודנטים**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| תאריך סיום  הלימודים | טלפון נייד | כתובת | ת.ז .9 ספרות | שם הסטודנט |
| 2023 | 0542859151 | קריית מוצקין, גושן 29א | 321816092 | אלכס אגרונוב |
| 2023 | 0539540139 | נהריה, שדרות ק. אברהם 7 | 333988939 | אלסו בוגדנוב |

שם המכללה בית הספר הארצי להנדסאים סמל המכללה: 72201

מסלול ההכשרה: הנדסאים

מגמת לימוד: הנדסת תוכנה 46/5

מקום ביצוע הפרויקט: בית הספר הארצי להנדסאים בקרית טכניון

1. **פרטי המנחה האישי**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| מקום עבודה/תפקיד | תואר | טלפון נייד | כתובת | שם המנחה |
| בה"ס להנדסאים בקרית טכניון | BA | 054-4226391 | חיפה, רח' פרץ מרקיש 14 | יבגניה צ'רנומז |

\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ חתימת הסטודנט חתימת המנחה האישי חתימת הגורם המקצועי מטעם מה"ט

דרך מנחם בגין 86 תל אביב ת.ד .36049 מיקוד 67138 טלפון: 7347521-03 פקס: 7347644-03

1. שם הפרויקט

"מערכת לניהול ומעקב אחר שיעורי בית של תלמידים"

1. רקע
   1. **תיאור ורקע כללי**

**תרגול מקצוע חדש, או תרגול אחר למידת תלמיד במקצוע קיים, זה משהו מאתגר הן למורה והן לתלמיד, בעיקר בגלל סביבת הלימודים שמתקיימת בבתי הספר כיום, שבה למורים אין אופציה לבדוק בזמן אמת את כל משימות הבית של כלל התלמידים, מכיוון שכיתות לימוד גדולות וקשב התלמידים ברובו הולך למכשירים האלקטרוניים. אנחנו באים לתת פיתרון חכם ומתקדם, שמותאם לזמננו. פתרון זה מספק מענה למערכת החינוך כמו לתלמידים. אנו בונים מערכת שבה בעצם למורה יש אפשרות לעקוב אחר משימות הבית של התלמיד באופן שוטף, כמו גם ליצור עוד תכנים ומשימות לימודיות לתלמיד ולספק ביקורת ומשוב על משימות אלו באופו מקוון.**

* 1. **מטרות המערכת:**

לייצר פלטפורמה שתאפשר למורים לייצר משימות ולעקוב אחר התקדמות התלמיד, בצורה מקוונת ונוחה. ובנוסף, לתת לתלמידים מערכת מקוונת אינטראקטיבית שתספק חווית למידה שונה ומעניינת.

1. סקירת מצב קיים בשוק, אילו בעיות קימות:

**לא מצאנו דברים דומים** למה שאנחנו עושים, ישנן מערכות ניהול כלליות שהמורה ישכול להזין את הציונים של הסטודנטים ופרטים שלהם. באותן מערכות הסטודנטים נכנסים לפאנל שלהם ויכולים לראות את הציונים ומשימות שנתנו להם, אבל ביצוע המשימות הוא לא נעשה במערכת עצמה.

1. מה הפרויקט אמור לחדש או לשפר:

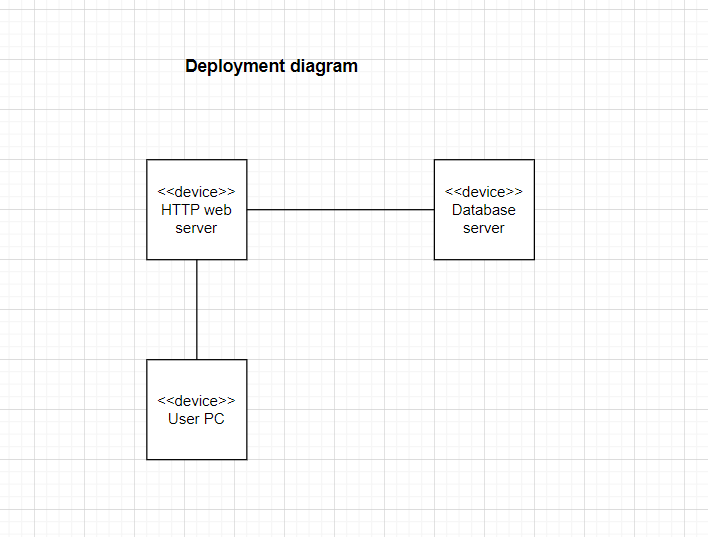
הפרוייקט בא לחדש ולהציע את ביצוע המשימות במערכת עצמה. כלומר, המורה יוצר את המשימות במערכת ומוסיף אותה לכיתה שהוא רוצה. מצד הסטודנט, הוא נכנס למערכת, רואה את המשימות שיש לו ועושה אותן במערכת עצמה. כמו כן, המשימות שניתן לעשות במערכת הן משימות יותר "מגרות", המשימות הן לא שאלה ותשובה סטנדרטיים אלא באות בצורת "משחק".

1. דרישות מערכת ופונקציונאליות
   1. **דרישות מערכת:** צריך שרת חיצוני וחיבור לאינטרנט, עמידות בזמן עומס.
   2. **דרישות פונקציונאליות:**

|  |  |  |
| --- | --- | --- |
| מספר דרישה | נוסח הדרישה | סוג דרישה |
| 1 | המערכת תכלול 3 סוגי משתמשים(אדמין, מורה, תלמיד) | דרישה תפעולית |
| 2 | המערכת תכלול כניסת משתמש(שם משתמש וסיסמה) | דרישה תפעולית |
| 3 | קיימת אפשרות יציאה מהמערכת | דרישה תפעולית |
| 4 | המערכת תכלול יצירת משתמש ובחירת תפקיד משתמש מורה/תלמיד(שם משתמש, סיסמה, מייל) | דרישה תפעולית |
| 5 | המערכת תכלול אפשרות שחזור סיסמה ושם משתמש(שכחתי סיסמה), שליחת מייל לאיפוס | דרישה תפעולית |
| 6 | החלפת סיסמה כל 3 חודש. | דרישת מאפייני איכות |
| 7 | כל משתמש יכול לעדכן את הפרטים האישיים בפרופיל כולל שינוי סיסמה | דרישה תפעולית |
| 8 | המערכת תבדוק ששם המשתמש והמייל לא קיימים כבר במערכת, במקרה וכן תהיה הודעה מתאימה | דרישת מידע |
| 9 | למורה יש אפשרות למחוק תלמיד מהכיתה במקרה הצורך | דרישה תפעולית |
| 10 | משתמש לא פעיל במשך חצי שנה ימחק. | דרישה תפעולית |
| 11 | המערכת תכלול בממשק המורה אפשרות חלוקה לפי קבוצות(כיתות). | דרישה תפעולית |
| 12 | למורה יש אפשרות ליצור כיתה | דרישה תפעולית |
| 13 | המערכת תעמוד בתוכן הולם. | דרישת מאפייני איכות |
| 14 | המורה יוצר את המשימות ע"י בחירת תבנית למשימה ומילוי תוכן. | דרישה תפעולית |
| 15 | תשובה נכונה מזכה את המשתמש בניקוד שנבקע על ידי המורה. | דרישה תפעולית |
| 16 | התלמיד יכול לבצע את המשימות שצירפו לו בכיתות שהוא נמצא בהן. | דרישה תפעולית |
| 17 | לכל משימה אצל התלמיד יש אייקון שמתווסף:  V – אם המשימה בוצעה בהצלחה  X- אחרת | דרישת מידע |
| 18 | המערכת תכלול רשימת פרסים/תעודות שנקבעו ע"י האדמין, ומקבלים אותם ע"י צבירת נקודות. | דרישת מידע |
| 19 | התלמיד יכול לראות את התעודות והפרסים שהוא קיבל. | דרישת מידע |
| 20 | המערכת תכלול אצל המורה סטטיסטיקה על כל תלמיד וכיתה. | דרישת מידע |
| 21 | המערכת תכלול אצל כל תלמיד את הסטטיסטיקה שלו. | דרישת מידע |
| 22 | אפשרות הורדה של הסטטיסטיקה(יחיד או קבוצה) בקובץ אקסל בטבלה | דרישת ממשק |
| 23 | המערכת תבצע גיבוי וחישוב של הסטטיסטיקות פעם ביום בשעה 12 בלילה | דרישת מאפייני איכות |
| 24 | לכל המשתמשים יש אפשרות צפייה בכיתות, תיקיות ומשימות וקיימות הרשאות:  אדמין, יכול לראות את כל הכיתות, תיקיות, והמשימות  מורה, יכול לראות רק את הכיתות, תיקיות, והמשימות שהוא יצר  תלמיד, יכול לראות רק את הכיתות, תיקיות, והמשימות שהוא מצורף אליהם. | דרישת מידע |
| 25 | אצל כל משתמש תהיה אפשרות כתיבת הודעות למשתמשים אחרים. | דרישה תפעולית |

1. בעיות צפויות במהלך הפיתוח ופתרונות (תפעוליות, טכנולוגיות, עומס ועוד):
   1. **תיאור הבעיות:**
2. דליפת פרטי משתמש.
3. התאמת האפליקציה למובייל.
4. חיבור למערכת מכל מקום.
5. אבטחה – עקב כניסה עם שם משתמש וסיסמה, תדרש הצפנה מצד הלקוח לשרת.
6. לימוד react, node js ברמה יותר גבוהה.
   1. **פתרונות אפשריים:**
7. דליפת פרטי משתמש – אכסון פרטי המשתמש באופן מוצפן.
8. התאמת האפליקציה למובייל – לימוד Bootstrap או עבודה יותר מורכבת עם JS ו CSS
9. חיבור למערכת מכל מקום – שימוש בשרת מארח שיתמוך ב nodejs ובאחסון של בסיס נתונים.
10. אבטחה – בעייה זאת ניתן לפתור באמצעות טכנולוגיית SSL
11. לימוד שפות ברמה גבוהה יותר – אין מנוס אלא ללמוד עצמאית.

1. פתרון טכנולוגי נבחר:
   1. **טופולוגית הפתרון-** המערכת מורכבת משרת, ממשק משתמש בצד הלקוח ,DB’s, טווח תקשורתי-אינטרנט, המערכת תיושם ברשת האינטרנט.



* 1. **טכנולוגיות בשימוש:**

**Visual studio code**- ליצרת front end ו back end.

* 1. **שפות הפיתוח:**

שפות פיתוח ליצירת back-end:

Node JS- שפה שנחשבת קלה לתפעול ושינויים, מסייעת בשיפור ביצועים, קהילת תומכים גדולה ופעילה.

MongoDB- הפרויקט מורכב מכמות רבה של נתונים ובעלי שונות גבוהה ו- MongoDB אידאלי לצורך טיפול בנתונים של מסמכים אלה. כל שדה רלוונטי המאוחסן במסמכי ה- MongoDB ניתן למיפוי קל ומיידי לאובייקט בתוכנה/אפליק', ולהיפך.

שפות פיתוח ליצירת front-end:

HTML- נשתמש בה בגלל שהמערכת היא מערכת אינטרנטית.

React- שפה נוחה וגמישה לשימוש, מאפשר פיתוח של אתרים סטטים, קוד ברור שמחולק באופן קל להבנה ולקריאה.

**תיאור הארכיטקטורה הנבחרת:**

המערכת שלנו דורשת חיבור לבסיס נתונים.בשביל זה הפרויקט שלנו יכלול תבנית Model-view-controller,על מנת שבקשות המתקבלות משכבת View יהיו מתורגמות בעזרת controller,ולאחרת מכן יהיו מועברות ל-model שיעשה את העבודה שלו מול בסיס נתונים.

* 1. **חלוקה לתכניות ומודולים:**

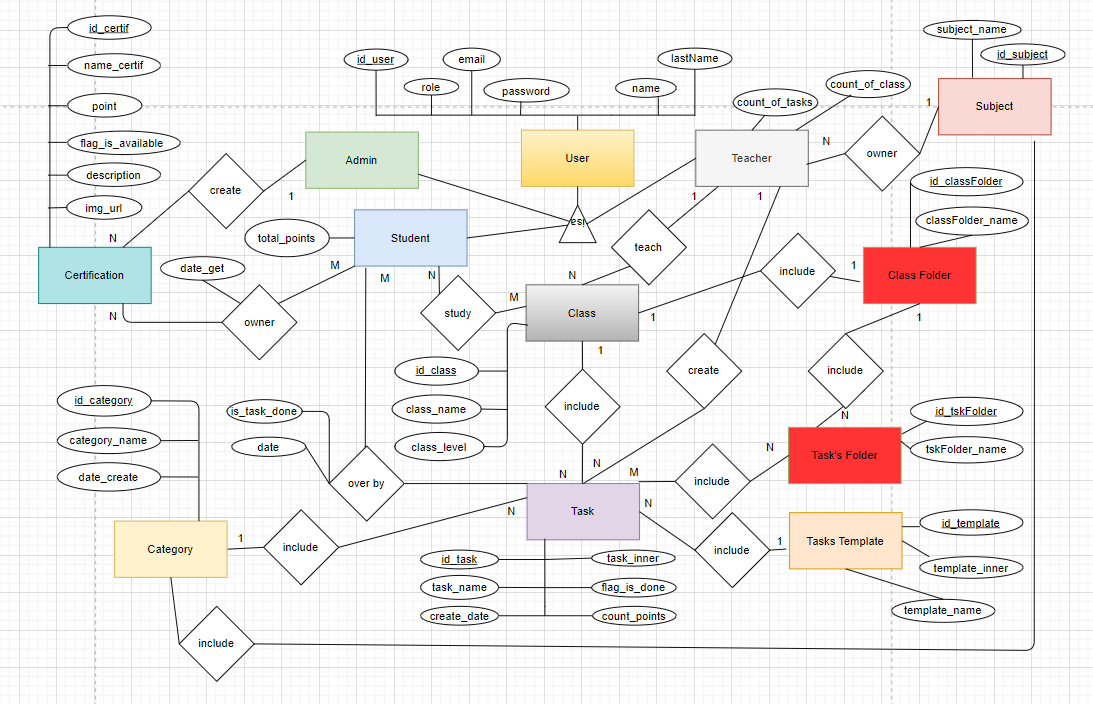
המערכת שתכלול תבנית MVC.שכבת ה-view תכיל בתוכה ממשק משתמש,שייתן אפשרות למשתמש לבצע פעולות גומלין כלשהן לפי רצונו.לאחר מכן פקודות אלה עוברות ל-controller שיכלול בתוכו מחלקות מיוחדות שתפקידן לעבד ולתרגם את המידע שהתקבל מהתצוגה,לשפה של בסיס נתונים.בסופו של דבר,הבקשה עוברת ל-model השכבה שעובדת ישירות מול בסיס נתונים. בנוסף,על המערכת שלנו ליצור סטטיסטיקות למורה ולתלמיד בפורמטים מסוימים.לצורך זה, עלינו לכתוב מחלקות שישמשו כ-adapter,ויוכלו להמיר את המידע שמתקבל מבסיס נתונים לפורמט הרצוי (כגון PDF,EXCEL וכו'.)

* 1. **סביבת השרת (מקומי, וירטואלי, ענן, שירות אירוח):**

במהלך הפיתוח של הפרוייקט נשתמש בשרת לוקאלי בלבד. כמו כן, לאחר סיום הפיתוח, נשתמש בשרת חיצוני.

* 1. ממשק המשתמש/לקוח - GUI – סקיצות, ברמה של שרטוט. כמובן, בהמשך זה יכול להשתנות, תכנון ראשוני.
  2. ממשקים למערכות אחרות/ API: אם יש צורך
  3. שימוש בחבילות תוכנה: -לפי צורך

1. שימוש במבני נתונים וארגון קבצים
   1. **נא פרט את מבני הנתונים:**



* 1. נא פרט את שיטת האיחסון (מאגר, קבצים ובאיזה טכנולוגיה) – תרגום ERD למודל טבלאי , תצוגה ברילציות.

1. **תרשימי מערכת מרכזיים** 
   1. **Use Case Diagram + Document**



| **שם** | Login |
| --- | --- |
| **תיאור** | התרחיש מתאר את תהליך ההתחברות למערכת |
| **שחקנים** | user |
| **סטטוס** | מחובר, לא מחובר |
| **תדירות** | כל פעם שהמשתמש מתחבר למערכת |
| **עקיבות לדרישות** | 2 |
| **תנאים מקדימים** | פתיחת חלון כניסה למערכת, הזנת שם משתמש וסיסמה |
| **תנאים מסיימים** | הלקוח נכנס למערכת לאחר לחיצה על כפתור ההתחברות. |
| **מזניק** |  |
| **תרחיש הצלחה ראשי** | 1. המשתמש מקליד שם משתמש וסיסמה במערכת 2. המערכת מאמתת את המשתמש דרך הפרטים שהזין 3. המשתמש יתחבר למערכת בהצלחה |
| **הסתעפויות** | המערכת תדפיס למשתמש שהשם משתמש או הסיסמה אינם נכונים |

|  |  |
| --- | --- |
| **שם** | to do task |
| **תיאור** | הסטודנט יכול לעשות את המשימות שקיימות אצלו. |
| **שחקנים** | סטודנט |
| **תדירות** | כל פעם שיש משימה חדשה במערכת |
| **עקיבות לדרישות** | 15,16,17,18,19 |
| **תנאים מקדימים** | לסטודנט יש משימה מוכנה לביצוע |
| **תנאים מסיימים** | הסטודנט מקבל ניקוד על ביצוע המשימה |
| **תרחיש הצלחה ראשי** | 1. הסטודנט נכנס לתקייה הרצויה 2. בוחר את המשימה הרצויה 2. מבצע את המשימה בהצלחה 3. מקבל ניקוד על הצלחת המשימה |
| **הסתעפויות** | אם לא הצליח את המשימה לא מקבל ניקוד |

|  |  |
| --- | --- |
| **שם** | get progress info |
| **תיאור** | הורדת סטטיסטיקה וקבלת מידע על התקדמות הסטודנט |
| **שחקנים** | מורה |
| **תדירות** | כל פעם שיש צורך למורה לבדוק סטטיסטיקה או התקדמות |
| **עקיבות לדרישות** | 20,21,22,23 |
| **תנאים מקדימים** | 1. קיימים סטודנטים אצל המורה במערכת |
| **תנאים מסיימים** | הופעת גרף סטטיסטיקה של משימות/קבלת מידע על הישגים של הסטודנט |
| **תרחיש הצלחה ראשי** | 1. בחירת כיתה מהרשימה 2. בחירה לקבלת סטטיסטיקה על סטודנט מסויים לפי תאריכים 3. הופעת גרף הסטטיסטיקה /הופעת מידע על הסטודנט |
| **הסתעפויות** | במקרה ויש תקלה תופיע הודעת שגיאה |

|  |  |
| --- | --- |
| **שם** | get progress info |
| **תיאור** | הורדת סטטיסטיקה וקבלת מידע על התקדמות הסטודנט |
| **שחקנים** | מורה |
| **תדירות** | כל פעם שיש צורך למורה לבדוק סטטיסטיקה או התקדמות |
| **עקיבות לדרישות** | 20,21,22,23 |
| **תנאים מקדימים** | 1. קיימים סטודנטים אצל המורה במערכת |
| **תנאים מסיימים** | הופעת גרף סטטיסטיקה של משימות/קבלת מידע על הישגים של הסטודנט |
| **תרחיש הצלחה ראשי** | 1. בחירת כיתה מהרשימה 2. בחירה לקבלת סטטיסטיקה על סטודנט מסויים לפי תאריכים 3. הופעת גרף הסטטיסטיקה /הופעת מידע על הסטודנט |
| **הסתעפויות** | במקרה ויש תקלה תופיע הודעת שגיאה |

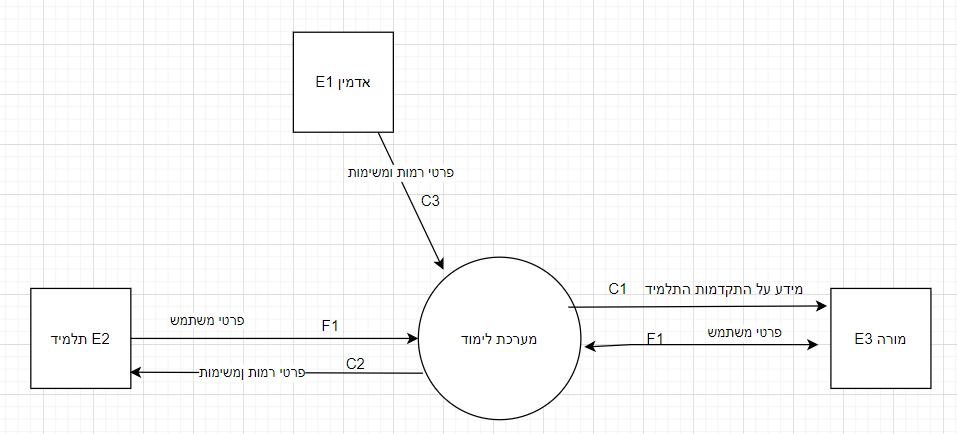
|  |  |
| --- | --- |
| **שם** | students management |
| **תיאור** | יצירה או מחיקת משתמשים, מיון משתמשים, בחירת משימות לכיתה והצגת/הורדת סטטיסטיקה על מעבר המשימה |
| **שחקנים** | מורה |
| **תדירות** | כל פעם שיש צורך באחד מהדברים |
| **עקיבות לדרישות** | 11,12,24,25 |
| **תנאים מקדימים** | קיימים משתמשים ברשימה של מורה, בחירת תקופת תאריכים לקבלת סטטיסטיקה |
| **תנאים מסיימים** | המשתמש נוסף/נמחק, רשימה ממויינת של הסטודנטים, המשימה נוספה בהצלחה, הופעת סטטיסטיקה כללית על הכיתה |
| **תרחיש הצלחה ראשי** | 1. יצירה של משתמשים: a. בוחר/יוצר תקיה של הכיתה b. מזין את הפרטים של תלמיד בודד או רשימה של כמה תלמידים c. המערכת מייצרת רנדומלי את הסיסמאות 2. מחיקה של משתמש: a. בוחר תקיה של הכיתה b. בוחר את התלמיד שצריך למחוק c. התלמיד נמחק מהרשימה 3. מיון משתמשים: a. בחירת התקיה הרצויה b. לחיצה על מיון משתמשים c. המערכת ממיינת את המשתמשים 4. בחירת משימות לכיתה: a. כניסה לתקיית הכיתה הרצויה b. בחירה מרשימת המשימות את המשימות שצריך להוסיף והגדרת תאריך הגשה c. משימות מתווספות לכיתה 5. הורדת/הצגת סטטיסטיקה: a. כניסה לתקיית הכיתה הרצויה b. הורדת הסטטיסטיקה על הכיתה/מעבר משימה c. הסטטיסטיקה מוצגת/יורדת בהצלחה |
| **הסתעפויות** | במידה והתהליך לא מצליח תופיע הודעת שגיאה מתאימה |

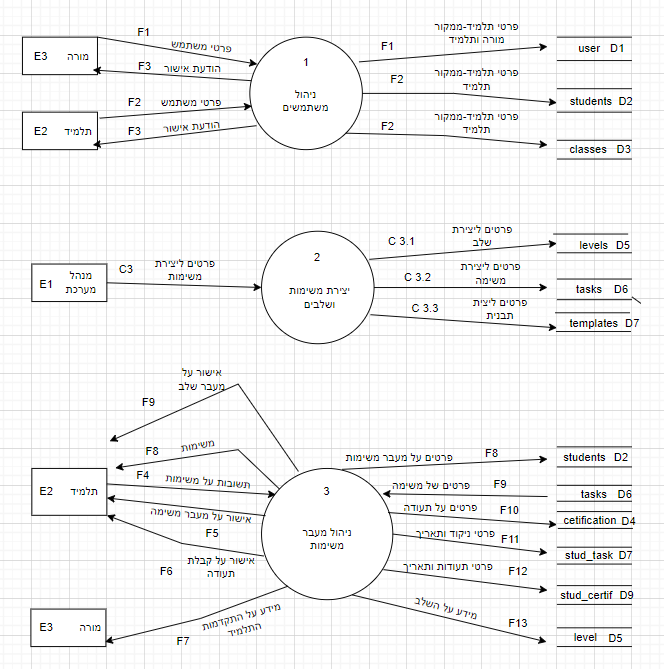
|  |  |
| --- | --- |
| **שם** | create remove tasks |
| **תיאור** | בחירת קטגוריה למשימות, בחירת תבנית למשימה וניקוד שמקבלים על המשימה או מחיקת משימה |
| **שחקנים** | מורה |
| **תדירות** | כל פעם שמייצרים משימה חדשה/מוחקים משימה |
| **עקיבות לדרישות** | 14 |
| **תנאים מקדימים** | נבחרה תבנית רצויה, נבחרה קטגוריה |
| **תנאים מסיימים** | המשימה נוצרה/נמחקה |
| **תרחיש הצלחה ראשי** | 1. נבחרה קטגוריה רצויה למשימה 2, נבחרה תבנית רצויה למשימה 3. הזנת נתונים למשימה 4. המשימה נוצרה בהצלחה/נמחקה בהצלחה |
| **הסתעפויות** | במידה והתהליך לא מצליח תופיע הודעת שגיאה מתאימה ולא יתווסף/ימחק כלום |

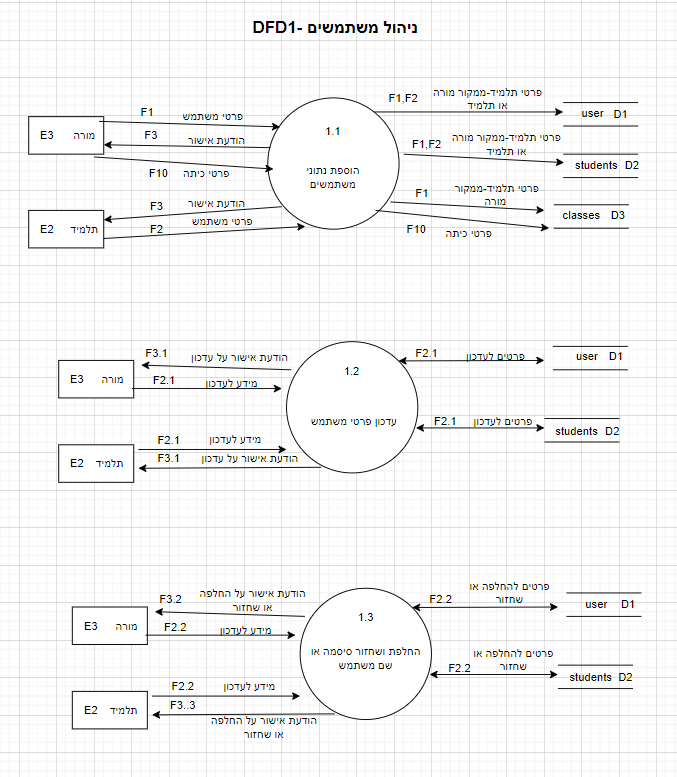
|  |  |
| --- | --- |
| **שם** | manage app |
| **תיאור** | יצירת תבניות, תעודות וניהול סיסמאות של משתמשים |
| **שחקנים** | אדמין |
| **תדירות** | כל פעם שיוצרים תבנית, תעודה או צריך לשחזר את הסיסמא של המשתמש |
| **עקיבות לדרישות** | 6,10,13 |
| **תנאים מקדימים** | גישה לבסיס נתונים, הזנת פרטים שרוצים לשנות או להוסיף |
| **תנאים מסיימים** | התבניות/תעודות נוצרות בהצלחה |
| **תרחיש הצלחה ראשי** | 1. כניסה לבסיס נתונים 2. הזנת פרטים רצויים(שינוי או הוספה) 3. שמירת פרטים |
| **הסתעפויות** | במידה והתהליך לא מצליח תופיע הודעת שגיאה מתאימה ולא יתווסף/ימחק כלום |

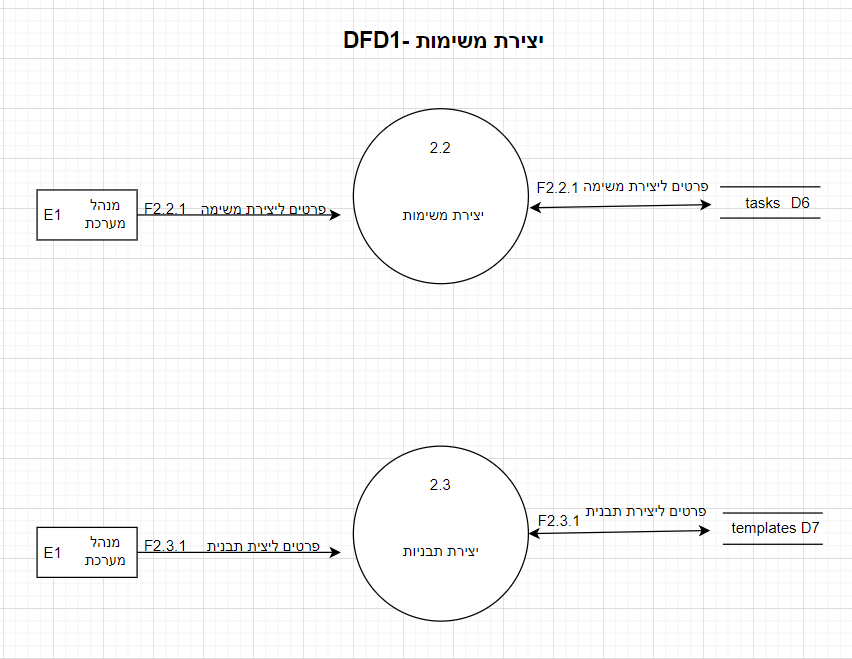
|  |  |
| --- | --- |
| **שם** | Registration |
| **תיאור** | הרשמה למערכת על ידי הזנת פרטים |
| **שחקנים** | מורה |
| **תדירות** | כל פעם שהמורה רוצה להירשם |
| **עקיבות לדרישות** | 1,4,8 |
| **תנאים מקדימים** | בחירת הרשמה למערכת |
| **תנאים מסיימים** | המשתמש נכנס למערכת בהצלחה ורואה את האיזור האישי |
| **תרחיש הצלחה ראשי** | 1. בחירת הרשמה או כניסה למערכת 2. הזנת פרטים 3. המערכת מאמתת את הפרטים 4. המורה נכנס או נרשם בהצלחה |
| **הסתעפויות** | אם הפעולה לא הצליחה תופיע הודעת שגיאה מתאימה |
| **שם** | logout |
| **תיאור** | יצאת משתמש מהמערכת |
| **שחקנים** | מורה, תלמיד, אדמין |
| **תדירות** | כל פעם שמישהו רוצה לצאת מהמערכת |
| **עקיבות לדרישות** | 3 |
| **תנאים מקדימים** | משתמש צריך להיות מחובר למערכת |
| **תנאים מסיימים** | המשתמש יצאה מהמערכת |
| **תרחיש הצלחה ראשי** | 1. לחיצה על כפתור יציאה מהמערכת 2. המשתמש יצא בהצלחה מהמערכת |
| **הסתעפויות** | אם התהליך לא הצליח תופיע הודעת שגיאה |

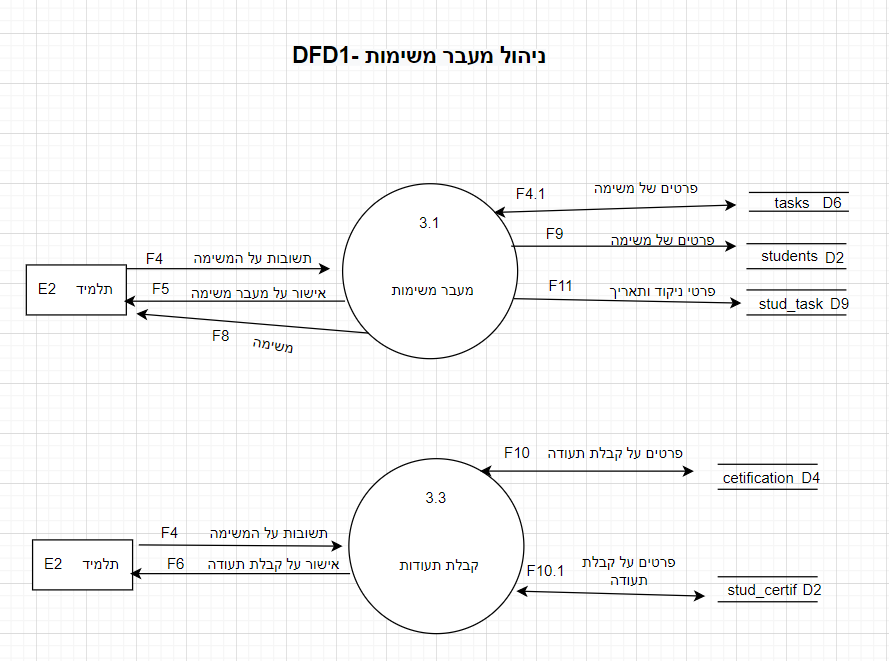
* 1. Sequence diagram
  2. **DFD0+DFD1:**











* 1. תרשים מחלקות

1. **תיאור המרכיב האלגוריתמי – חישובי** 
   1. איזה בעיה בא לפתור, איך יפתור? –אלגוריתם לניהול בדיקות, אם יש מקום לדייר, אלגוריתמית לפתור בעיות אלו, מומלץ לצרף Activity Diagram, אבל אפשר גם להשתמש באלגוריתם מילולי.

1. **תיאור/התייחסות לנושאי אבטחת מידע**

נא לציין אזורים הדורשים אבטחה, כגון: שרת, בקרת גישה לאתר, חשבונות משתמשים, מאגרי מידע וכיצד ניתן מענה.

נא ציין מס' מקרים ותגובות להם ניתן מענה אבטחתי.-הרשאות וסיסמאות , מה מנהל עושה, משתמש רשום מה מבצע, משתמש לא רשום

1. **משאבים הנדרשים לפרויקט:** 
   1. מספר שעות המוקדש לפרויקט, חלוקת עבודה בין חברי הצוות – לזוג 600, 300 לכל סטודנט
   2. ציוד נדרש –חיבור לאינטרנט, ציוד נוסף ברמת מחשב, מצלמות.
   3. תוכנות נדרשות-העתקה סעיף 7.2
   4. ידע חדש שנדרש ללמוד לצורך ביצוע הפרויקט - ברור
   5. ספרות ומקורות מידע –לפחות 3 מקורות , כמו stackoverflow, אבל קישור מלא, לא רק שם אתר

1. **תכנית עבודה ושלבים למימוש הפרויקט- ניתוח פרויקט תאריך 19.2.2022**, 14 אפריל תכנון ממשק משתמש איך מסכים יראו, כתיבת קוד 15 יולי , בדיקות תוכנה 15 אוגוסט , כתיבת ספר פרויקט 23 אוגוסט ,1.9 הגשת ספר פרויקט.

1. **תכנון הבדיקות שיבוצעו** 
   1. נא פרט בטבלה ,בדיקות תהליכיות ברמת משתמש בהן נדרשת המערכת לעמוד

(full Flow)—מסמך דרישות שמוכן , ולכל דרישה איך חושבים לפתור את זה, אחרי הוספה לבסיס נתונים בודקים הם באמת נוסף לבסיס נתונים.

חותמים 2 חתימות על אותה שורה, דף חתימות זה עמוד אחד! לבדוק לפני הדפסה!

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ חתימת הסטודנט חתימת המנחה האישי

1. **הערות ראש המגמה במכללה**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **אישור ראש המגמה**

שם: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ חתימה: \_\_\_\_\_\_\_\_\_\_\_\_ תאריך:\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **הערות הגורם המקצועי מטעם מה"ט**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **אישור הגורם המקצועי מטעם מה"ט**

שם: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ חתימה: \_\_\_\_\_\_\_\_\_\_\_\_ תאריך:\_ \_\_\_\_\_\_\_\_\_\_\_\_