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 |

אלסו בוגדנוב, אלכס אגרונוב

\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 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:

**מערכות דומות הם classroom ומודל. הקלאסרום לדוגמה היא מערכת פשוטה ונוחה לשימוש אין הרבה פונקציונליות שיכולה לסבך. התפעול מובן וברור. במערכות אלו הסוג תרגילים שניתן לתת די מוגבל ואין אופציה לעשות דברים מושכים ומאתגרים לגילאים נמוכים(ילדים).**

**כמו כן קיימת מערכת יותר מתאימה לילדים** [**https://learningapps.org/**](https://learningapps.org/) **. במערכת זו יש יותר אופציות ויותר סגנונות של תרגילים שאפשר לתת כמו למשל תשחצים או התאמת מילה לתמונה וכו'.**

1. מה הפרויקט אמור לחדש או לשפר:

הפרוייקט בא לחדש את מגוון האופציות והסגנונות למשימה. כמו למשל כאן [**https://learningapps.org/**](https://learningapps.org/) שיש משימות של תשחצים או מיון תמונות והתאמות למיניהם. גם אצלינו יש מגוון של אפשרויות דומות. מצד המורה, יש אפשרות הוספת משימות שקלה לביצוע ולא מסובכת, בחירת סוג התרגיל, ושיוך שלו לכיתה המתאימה. כמו כן, מעקב קל ונוח של כל תלמיד או כיתה והשוואה בין כיתות שונות(סטטיסטיקות).

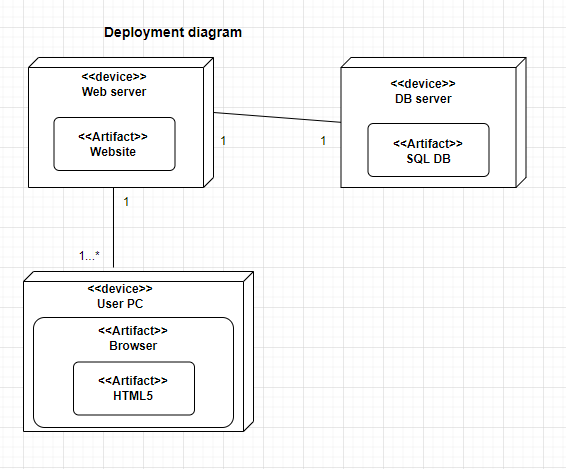
מצד הסטודנט, מערכת אינטרקטיבית ונוחה לשימוש, מעקב נוח אחרי התרגילים שקיימים אצלו.

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

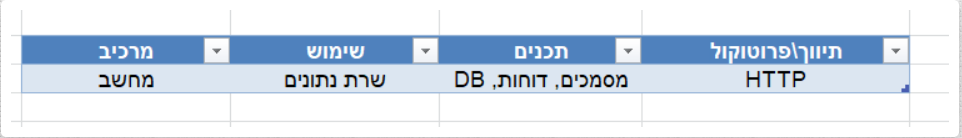
|  |  |  |
| --- | --- | --- |
| מספר דרישה | נוסח הדרישה | סוג דרישה |
| 1 | המערכת תכלול 3 סוגי משתמשים(אדמין, מורה, תלמיד) | דרישה תפעולית |
| 2 | המערכת תכלול כניסת משתמש(שם משתמש וסיסמה) | דרישה תפעולית |
| 3 | קיימת אפשרות יציאה מהמערכת | דרישה תפעולית |
| 4 | המערכת תכלול יצירת משתמש ובחירת תפקיד 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 |

1. בעיות צפויות במהלך הפיתוח ופתרונות (תפעוליות, טכנולוגיות, עומס ועוד):
   1. **תיאור הבעיות:**
2. דליפת פרטי משתמש.
3. התאמת האפליקציה למובייל.(אם תהיה גרסת מובייל)
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.

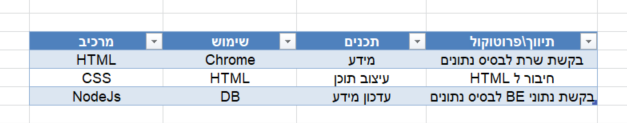
1. פתרון טכנולוגי נבחר:
   1. **טופולוגית הפתרון-** המערכת מורכבת משרת, ממשק משתמש בצד הלקוח ,DB’s, טווח תקשורתי-אינטרנט, המערכת תיושם ברשת האינטרנט.



פרטי חומרה



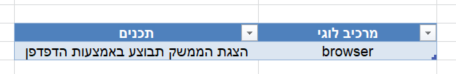
פרטי תוכנה



ממשקים פיזים



ממשקים לוגים



* 1. **טכנולוגיות בשימוש:**

Visual studio code- ליצרת front end ו back end.

MySQL – מערכת לבסיס נתונים

* 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(לא בטוח שמשתמשים בה זה עדיין בתהליך החלטה)- שפה נוחה וגמישה לשימוש, מאפשר פיתוח של אתרים סטטים, קוד ברור שמחולק באופן קל להבנה ולקריאה.

במקרה ובסוף הוחלט שלא משתמשים בריאקט, אז בונים את המערכת עם JS וCSS.

* 1. **תיאור הארכיטקטורה הנבחרת:**

המערכת דורשת חיבור לבסיס הנתונים, לכן נשתמש ב MVVM בעל מבנה 3 השכבות.

הרעיון הוא הפרדה בין View ל Model כשההתקשרות ביניהם היא באמצעות פקודות להעברת מסרים, ו Data biling בין VIEW ל ViewModel כדי להציג נתונים.

* 1. **חלוקה לתכניות ומודולים:**

לפי ארכיטקטורת MVVM:

Model – מודל מתאר את אובייקט ה DATA של התוכנית.

View – זה החלק האחראי על הנראות(גרפיקה) וממשק המשתמש.

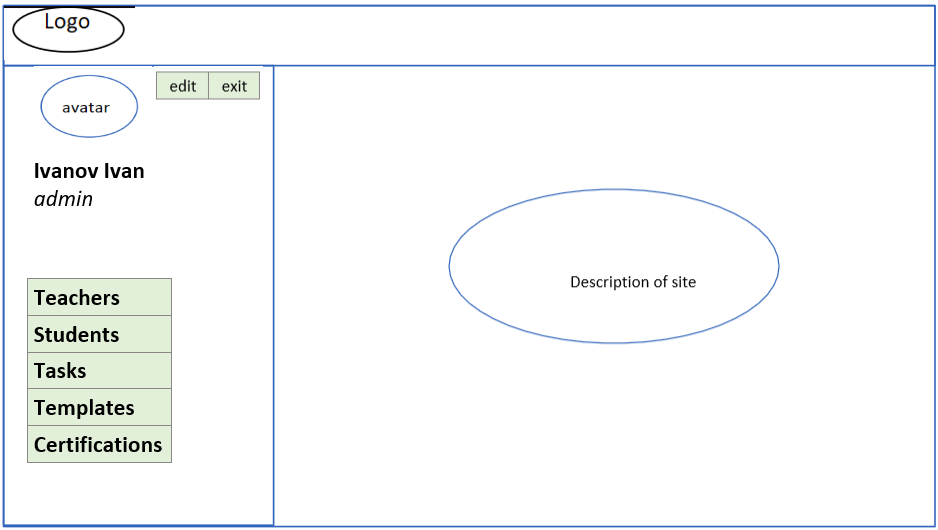
ViewModel – החלק האחראי על התקשרות בין החלקים הקודמים ומחזיק את רכיבי ה UI.

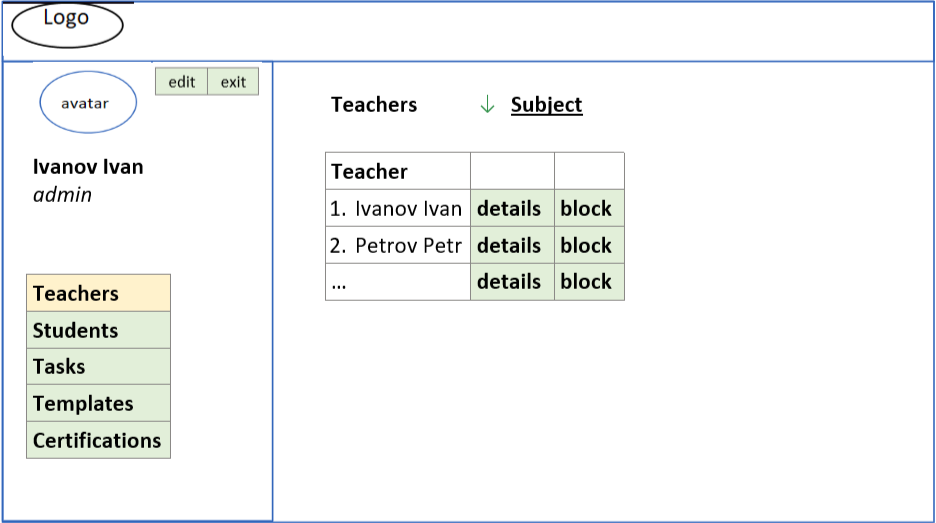
* 1. **סביבת השרת (מקומי, וירטואלי, ענן, שירות אירוח):**

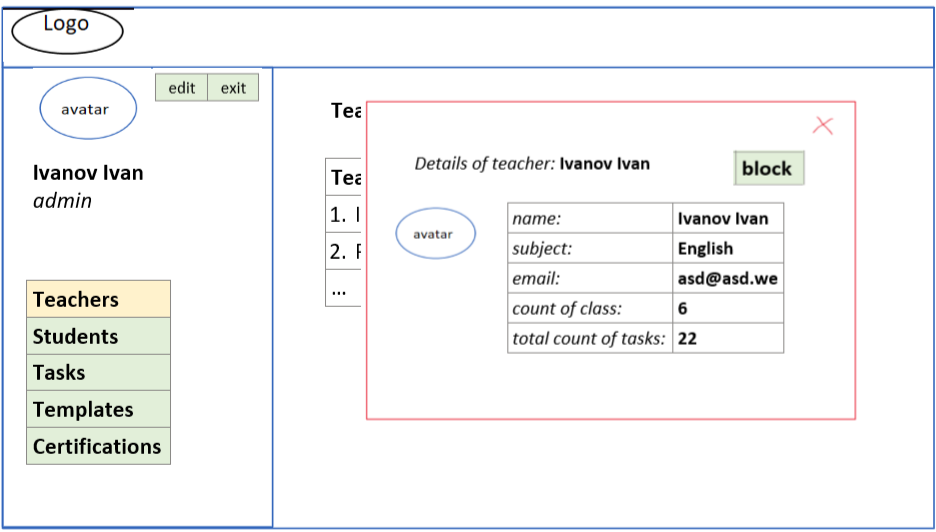
במהלך הפיתוח של הפרוייקט נשתמש בשרת לוקאלי בלבד. כמו כן, לאחר סיום הפיתוח, נשתמש בשרת חיצוני.

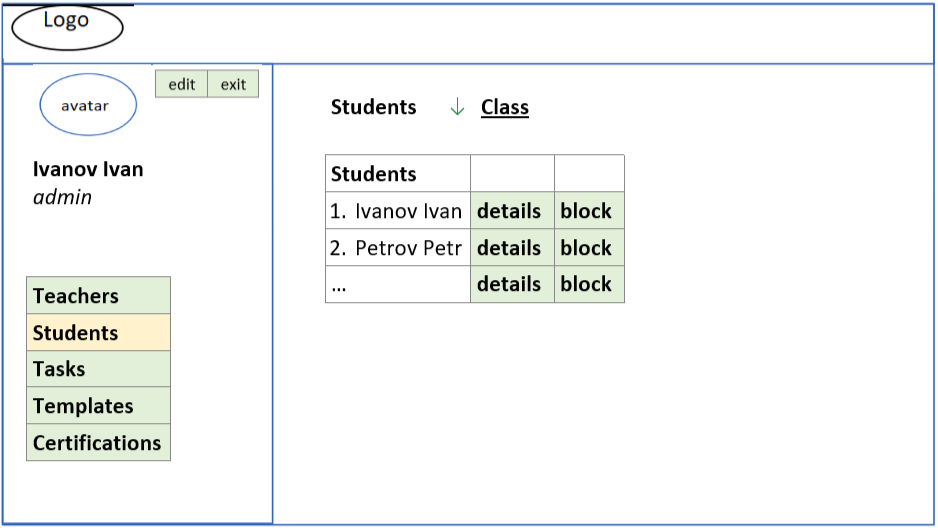
* 1. **ממשק המשתמש/לקוח – GUI**:

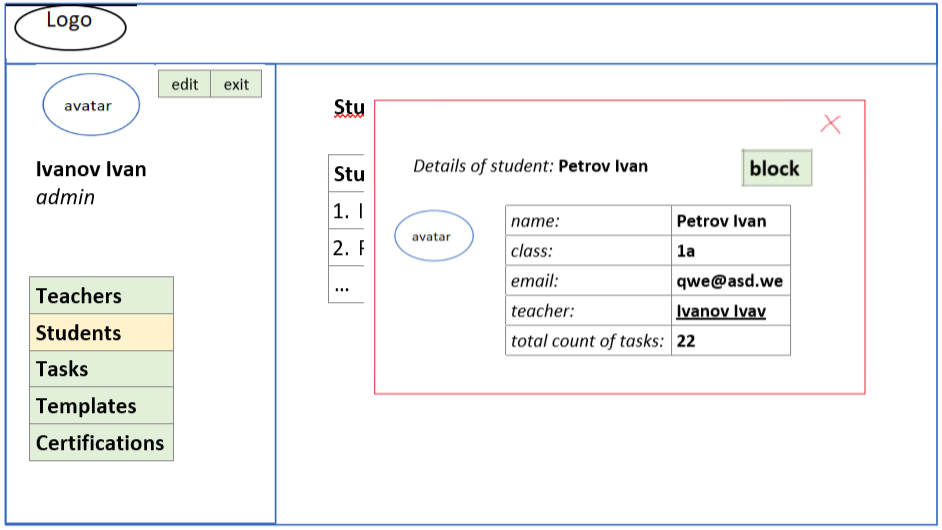
ממשק של האדמין

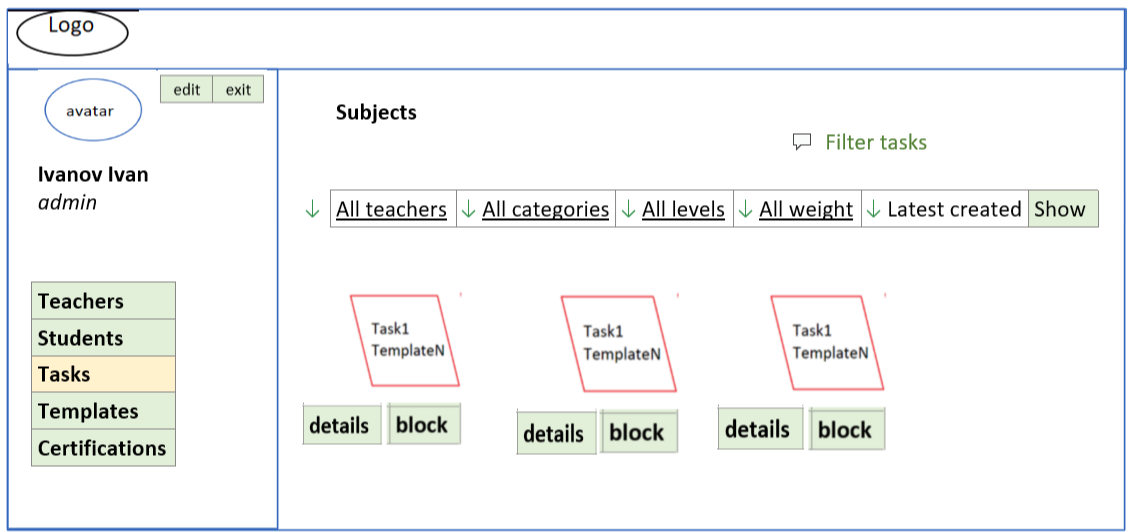


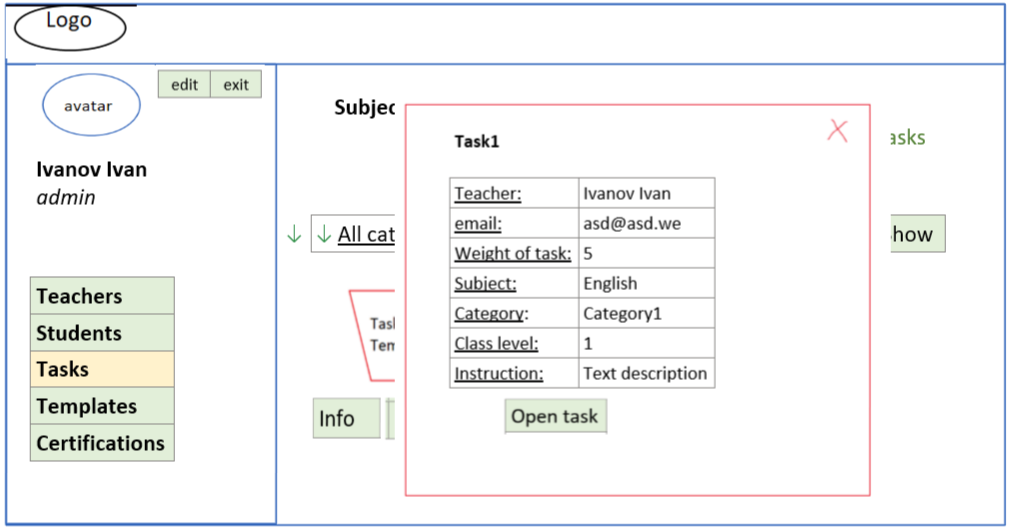


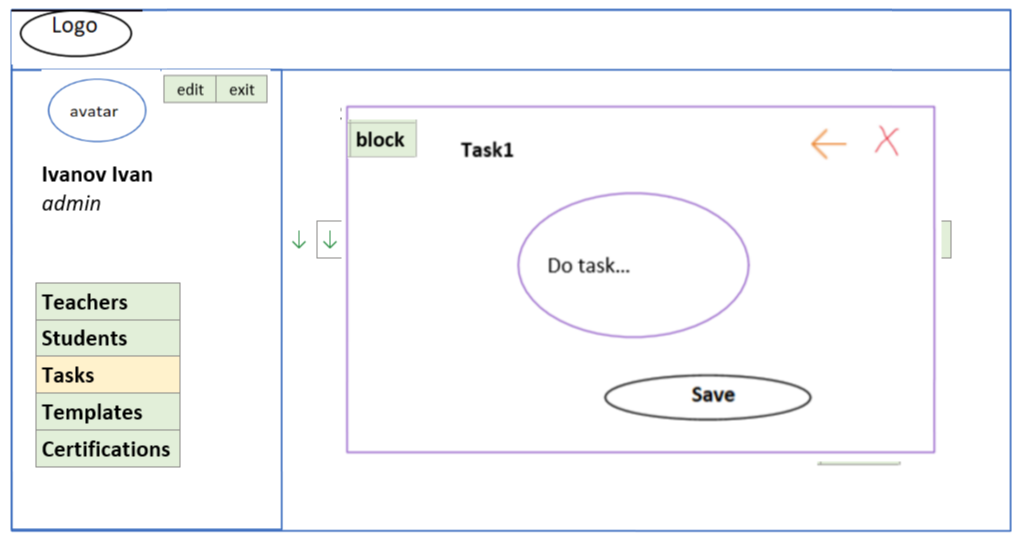


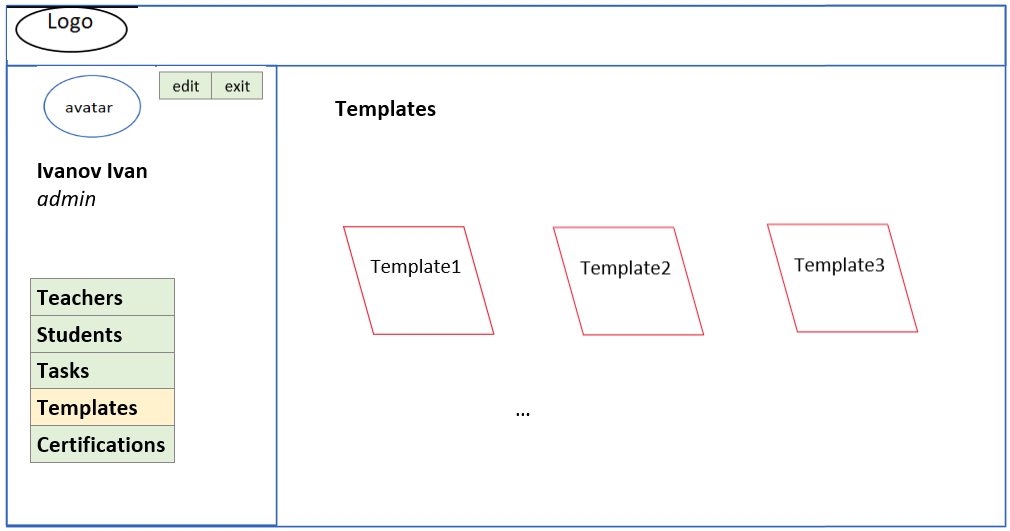


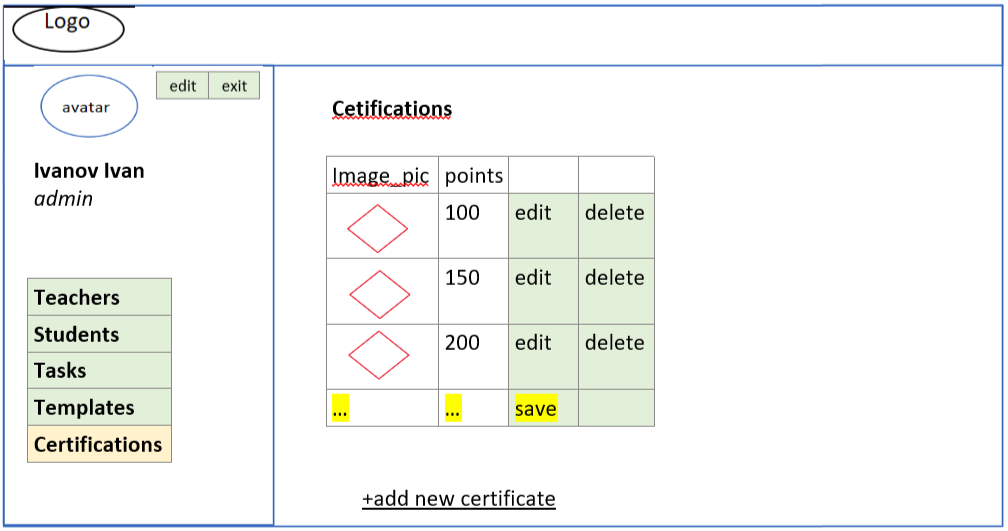




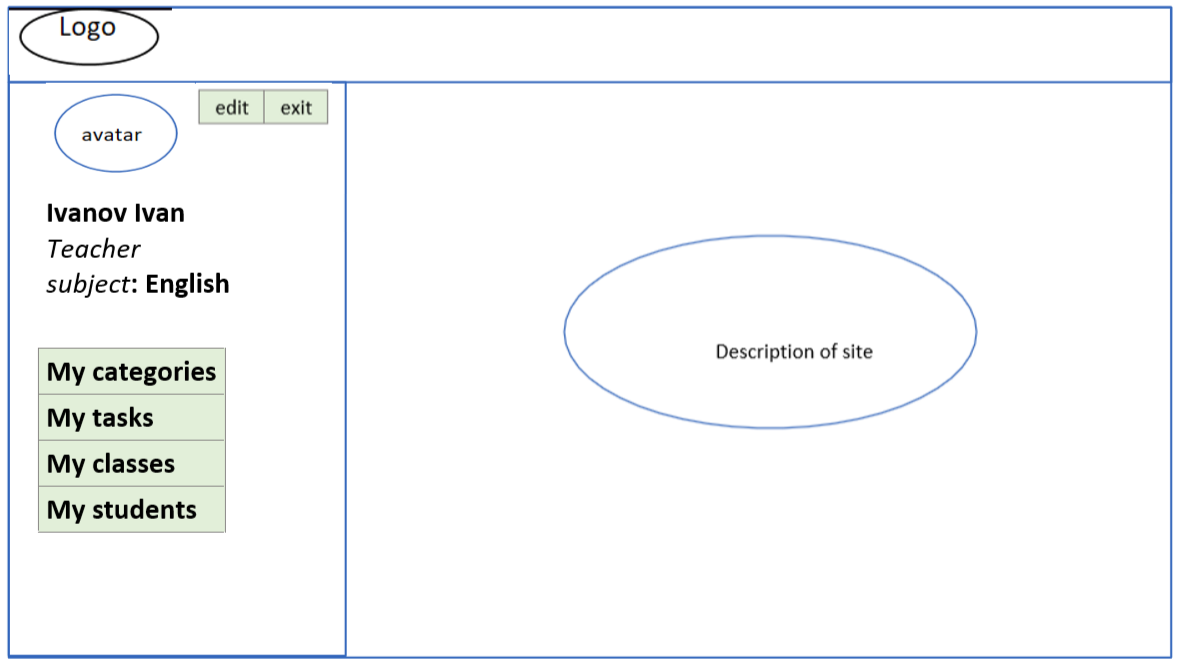


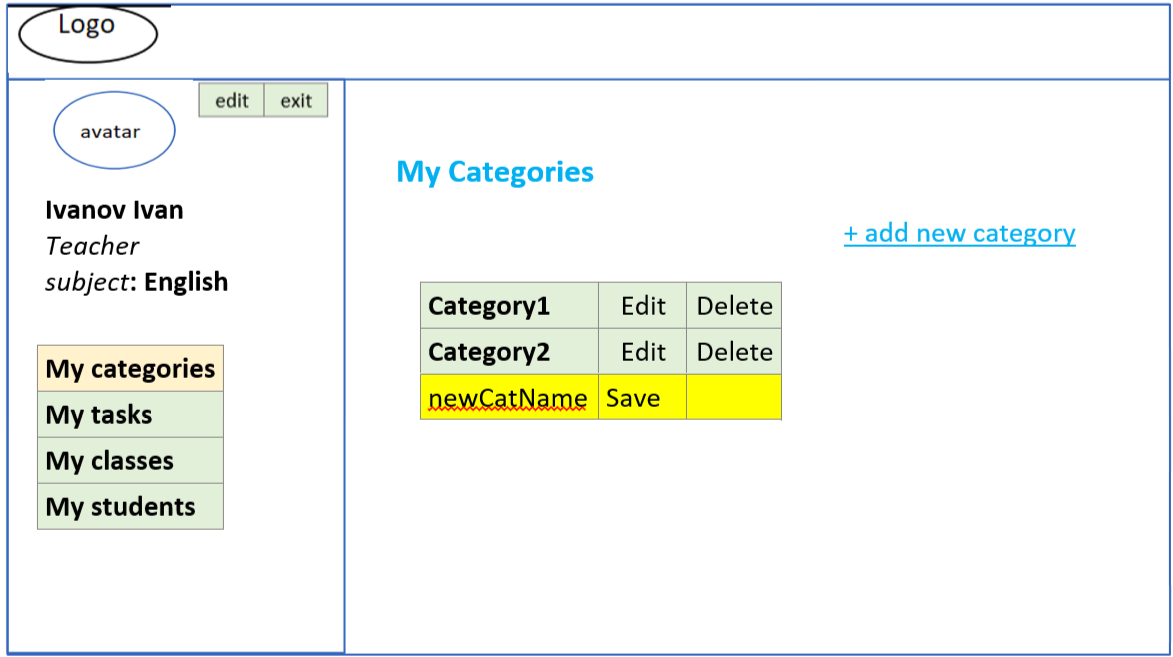


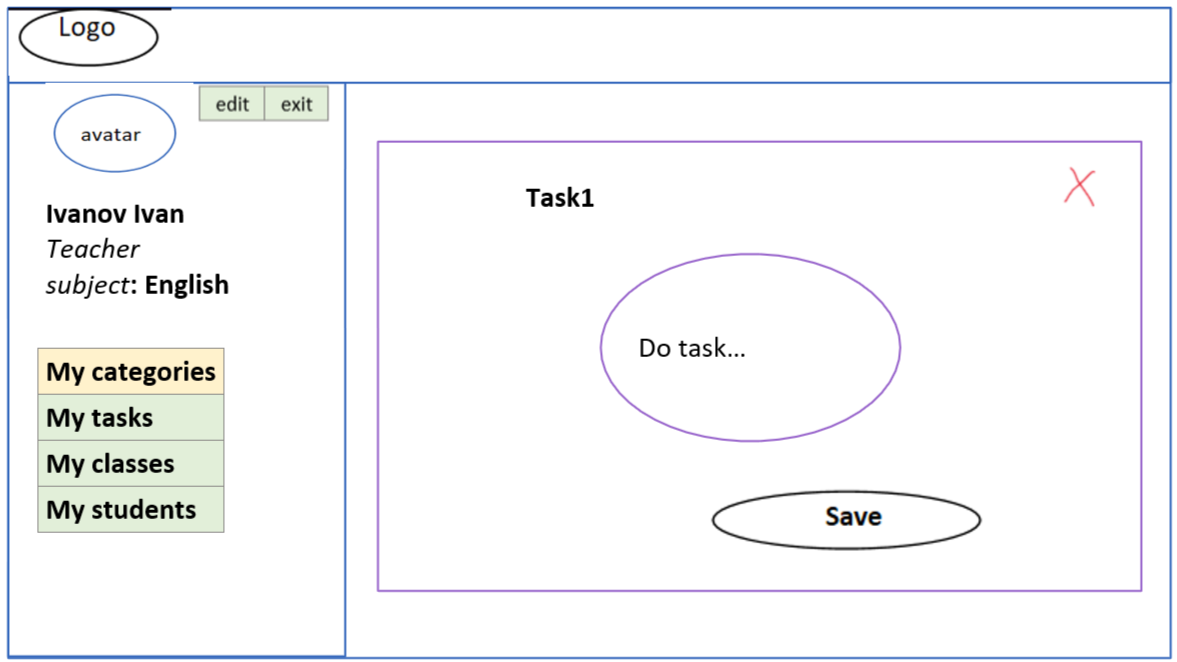


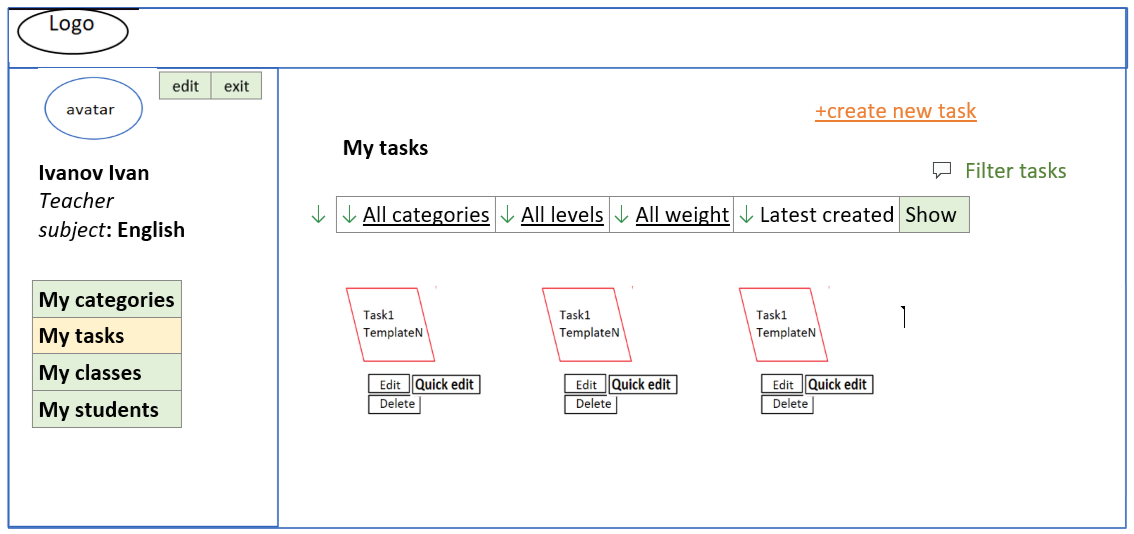


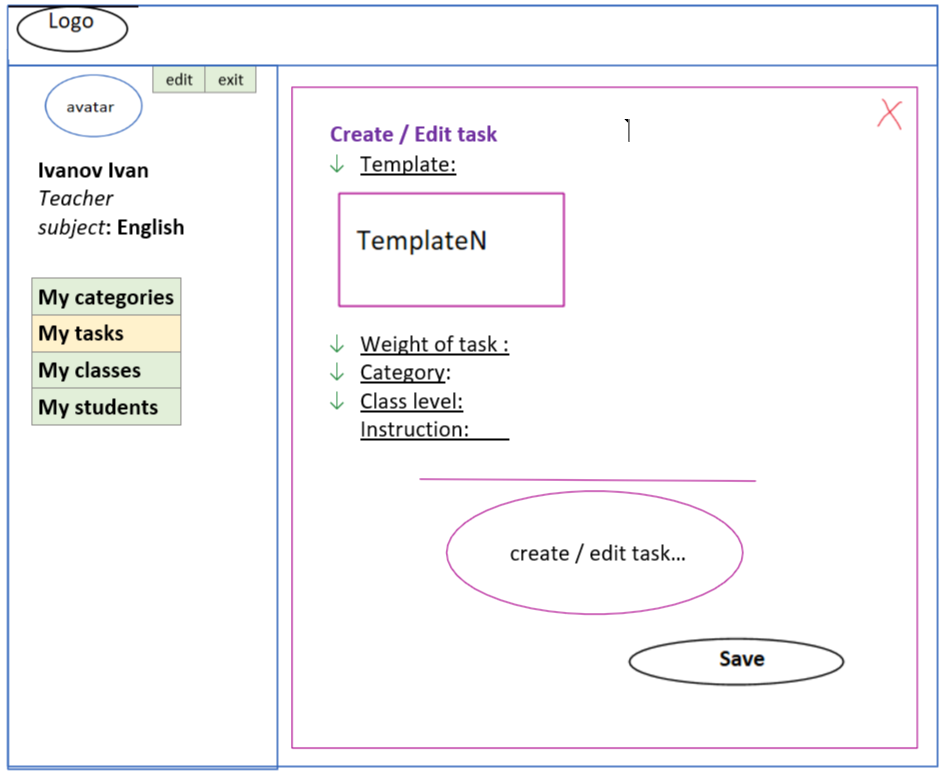
ממשק של המורה

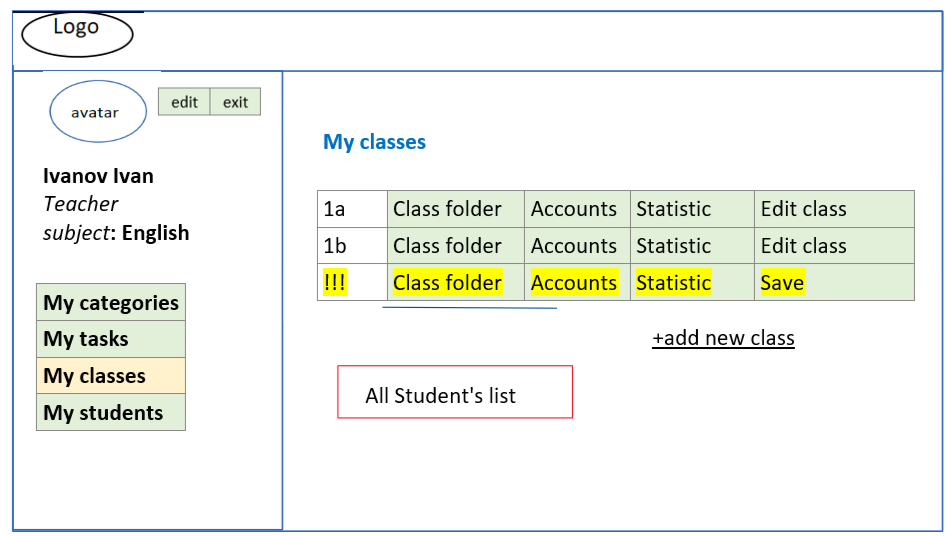


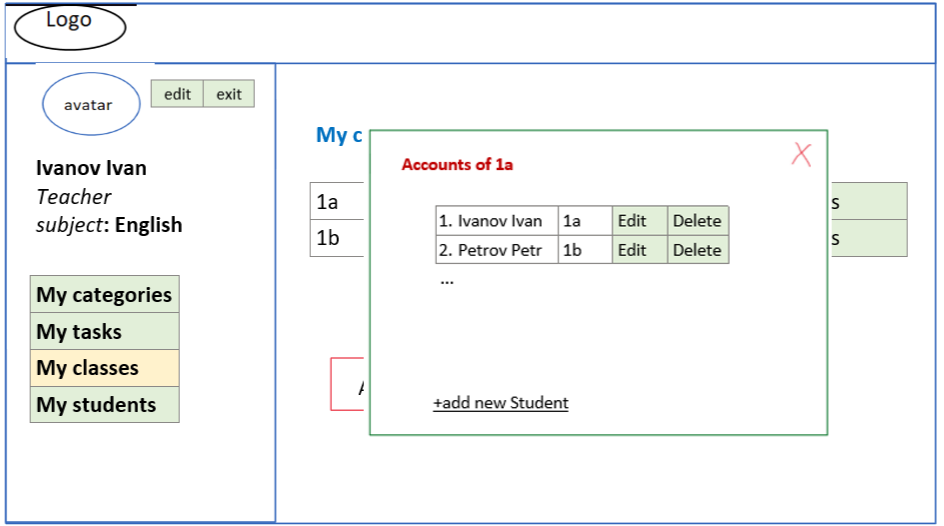




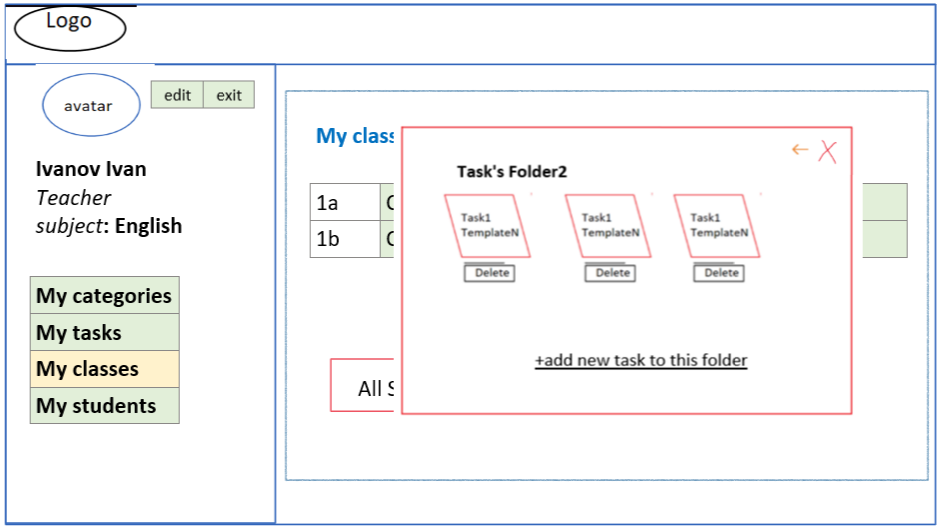


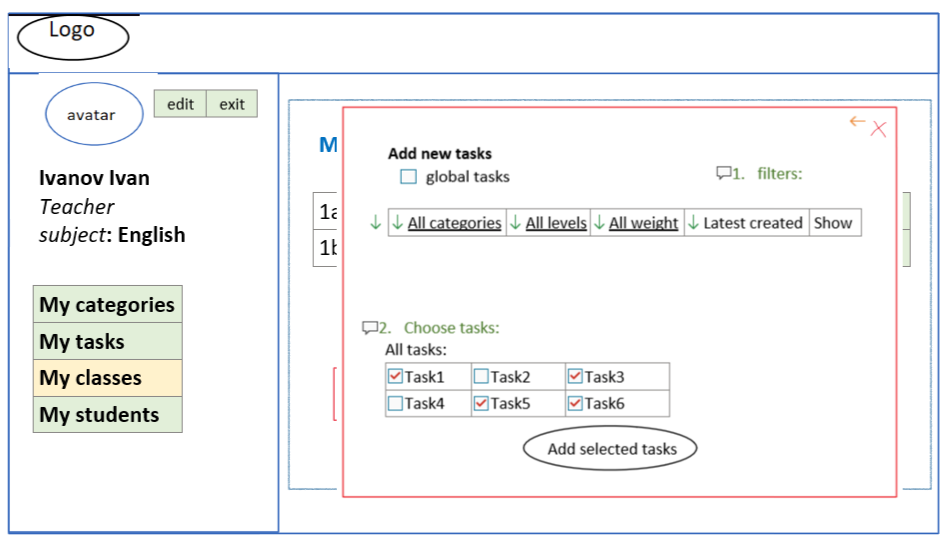




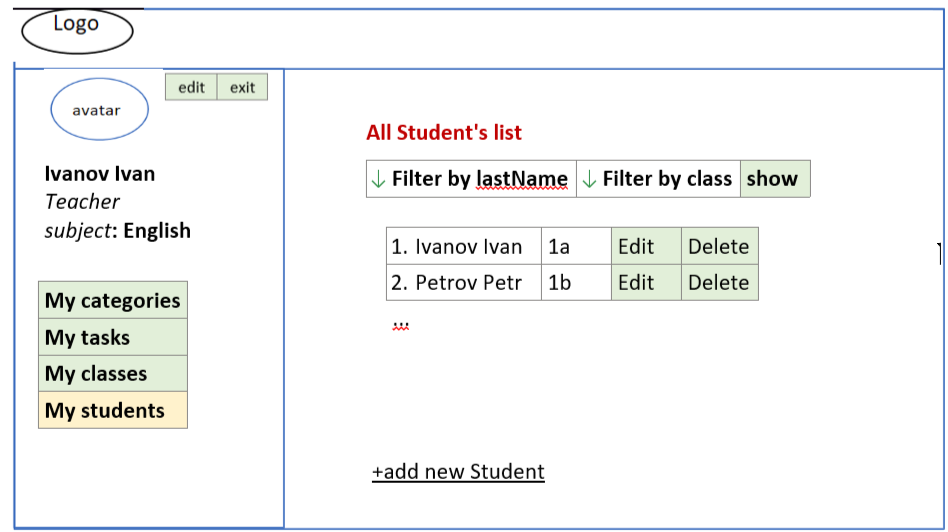






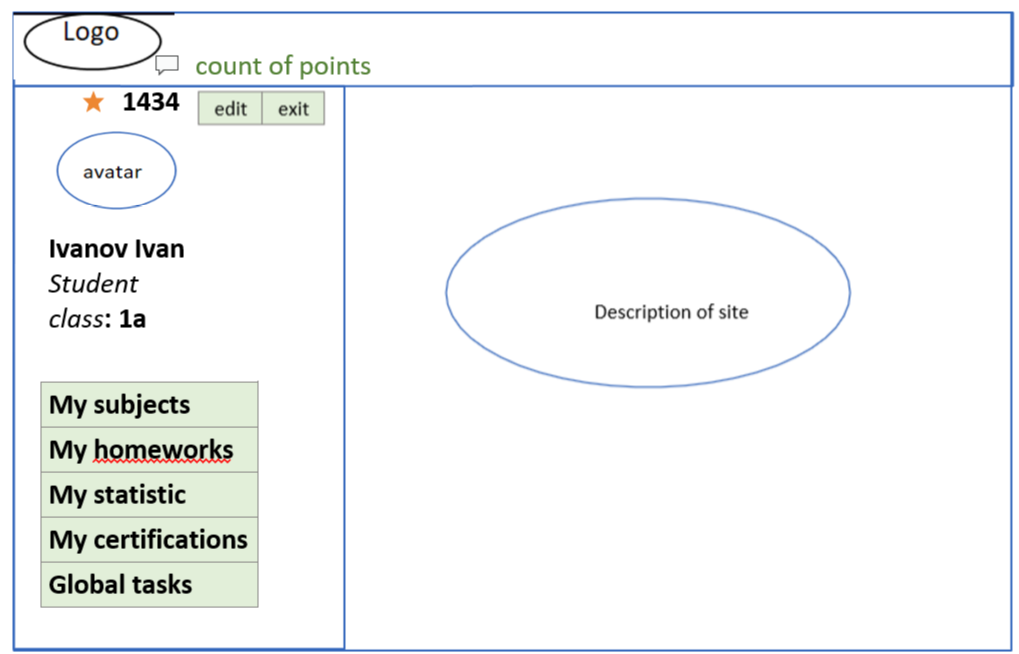


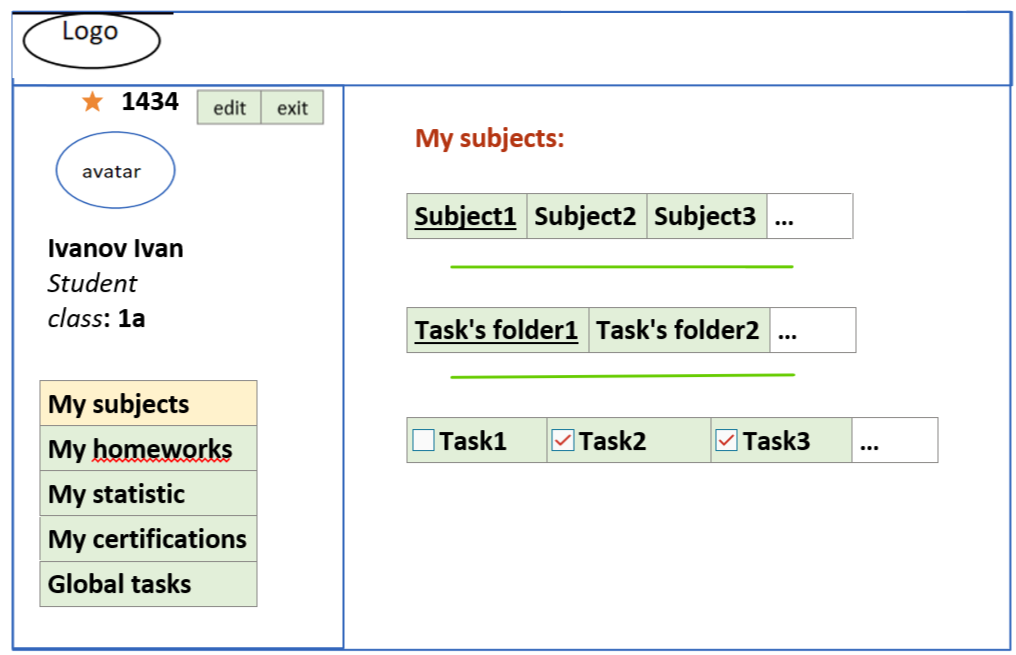




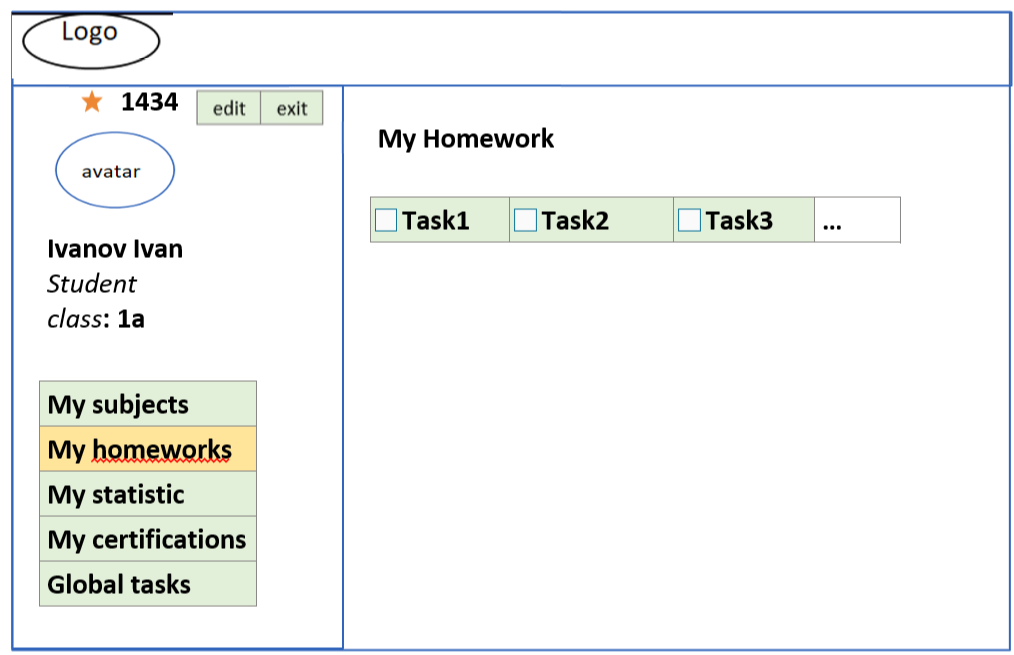


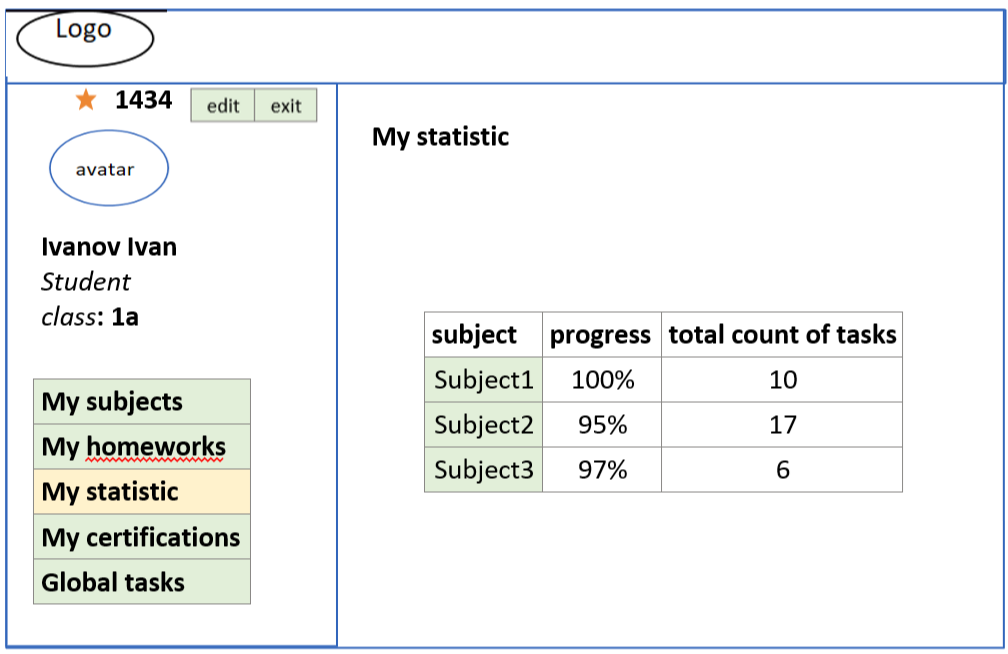
ממשק של התלמיד

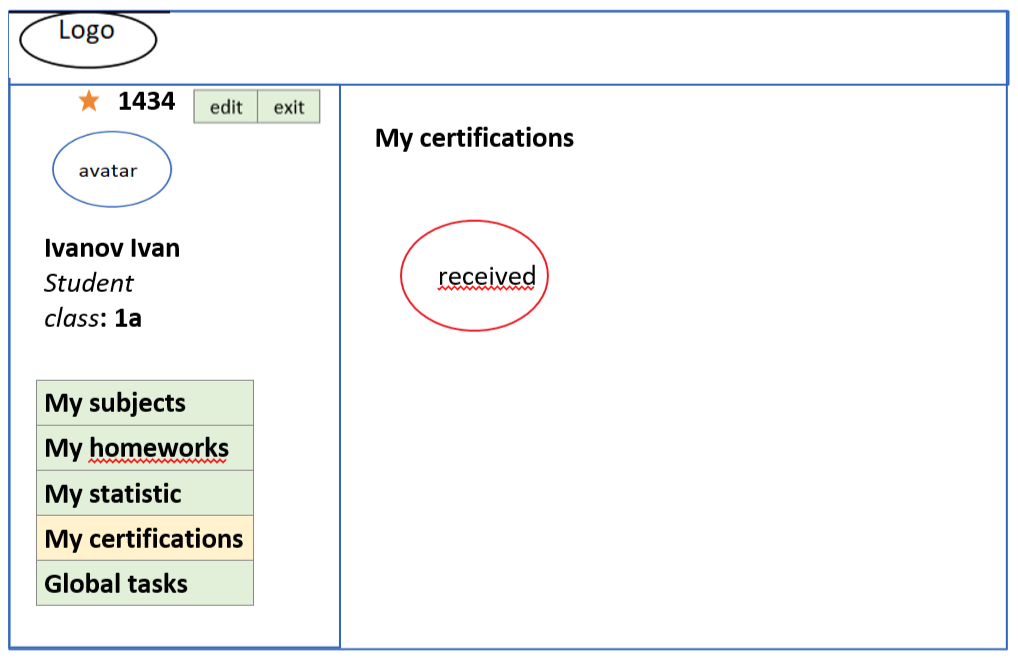


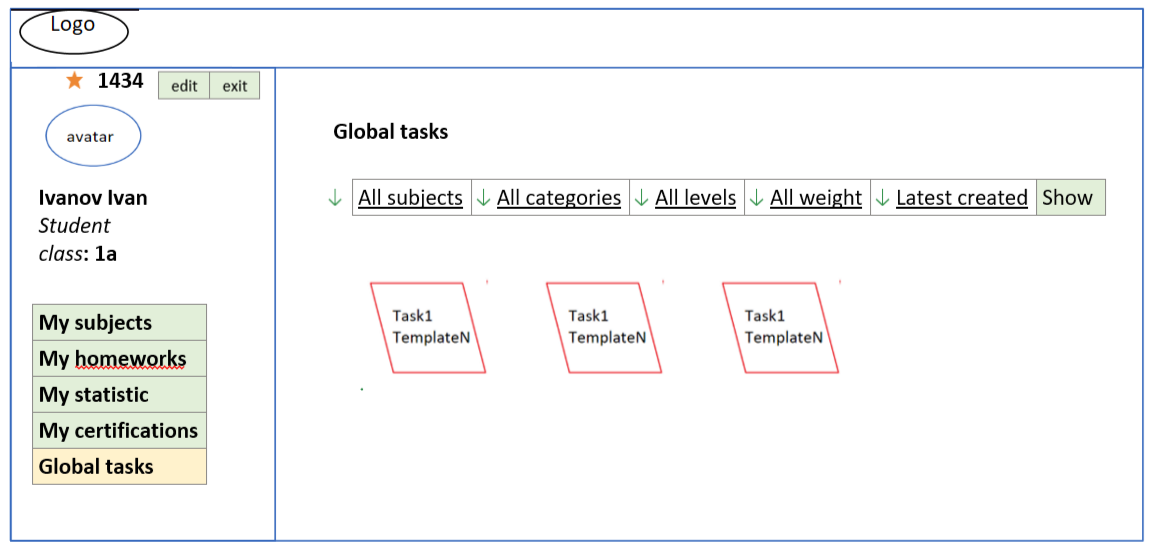






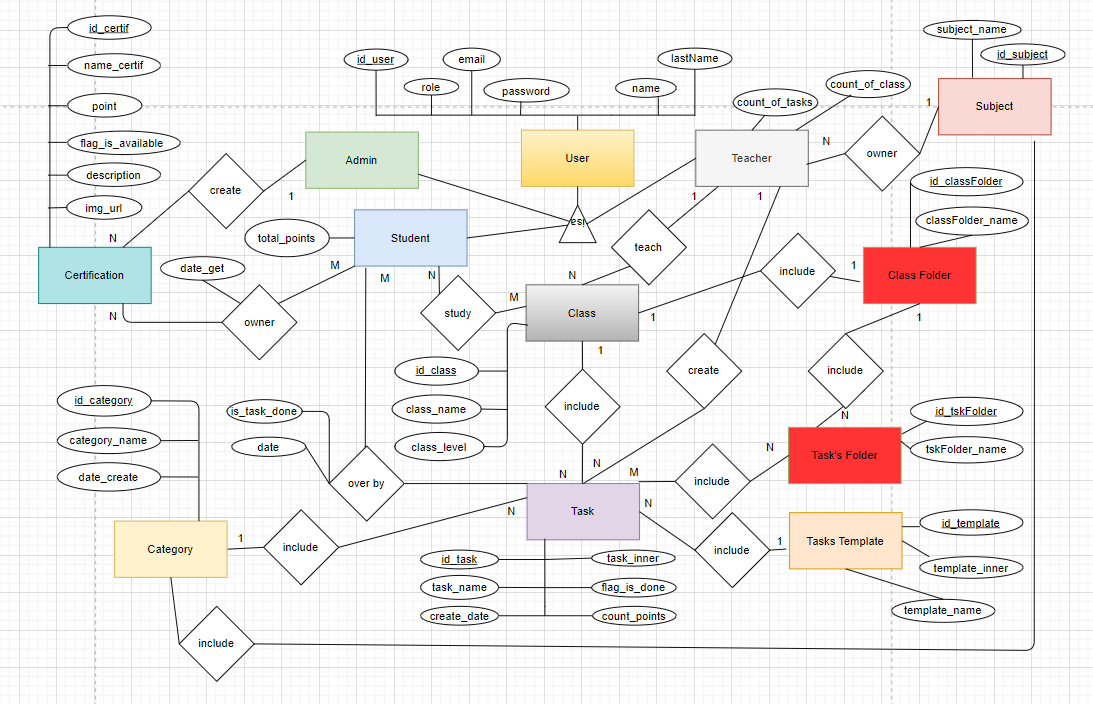






* 1. **ממשקים למערכות אחרות/ API:** כרגע אין צורך בשימוש ב API
  2. **שימוש בחבילות תוכנה:** אין צורך בתוכנה מסויימת להפעלת המערכת.

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



* 1. **נא פרט את שיטת האיחסון:**
* User(id\_user, role, email, password, name, lastNmae)
* Student(id\_user, total\_points, **id\_class**)
* Teacher(id\_user, count\_of\_tasks, **id\_subject**)
* Certification(id\_certif, name\_certif, point, flag\_as\_available, description, img\_url)
* Subject(id\_subject, subject\_name)
* Class folder(id\_classFolder, classFolder\_name)
* Task(id\_task , task\_name, create\_date, task\_inner, flag\_is\_done, count\_points, **id\_teacher**, **id\_class, id\_taskTemplate, id\_category)**
* **Task`s Folder(**id\_tskFolder, tskFolder\_name, **id\_classFolder)**
* **Class(**id\_class, class\_name, class\_level, **id\_teacher)**
* **Category(**id\_category, category\_name, date\_create, **id\_subject)**
* **TaskTemplate(**id\_template, template\_name, template\_inner)
* **Student\_Certification(**id\_user, id\_certif, data\_get)
* **Student\_Task(**id\_user, id\_task, is\_task\_done, date)
* **Student\_Class(**id\_user, id\_class)
* **Task\_Task`sFolder(**id\_task, id\_tskFolder)

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



| **שם** | Login and registration |
| --- | --- |
| **תיאור** | התרחיש מתאר את תהליך ההתחברות וההרשמה למערכת |
| **שחקנים** | user |
| **סטטוס** | מחובר, לא מחובר |
| **תדירות** | כל פעם שהמשתמש מתחבר\נרשם למערכת |
| **עקיבות לדרישות** | 1,2,4,8 |
| **תנאים מקדימים** | פתיחת חלון כניסה למערכת, הזנת שם משתמש וסיסמה(או פרטי הרשמה) |
| **תנאים מסיימים** | הלקוח נכנס למערכת לאחר לחיצה על כפתור ההתחברות\הרשמה |
| **תרחיש הצלחה ראשי** | 1. המשתמש מקליד פרטי התחברות.הרשמה 2. המערכת מאמתת את המשתמש דרך הפרטים שהזין 3. המשתמש יתחבר\נרשם למערכת בהצלחה |
| **הסתעפויות** | המערכת תדפיס למשתמש שאחד הפרטים אינו תקין |

|  |  |
| --- | --- |
| **שם** | To do task student |
| **תיאור** | הסטודנט יכול לעשות את המשימות שקיימות אצלו. |
| **שחקנים** | student |
| **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 תעודות |
| **שחקנים** | student |
| **תדירות** | כל פעם שיש צורך לבדוק סטטיסטיקה או תעודות שקיבל |
| **עקיבות לדרישות** | 20,21,22,23 |
| **תנאים מקדימים** | 1. קיימים נתונים להורדה |
| **תנאים מסיימים** | הופעת גרף סטטיסטיקה של משימות/קבלת מידע על הישגים של הסטודנט |
| **תרחיש הצלחה ראשי** | 1. בחירת אופציה רצוייה להורדה(סטטיסטיקה\תעודות) 2. הופעת גרף הסטטיסטיקה /הופעת תעודותשל הסטודנט |
| **הסתעפויות** | במקרה ויש תקלה תופיע הודעת שגיאה |

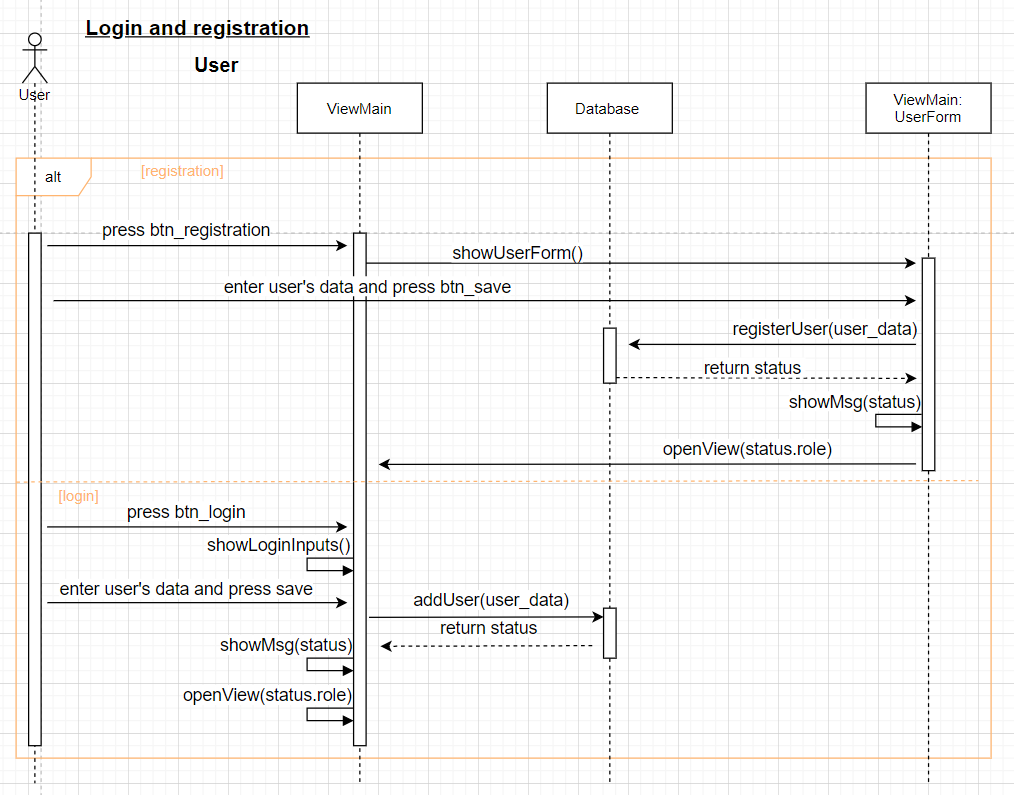
|  |  |
| --- | --- |
| **שם** | update user |
| **תיאור** | עדכון פרטי משתמש |
| **שחקנים** | אדמין, מורה, תלמיד (משתמש כללי) |
| **תדירות** | כל פעם שיש צורך למשתמש לעדכן פרטים |
| **עקיבות לדרישות** | 5,7,9 |
| **תנאים מקדימים** | משתמש קיים במערכת, אימות פרטים |
| **תנאים מסיימים** | הפרטים מתעדכנים במערכת, מופיעה הודעת הצלחה |
| **תרחיש הצלחה ראשי** | 1. המשתמש מזין את הפרטים החדשים שרוצה לשנות 2. לוחץ על שמור 3. המערכת מאמתת שהפרטים תקינים ושם המשתמש לא קיים כבר במערכת 4. הפרטים מתעדכנים |
| **הסתעפויות** | במידה ואחד מהפרטים אינו תקין תופיע הודעה מתאימה והפרטים לא משתנים |

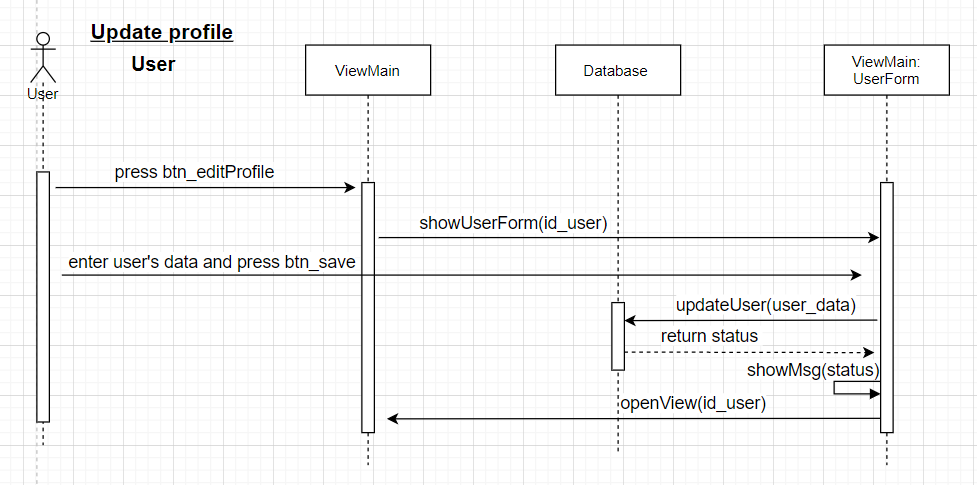
|  |  |
| --- | --- |
| **שם** | students management(teacher) |
| **תיאור** | יצירה או מחיקת משתמשים, מיון משתמשים, בחירת משימות לכיתה והצגת/הורדת סטטיסטיקה על מעבר המשימה |
| **שחקנים** | מורה |
| **תדירות** | כל פעם שיש צורך באחד מהדברים |
| **עקיבות לדרישות** | 11,12,24 |
| **תנאים מקדימים** | קיימים 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 |

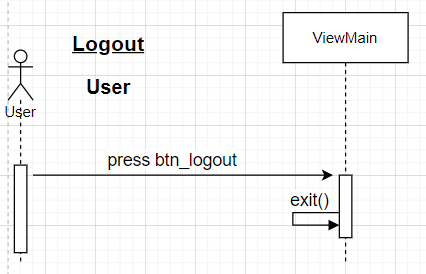
|  |  |
| --- | --- |
| **שם** | Task manage(teacher) |
| **תיאור** | בחירת קטגוריה למשימות, בחירת תבנית למשימה וניקוד שמקבלים על המשימה או מחיקת משימה |
| **שחקנים** | מורה |
| **תדירות** | כל פעם שמייצרים משימה חדשה/מוחקים משימה |
| **עקיבות לדרישות** | 14 |
| **תנאים מקדימים** | נבחרה תבנית רצויה, 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. בחירת פעולה רצויה 3. הזנת\שינוי פרטים  4. שמירת פרטים |
| **הסתעפויות** | במידה והתהליך לא מצליח תופיע הודעת שגיאה מתאימה ולא יתווסף/ימחק כלום |

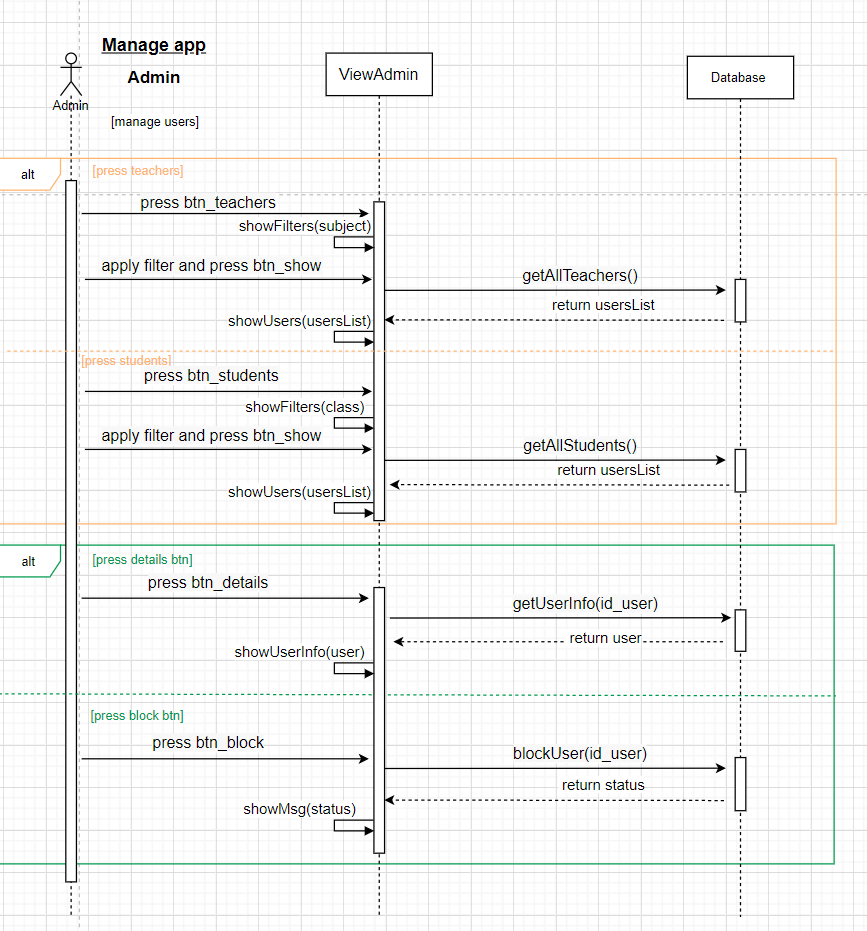
|  |  |
| --- | --- |
| **שם** | logout |
| **תיאור** | יצאת משתמש מהמערכת |
| **שחקנים** | מורה, תלמיד, אדמין |
| **תדירות** | כל פעם שמישהו רוצה לצאת מהמערכת |
| **עקיבות לדרישות** | 3 |
| **תנאים מקדימים** | משתמש צריך להיות מחובר למערכת |
| **תנאים מסיימים** | המשתמש יצאה מהמערכת |
| **תרחיש הצלחה ראשי** | 1. לחיצה על כפתור יציאה מהמערכת 2. המשתמש יצא בהצלחה מהמערכת |
| **הסתעפויות** | אם התהליך לא הצליח תופיע הודעת שגיאה |

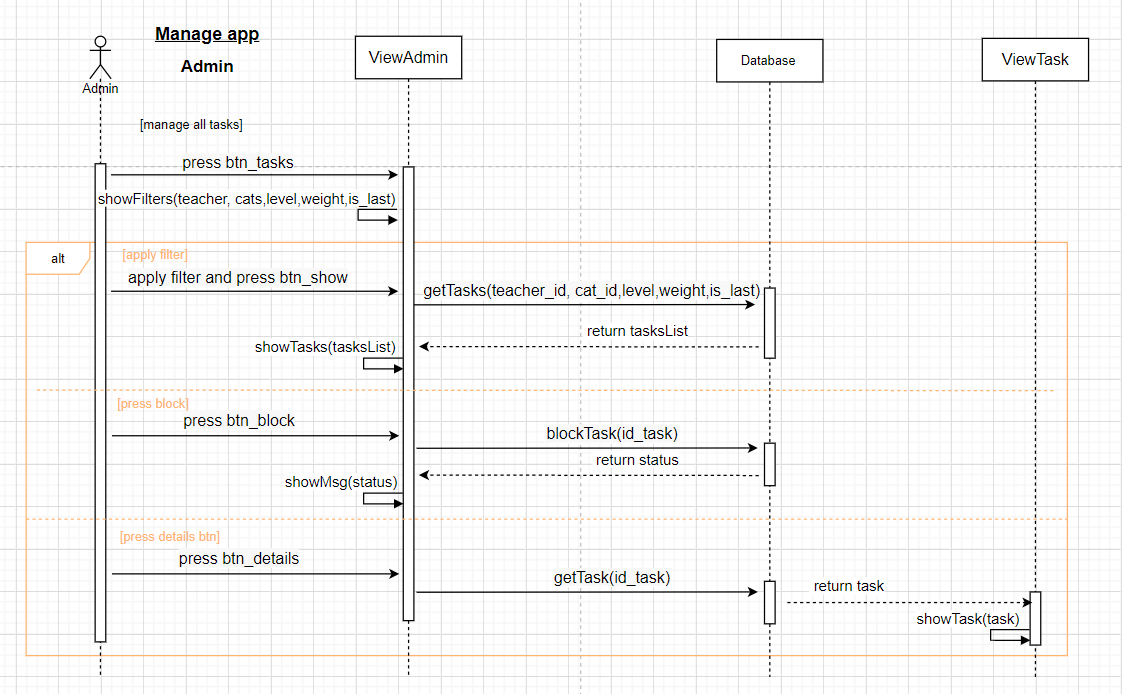
* 1. **Sequence diagram**

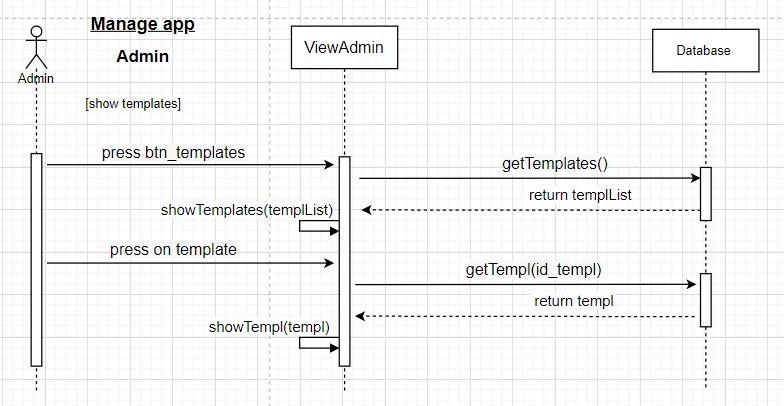


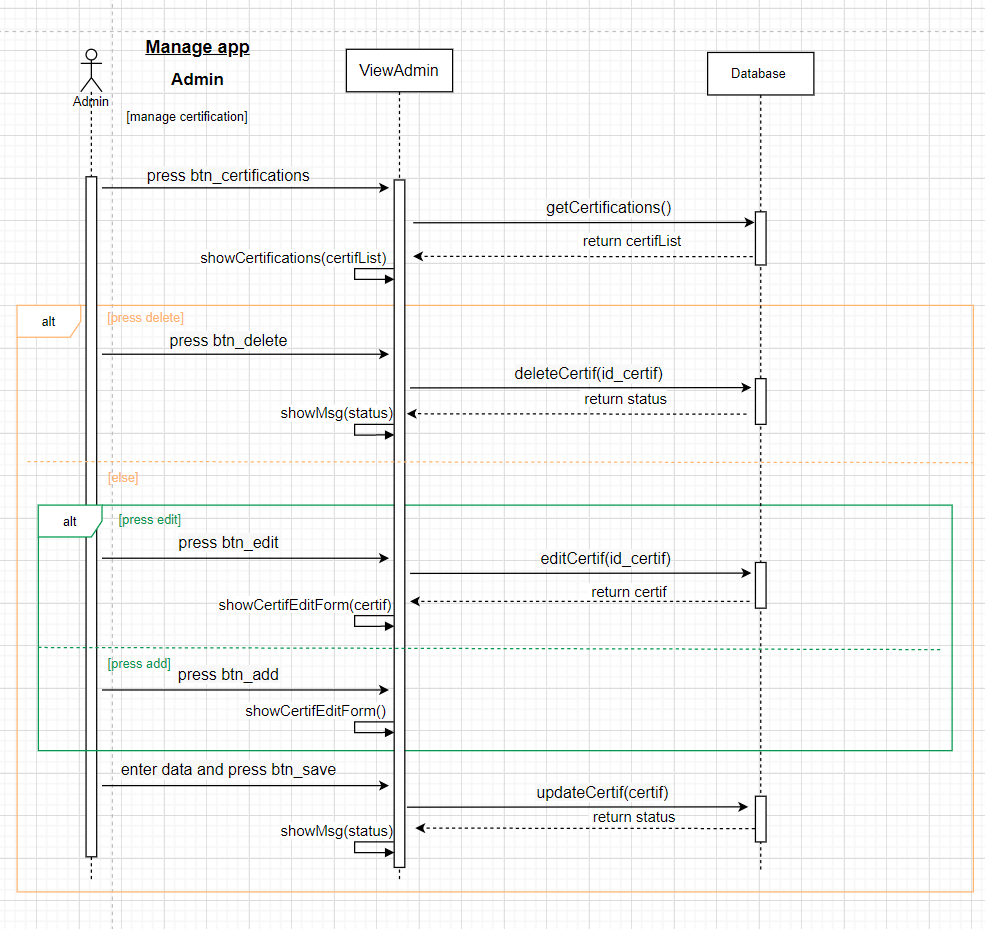


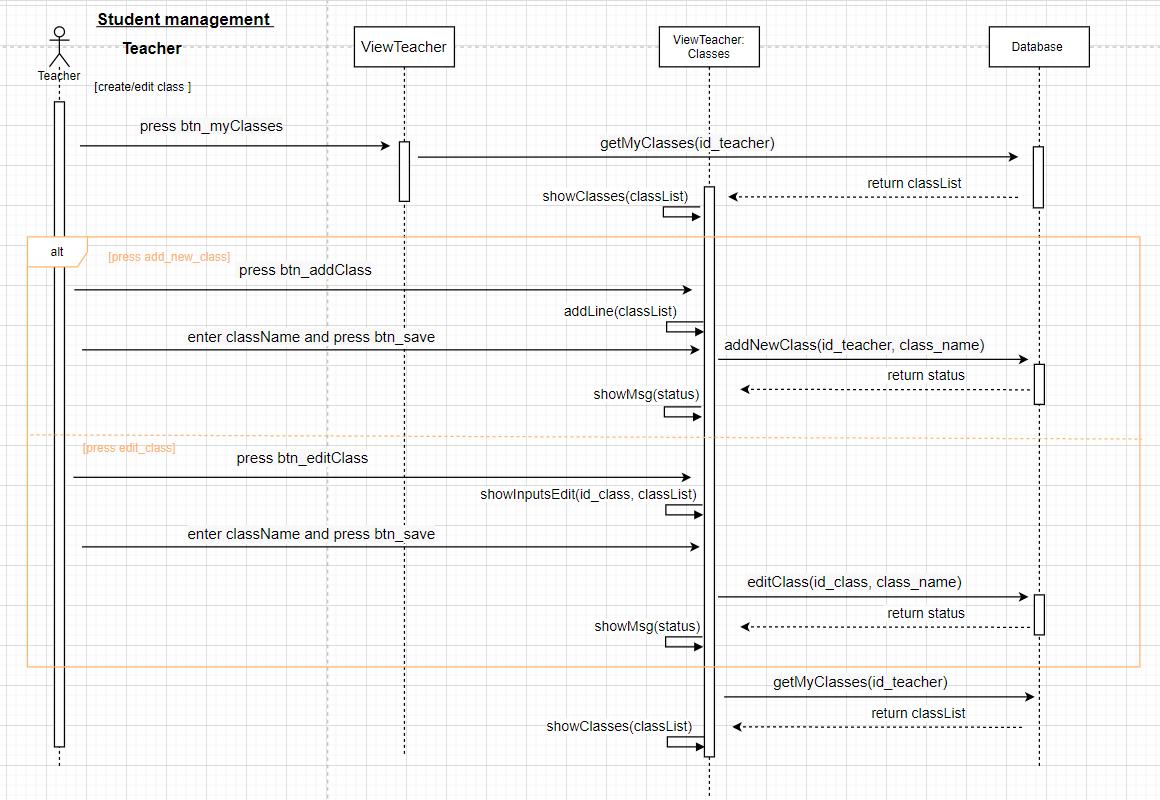




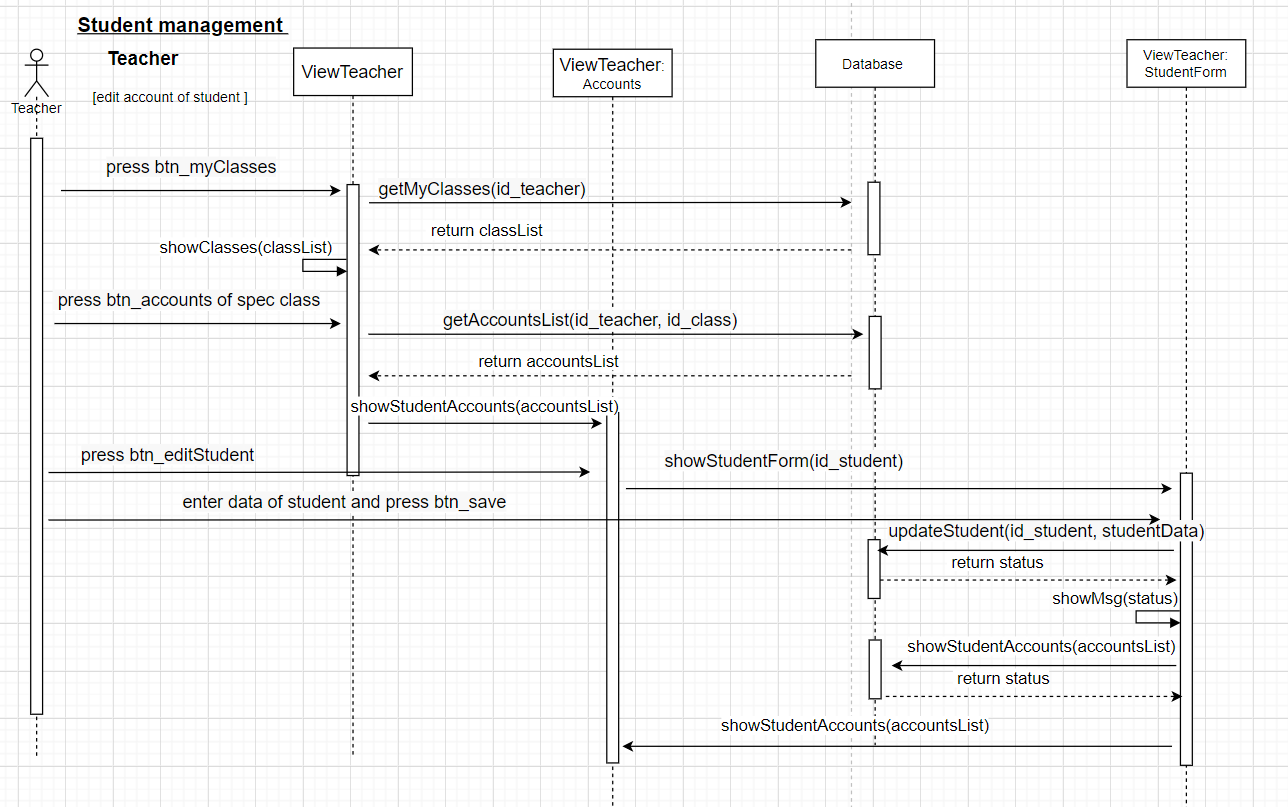


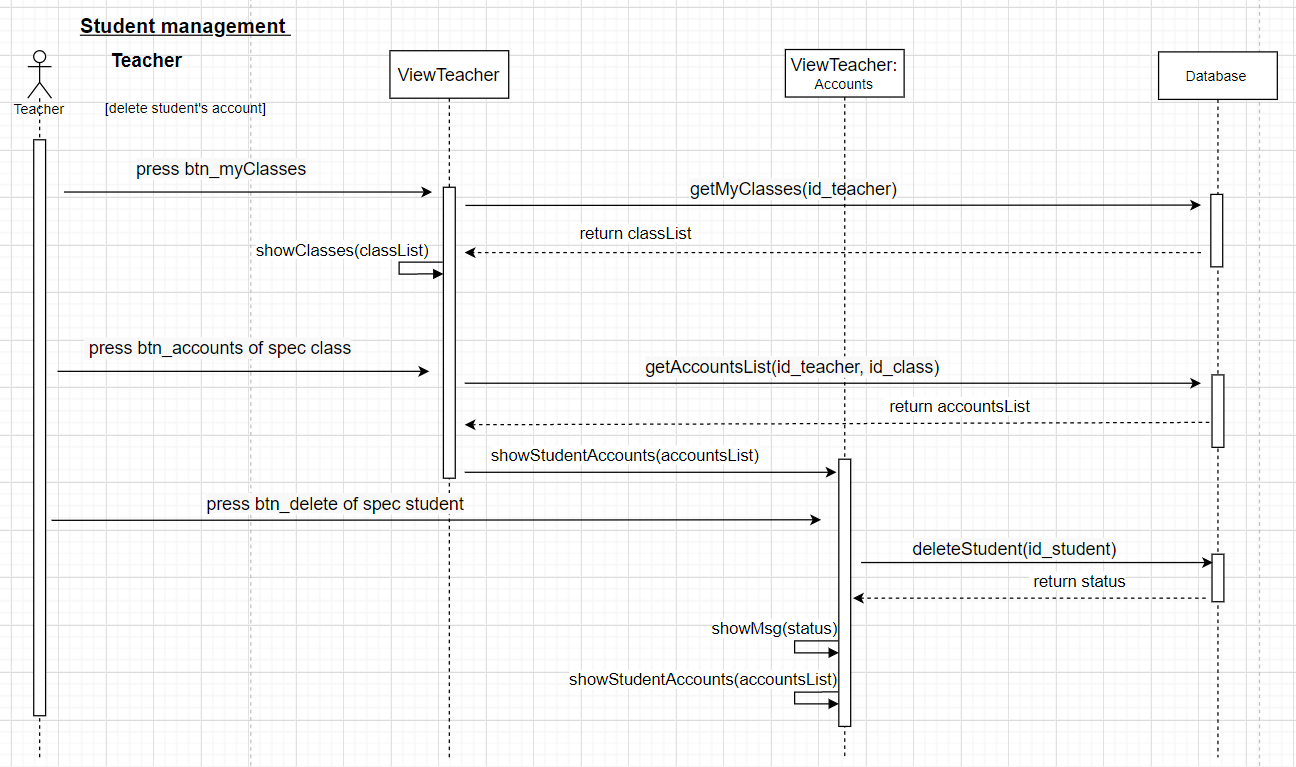


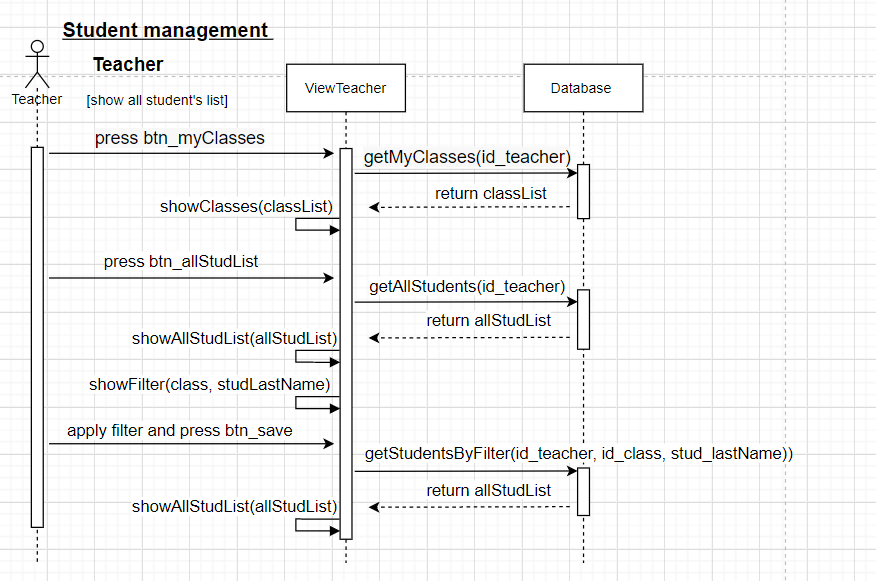


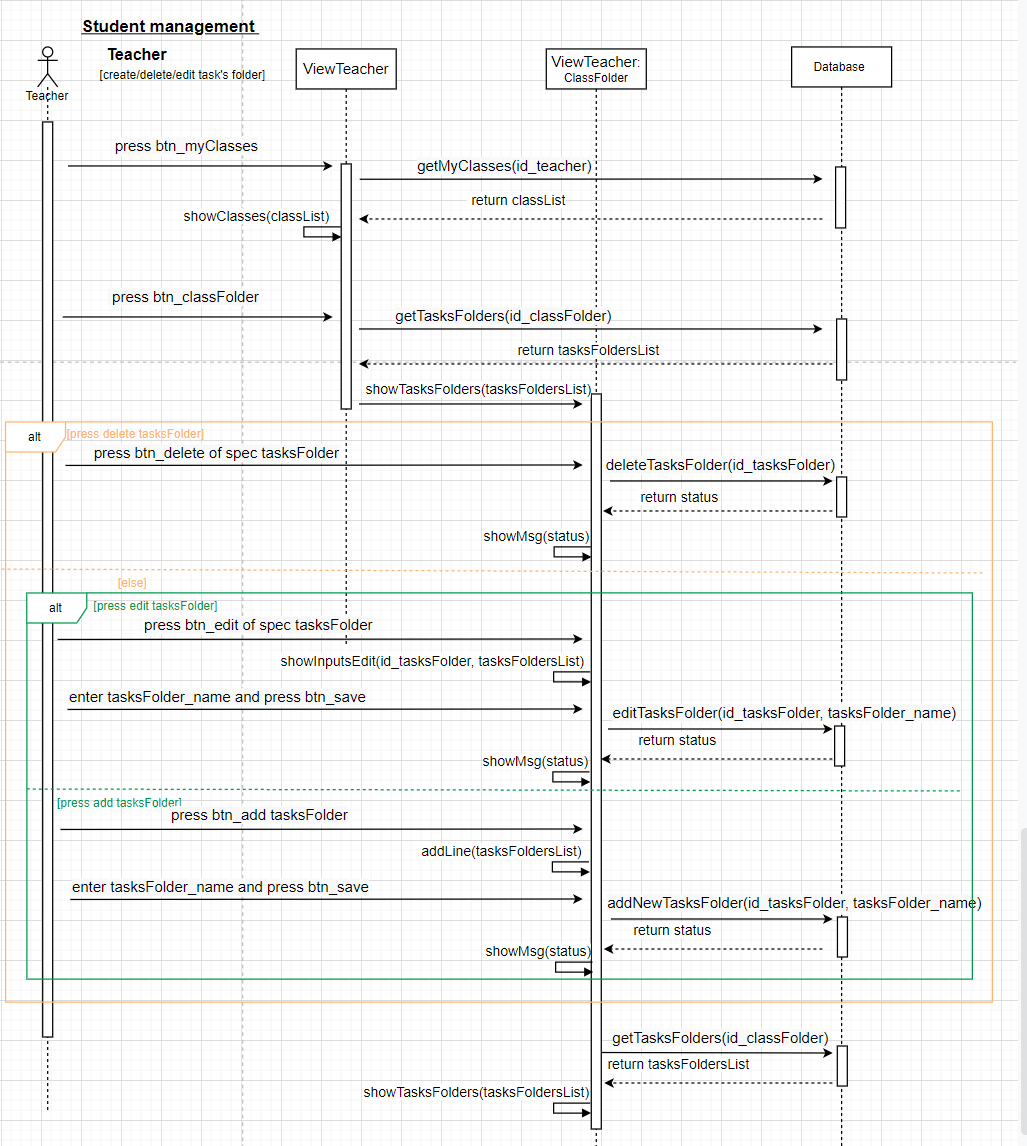


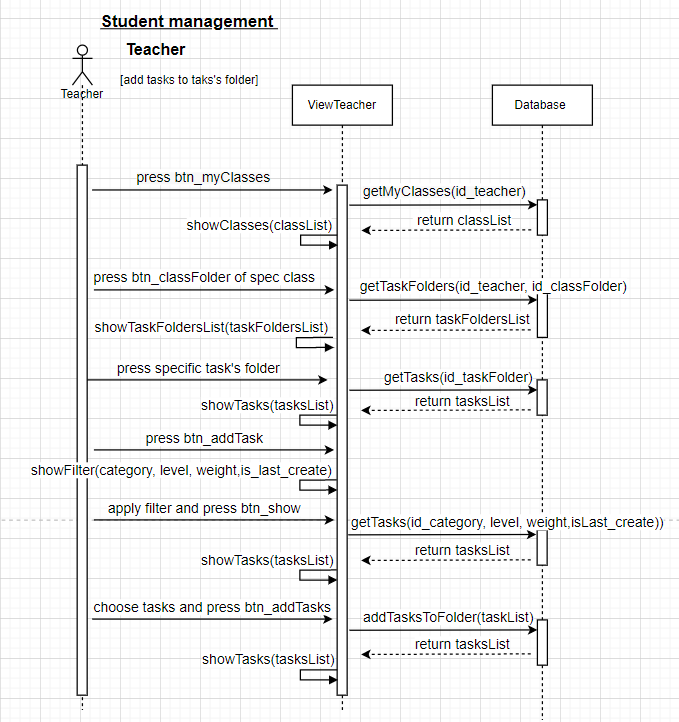




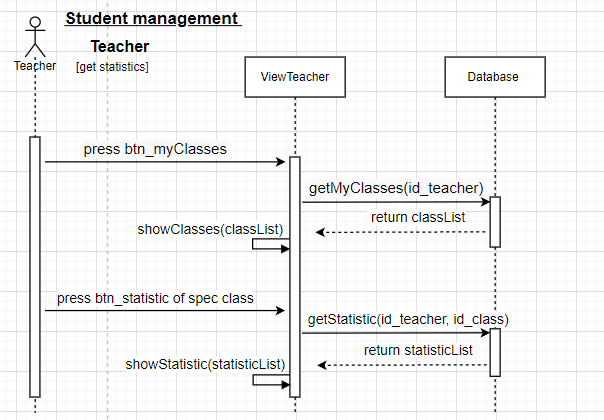


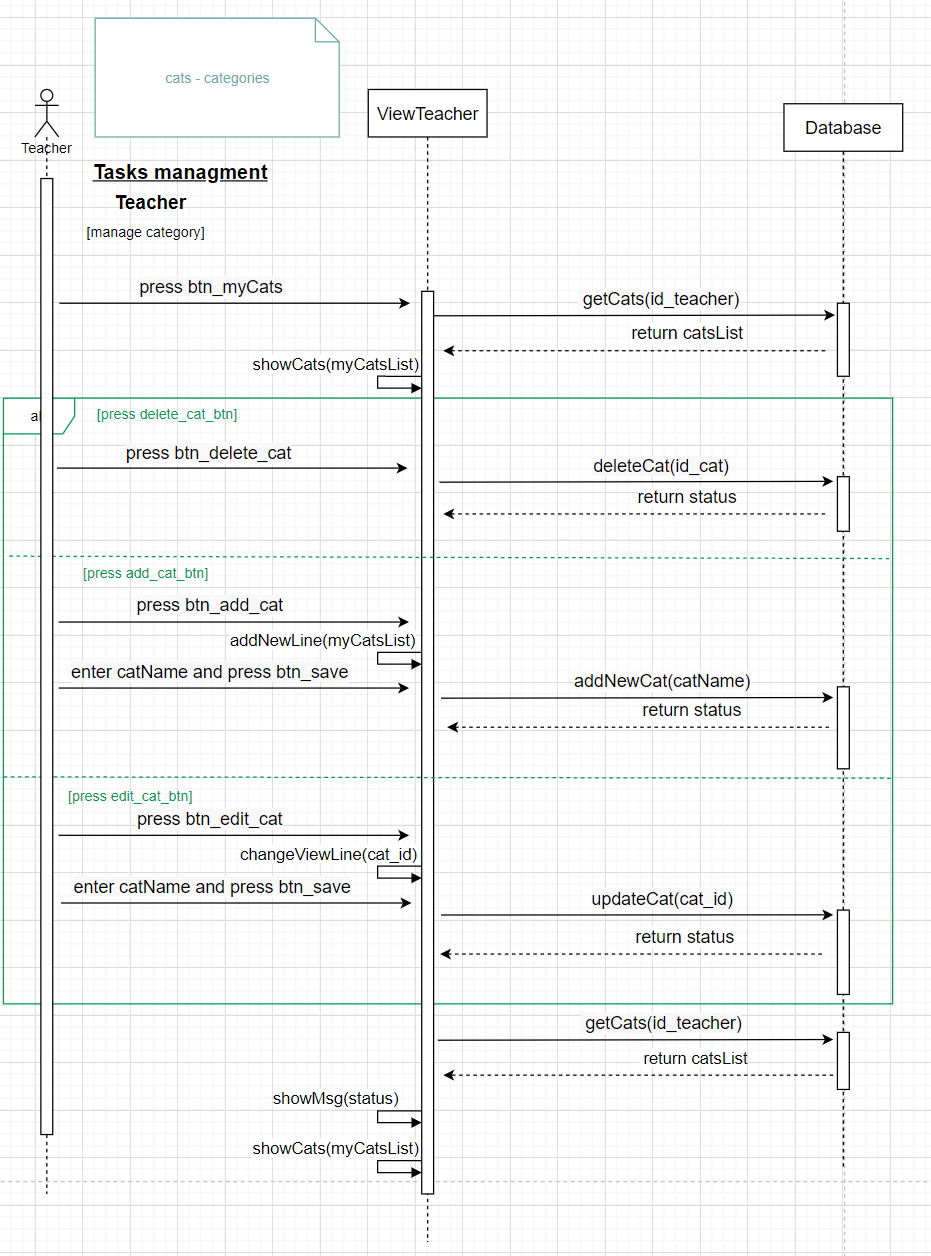


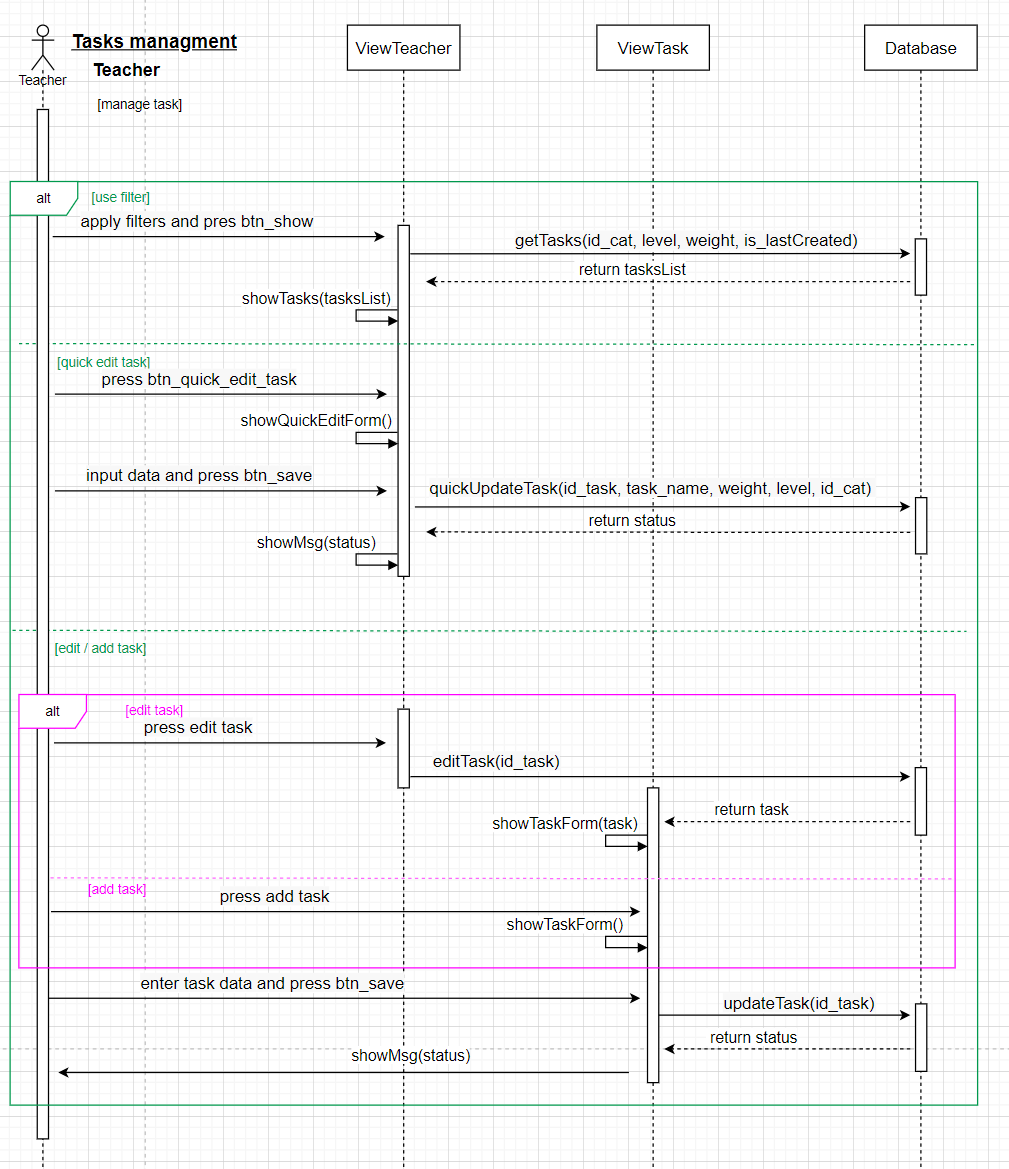


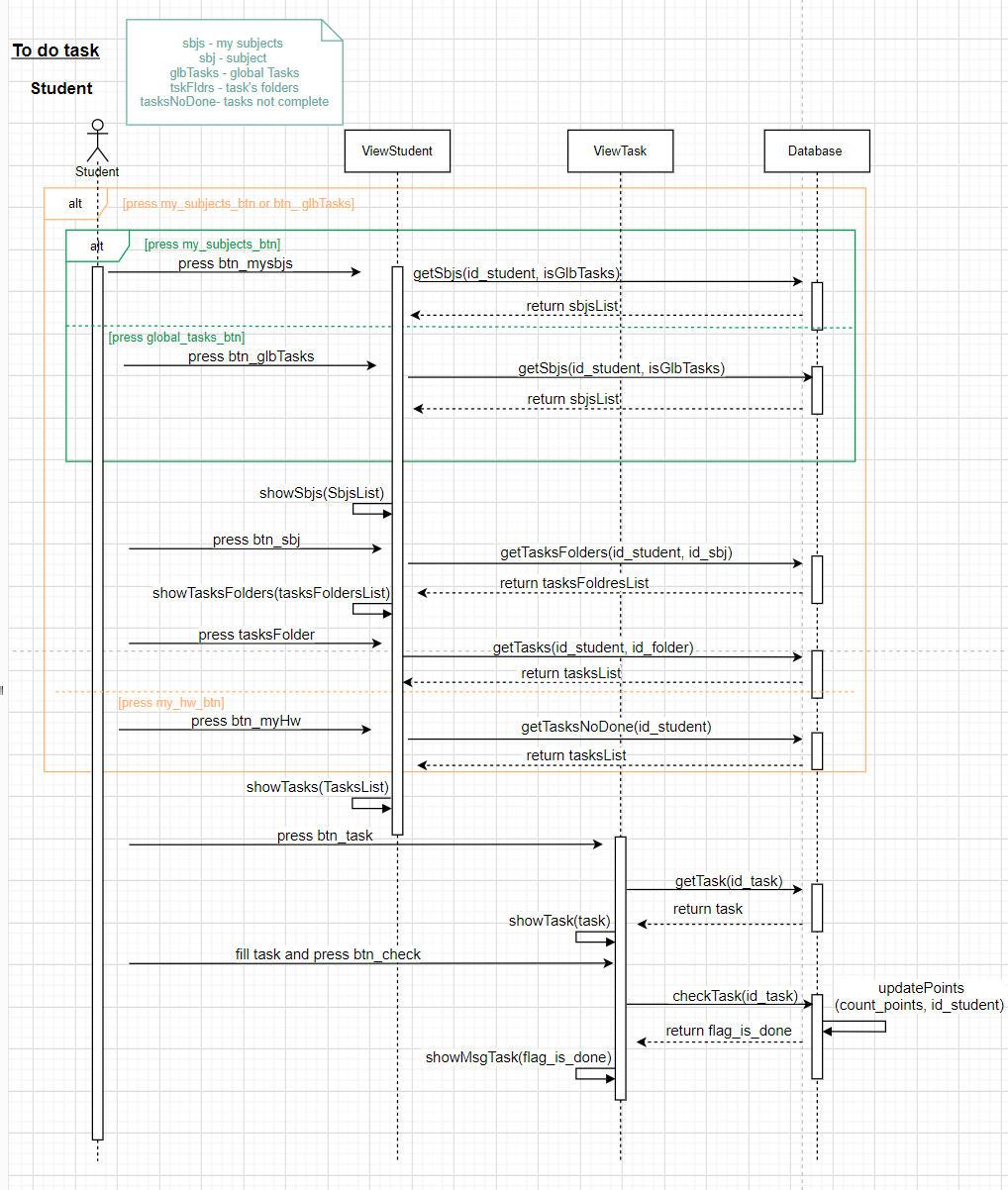


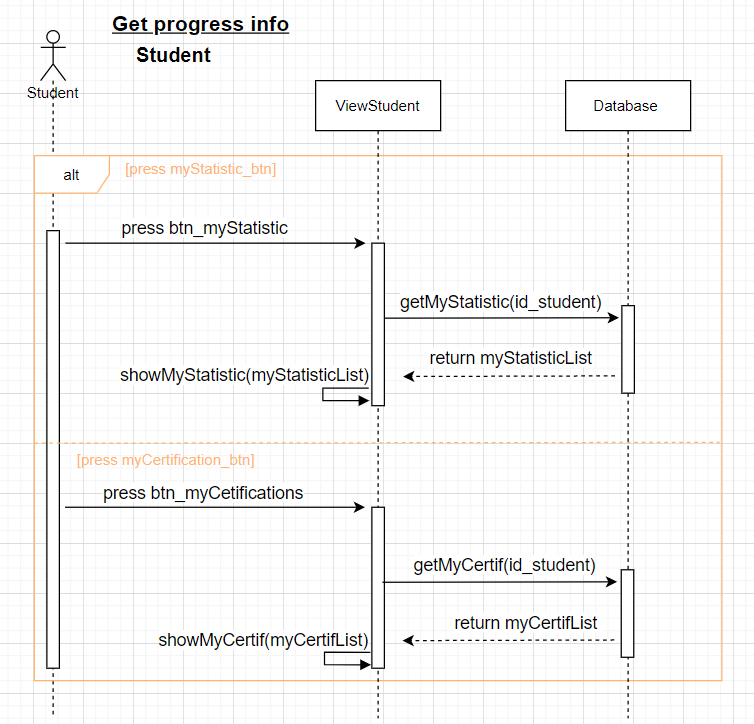




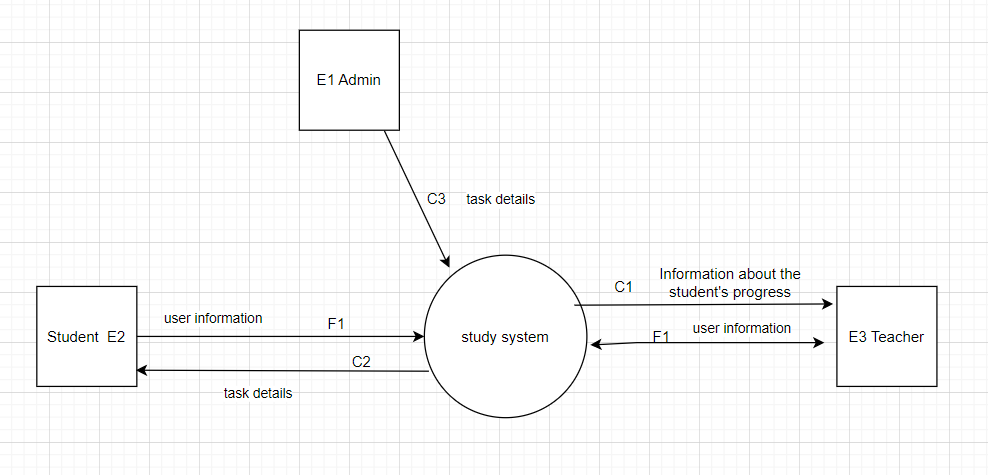


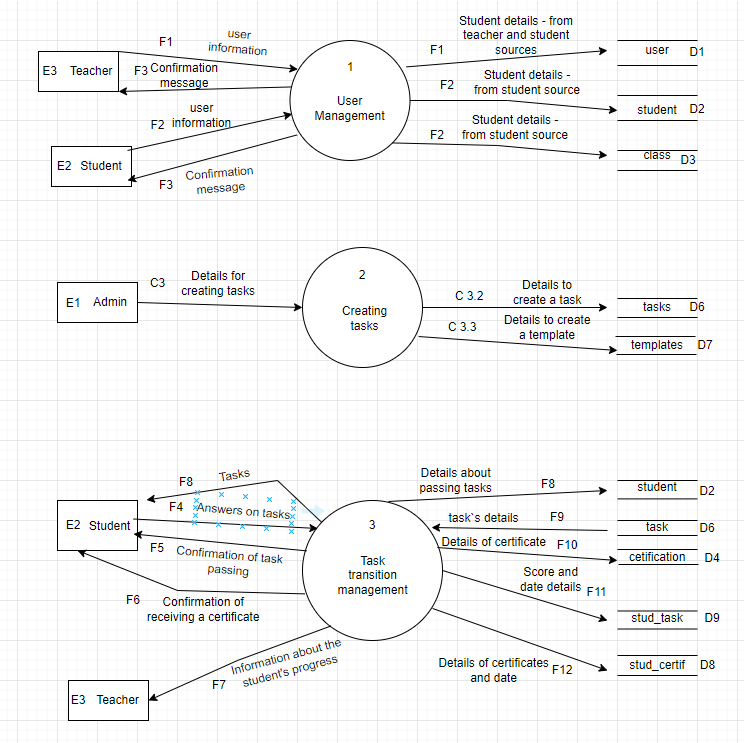


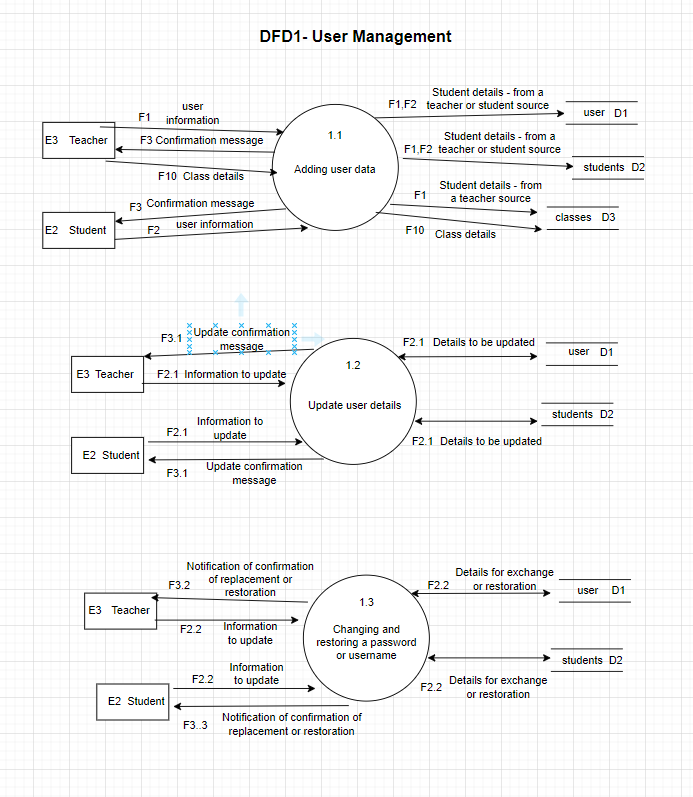


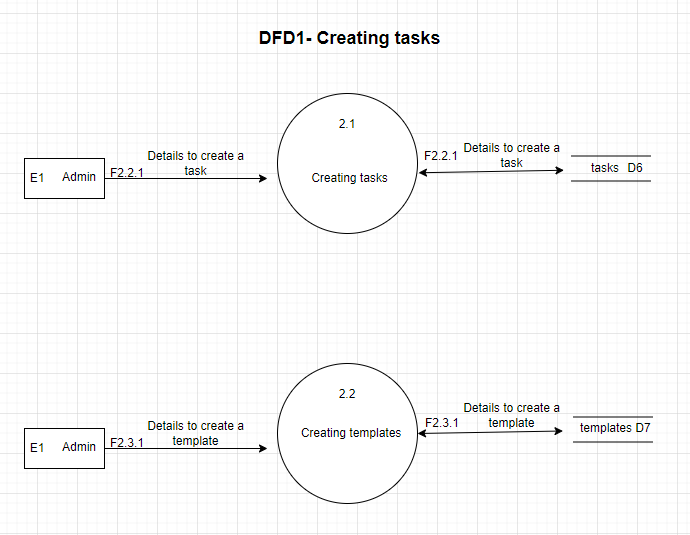


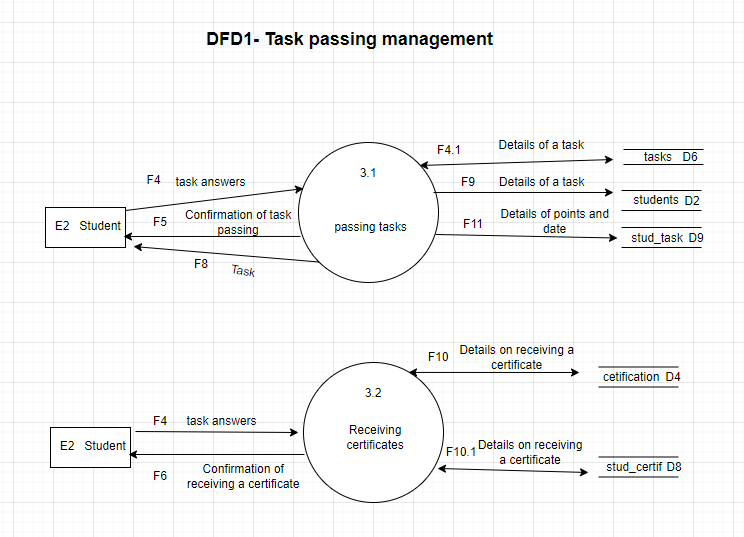
* 1. **DFD0+DFD1:**



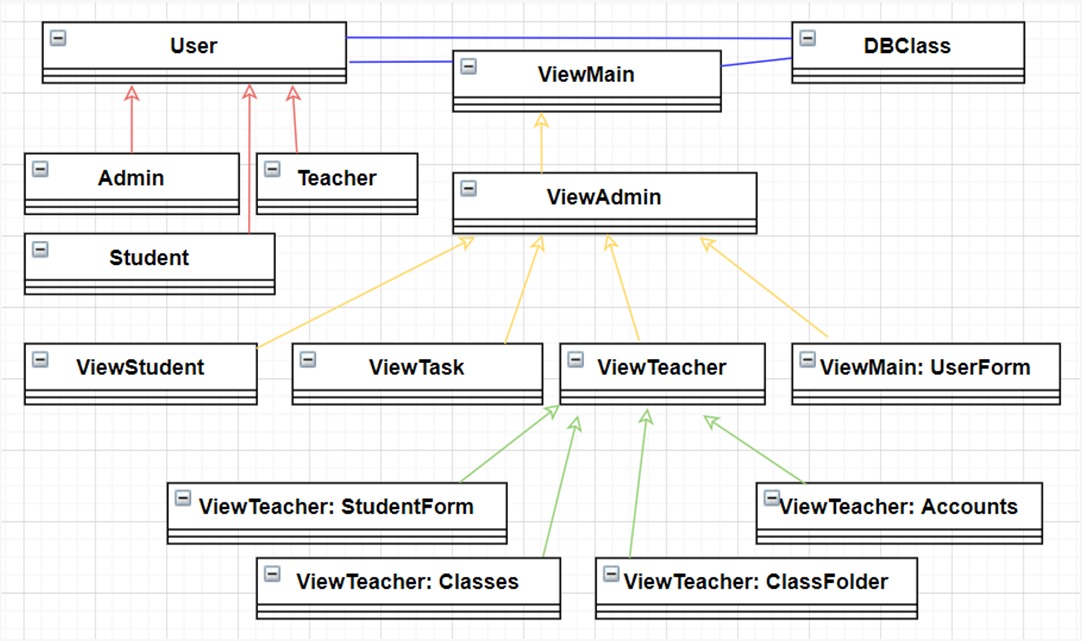


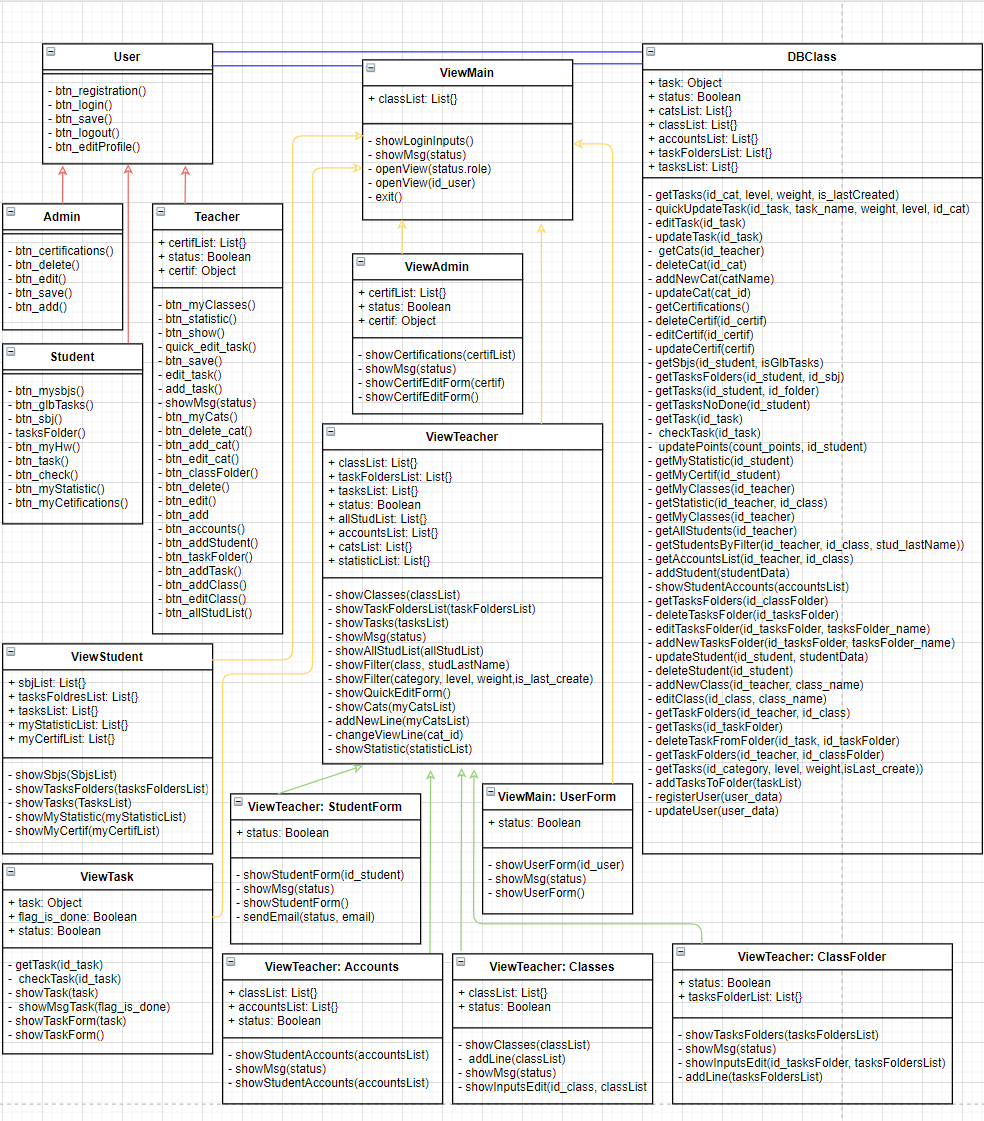




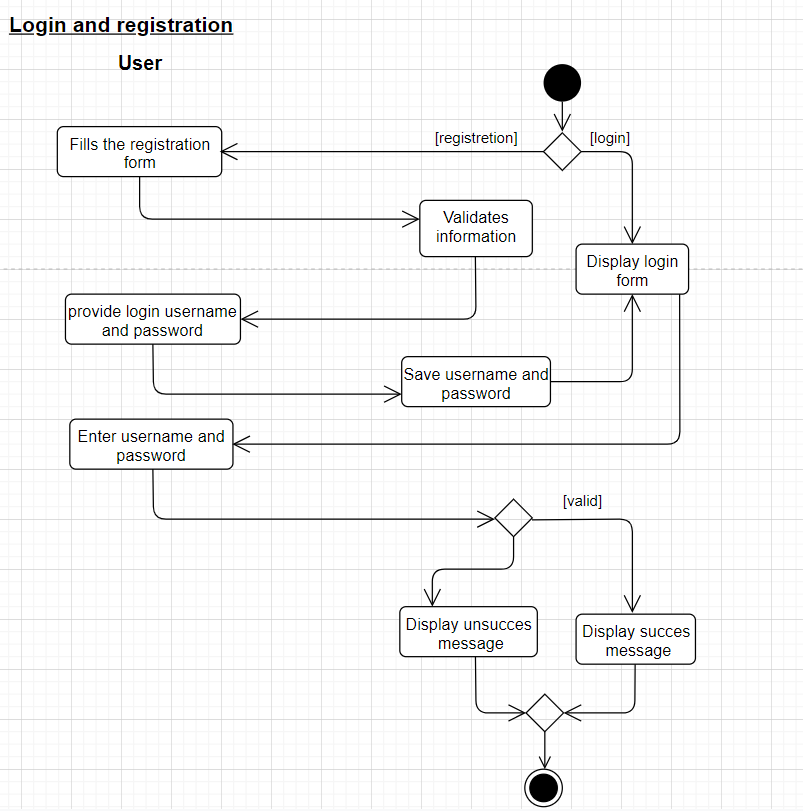


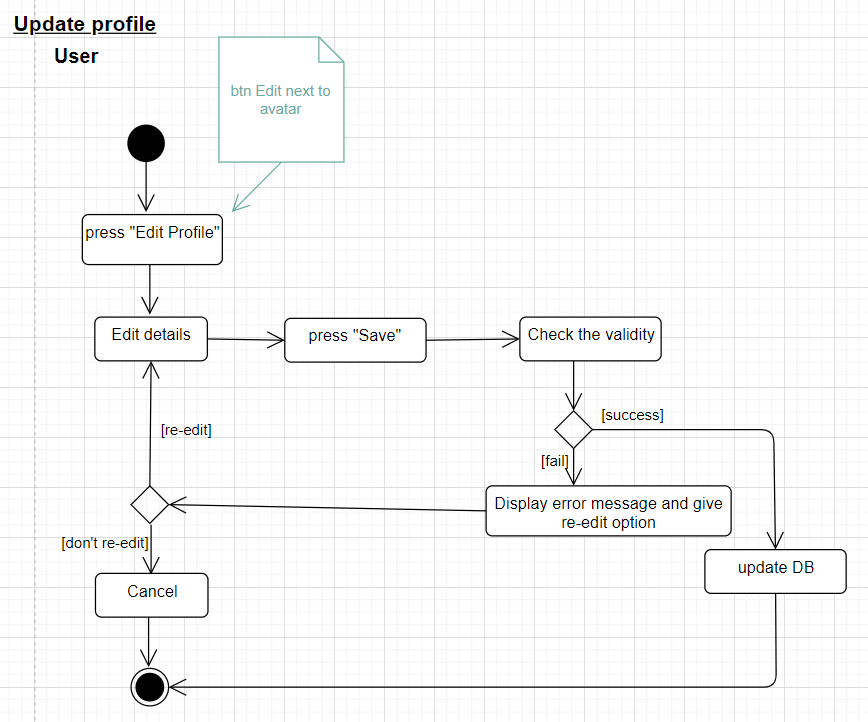
* 1. **תרשים מחלקות**

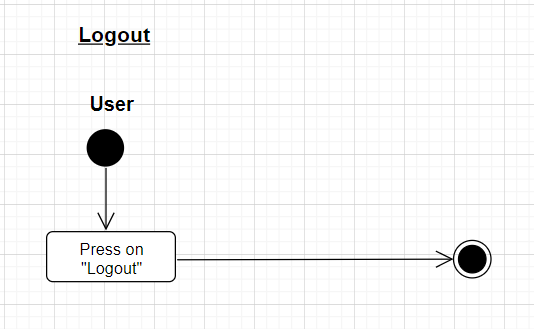


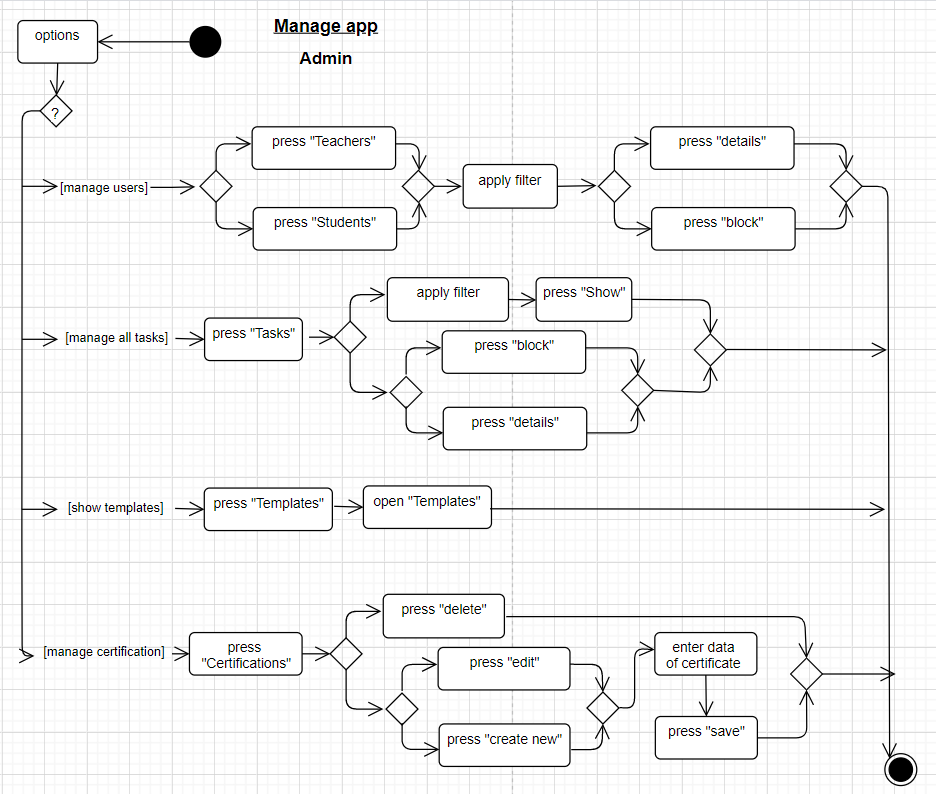


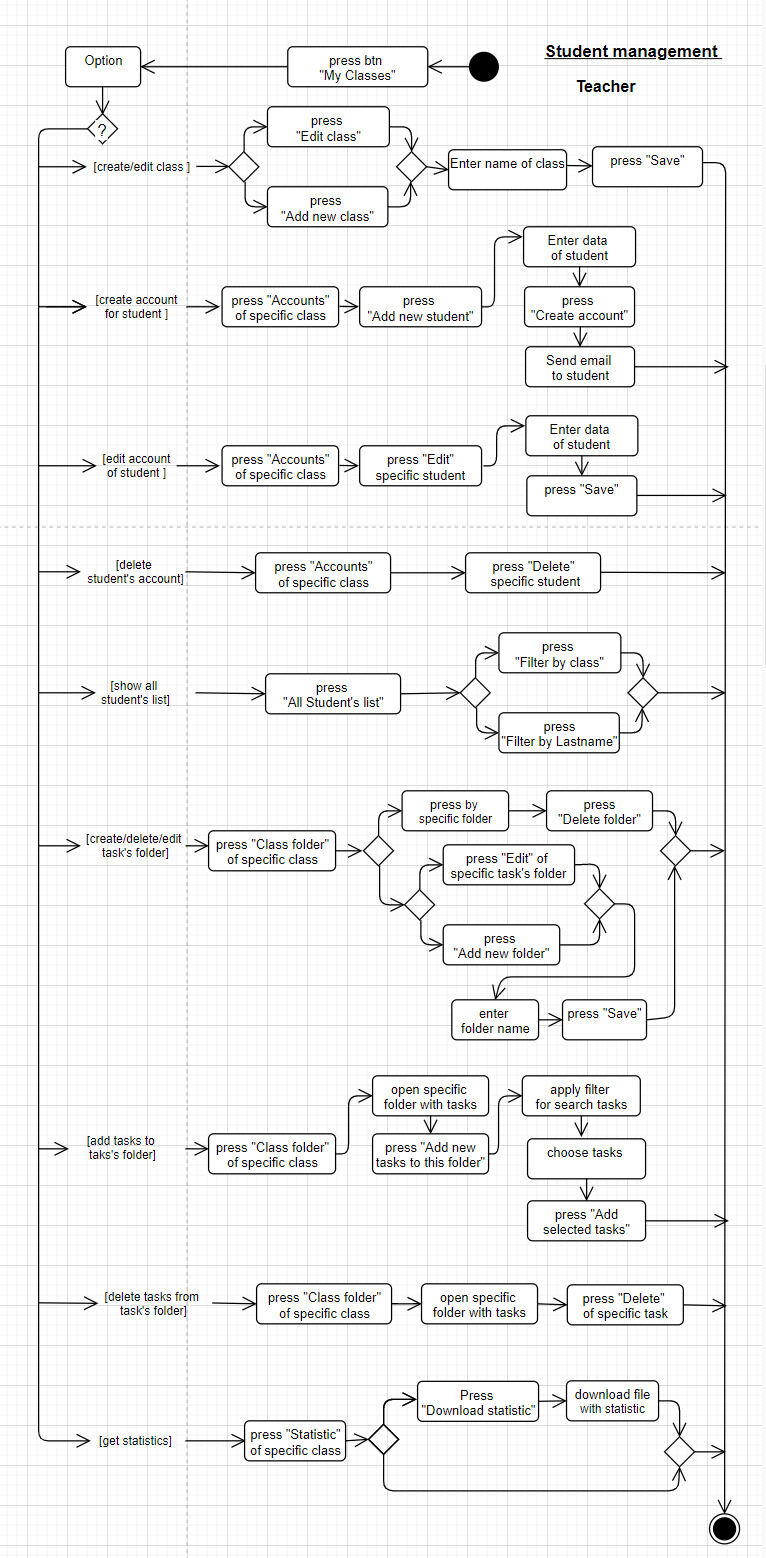
1. תיאור המרכיב האלגוריתמי – חישובי
   1. **איזה בעיה בא לפתור, איך יפתור?**

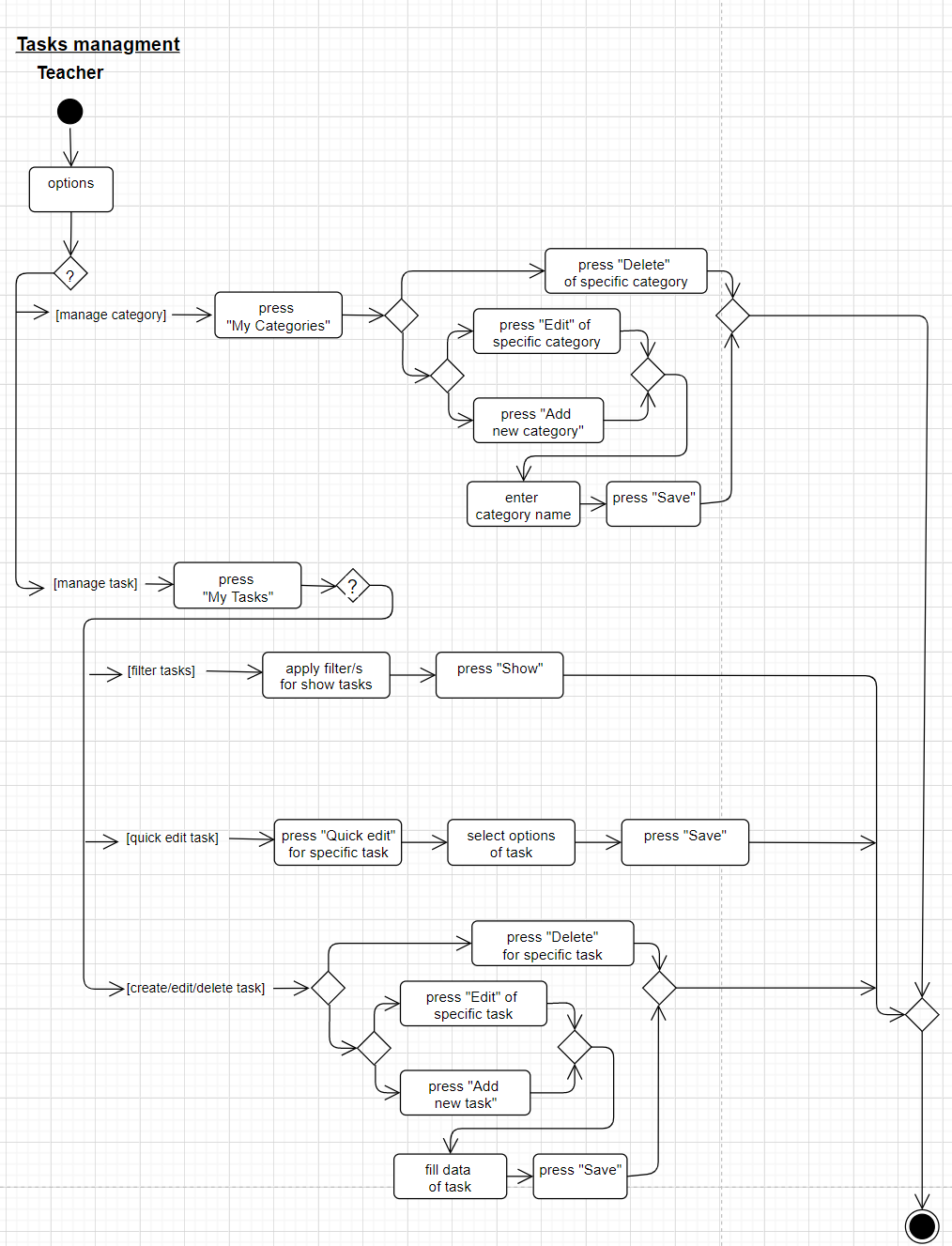


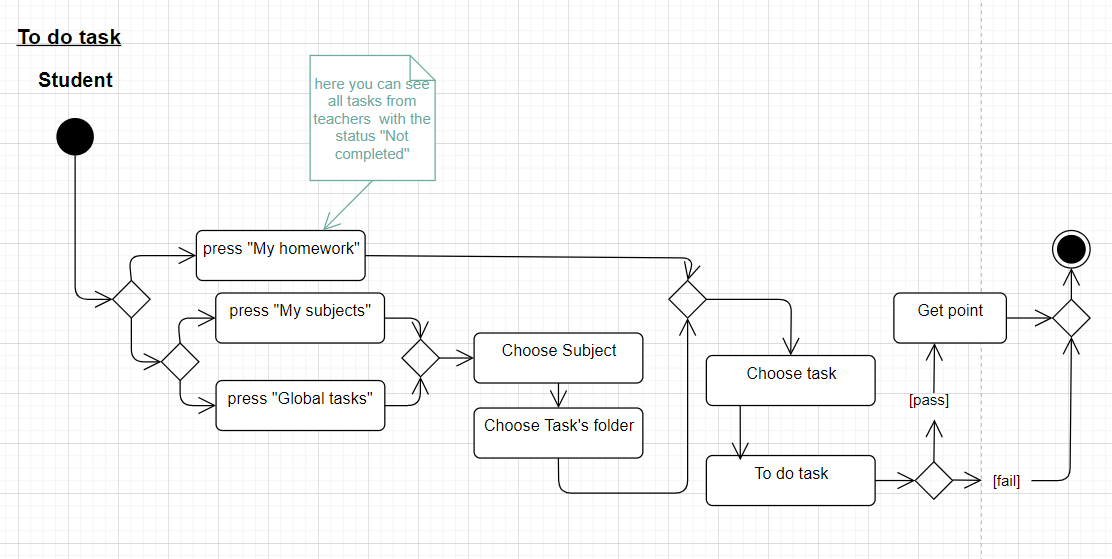


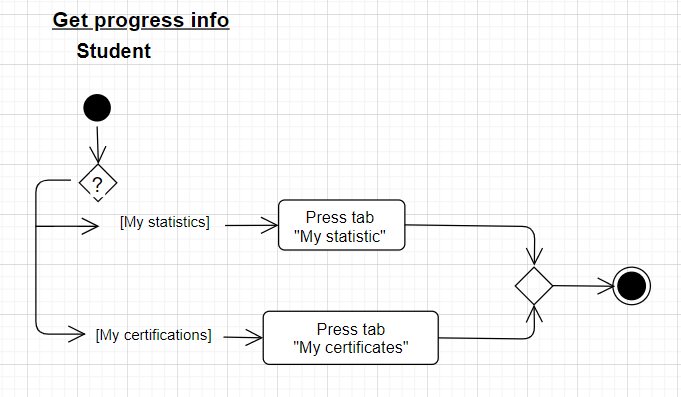












1. תיאור/התייחסות לנושאי אבטחת מידע:

הצפנות: **כל הסיסמאות ונתוניהם יהיו מוצפנים.**

גישה לאתר: **נשתמש ב** SSL **כדי להצפין את המידע ושיהיה מעבר בטוח.**

משתמש לא רשום: **אין אפשרות למשתמש לא רשום להשתמש במערכת.**

משתמש רשום(תלמיד): **משתמש רשום יכול להתחבר למערכת. ההתחברות היא ע"י הזנת מייל וסיסמה, המערכת תעשה בדיקה האם המשתמש אכן קיים בבסיס הנתונים, ותיבדק הסיסמה לאחר מכן.**

משתמש רשום(צורה): **תהיה לו אפשרות ליצור משימות, מעקב על סטטיסטיקות, פתיחת כיתות חדשות במערכת והוספת תלמידים במקרה הצורך.**

משתמש רשום(אדמין): **תהיה לו אפשרות להוסיף או למחוק משתמשים במקרה הצורך, יוכל להוסיף טמפלייטים ותעודות יהיה לו מעקב אחר משימות ויוכל להחליט אם להשאיר או לחסום משימה מססויימת.**

1. משאבים הנדרשים לפרויקט:
   1. **מספר שעות המוקדש לפרויקט, חלוקת עבודה בין חברי הצוות:** לזוג 600

300 לכל אחד, כל אחד יעשה חלק מסויים ובמקביל השני יעבוד על חלק אחר וכל פעם נאחד את הדברים.

* 1. **ציוד נדרש:** חיבור לאינטרנט, מחשב(עם כל חלקיו כמו מקלדת,עכבר ומסך)
  2. **תוכנות נדרשות:**

**VScode –** לפיתוח fronfEnd ו backend

**MySQL –** מערכת לבסיס נתונים

* 1. **ידע חדש שנדרש ללמוד לצורך ביצוע הפרויקט:**

יבוא ויצוא קבצים מסוגים שונים(כגון pdf, excel).

לחזק את הידע ב nodeJS, react.

ללמוד שימוש וניהול נכון של מבנה הנתונים בענן.

* 1. **ספרות ומקורות מידע:**

Stackoverflow: <https://stackoverflow.com/>

MDN: <https://developer.mozilla.org/en-US/>

Codepen: [https://codepen.io](https://codepen.io/)

1. תכנית עבודה ושלבים למימוש הפרויקט:
2. ניתוח פרויקט ובניית הצעת פרויקט עד לתאריך 03/03/2023
3. תכנון ממשק משתמש עד לתאריך 14/04/2023
4. כתיבת קוד עד לתאריך 15/07/2023
5. בדיקות תוכנה עד לתאריך 15/08/2023
6. כתיבת ספר פרויקט עד לתאריך 23/08/2023
7. הגשת ספר פרויקט 01/09/2023

1. תכנון הבדיקות שיבוצעו

**14.1 בדיקות תהליכיות ברמת משתמש בהן נדרשת המערכת לעמוד:**

|  |  |  |  |
| --- | --- | --- | --- |
| מס' דרישה | נוסח הדרישה | סוג דרישה | תיאור אופן בדיקת הדרישה |
| 1 | המערכת תכלול 3 סוגי משתמשים(אדמין, מורה, תלמיד) | דרישה תפעולית | לבדוק שאכן קיימים 3 סוגי משתמשים ויש לכל אחד גישה למערכת. |
| 2 | המערכת תכלול כניסת משתמש(שם משתמש וסיסמה) | דרישה תפעולית | נסיון כניסה עם כל אחד מהמשתמשים.   1. נסיון כניסה עם אותו שם משתמש על כל המשתמשים.(אמור להציג הודעת שגיאה) 2. נסיון כניסה עם שם משתמש וסיסמה מיוחדת לכל אחד מהמשתמשים.(אמור להכנס למערכת בהצלחה) |
| 3 | קיימת אפשרות יציאה מהמערכת | דרישה תפעולית | נסיון לצאת מהמערכת עם כל אחד מהמשתמשים.  לחיצה על כפתור מיוחד ליציאה. |
| 4 | המערכת תכלול יצירת משתמש ובחירת תפקיד משתמש מורה/תלמיד(שם משתמש, סיסמה, מייל) | דרישה תפעולית | נסיון ליצור משתמש מורה ומשתמש תלמיד. בדיקה שלכל אחד מהמשתמשים יש את ההרשאות שאמורות להיות לו. נסיון ליצור משתמשים מסוג שונה אבל עם אותם פרטים(אמורה לצאת הודעת שגיאה) |
| 5 | המערכת תכלול אפשרות שחזור סיסמה ושם משתמש(שכחתי סיסמה), שליחת מייל לאיפוס | דרישה תפעולית | נסיון לשחזר סיסמה, לחיצה על כפתור מיוחד לשחזור, מילוי פרטים, בדיקה האם המייל הגיעה לכתובת הנכונה והאם ניתן ממנו להגיע לעמוד שחזור, מילוי הפרטים לשחזור ונסיון כניסה למערכת עם הפרטים החדשים.  בדיקה לכל סוגי המשתמשים. |
| 6 | החלפת סיסמה כל 3 חודש. | דרישת מאפייני איכות | כדי לבדוק את הדרישה צריך לשנות את התאריך במחשב לתאריך הרצוי(+3 חודש), לנסות להכנס למערכת ולראות האם קיימת הודעה להחלפת סיסמה. |
| 7 | כל משתמש יכול לעדכן את הפרטים האישיים בפרופיל כולל שינוי סיסמה | דרישה תפעולית | נסיון(עם כל סוג משתמש) לעדכן פרטים חדשים כמו שם פרטי, מייל, סיסמה, כיתה וכו'. לחיצה על אישור ובדיקה האם הפרטים התעדכנו. |
| 8 | המערכת תבדוק ששם המשתמש והמייל לא קיימים כבר במערכת, במקרה וכן תהיה הודעה מתאימה | דרישה תפעולית | נסיון ליצור משתמש חדש עם כתובת מייל שכבר קיימת במערכת.(אמורה לצאת הודעת שגיאה) |
| 9 | למורה יש אפשרות למחוק תלמיד מהכיתה במקרה הצורך | דרישה תפעולית | כניסה לממשק המורה, נסיון מחיקה של אחד התלמידים על ידי לחיצה על הכפתור מחיקה. |
| 10 | משתמש לא פעיל במשך חצי שנה ימחק. | דרישה תפעולית | בשביל הבדיקה צריך לשנות את התאריך המחשב(+6 חודש), כל המשתמשים אמורים להימחק כי אף אחד מהם לא היה פעיל. |
| 11 | המערכת תכלול בממשק המורה אפשרות חלוקה לפי קבוצות(כיתות). | דרישה תפעולית | כניסה לממשק המורה, נסיון ליצור כמה משתמשים(תלמידים) לשייך אותם לכיתות מסויימות. להיכנס לתקייה של כיתה מסויימת ולראות האם החלוקה שרצינו קיימת. |
| 12 | למורה יש אפשרות ליצור כיתה | דרישה תפעולית | נסיון ליצור כיתה על ידי לחיצה על כפתור יצירת כיתה. |
| 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 | נסיון לכתוב תוכן הולם ולראות שהוא משמר ואז לנסות לכתוב תוכן לא הולם ולראות האם מופיעה הודעת שגיאה. |
| 16 | התלמיד יכול לבצע את המשימות שצירפו לו בכיתות שהוא נמצא בהן. | דרישה תפעולית | כניסה לממשק המורה, נסיון ליצור משימה מסויימת ותוך כדי לבחור תבנית ולמלא תוכן הולם, לשמור את המשימה ולראות שהיא נשמרה בהצלחה עם התבנית והתוכן שרצינו. |
| 17 | לכל משימה אצל התלמיד יש אייקון שמתווסף:  V – אם המשימה בוצעה בהצלחה  X- אחרת | דרישת מידע | נסיון לעשות פעם אחת את המשימה בהצלחה ולראות האם אכן מתווסף אייקון מתאים, ולעשות נסיון ולענות לא נכון על המשימה ולראות האם אכן מתווסף אייקון מתאים. |
| 18 | המערכת תכלול רשימת פרסים/תעודות שנקבעו ע"י האדמין, ומקבלים אותם ע"י צבירת נקודות. | דרישת מידע | לענות על כמה משימות בהצלחה, ולראות האם אכן המשתמש(תלמיד) מקבל פרסים ותעודות לפי מה שנקבע בפרטי המשימה, אותה בדיקה לגבי קבלת הניקוד. |
| 19 | התלמיד יכול לראות את התעודות והפרסים שהוא קיבל. | דרישת מידע | לאחר מילוי התשובות הנכונות על המשימה, נכנסים לאזור הפרסים והתעודות ובודקים אם אכן רואים את כל הפרסים וכל התעודות שאמורות להופיע. |
| 20 | המערכת תכלול אצל מורה סטטיסטיקה על כל תלמיד וכיתה. | דרישת מידע | כניסה לממשק המורה, כניסה לאזור סטטיסטיקות ובדיקה האם אכן קיימות סטטיסטיקות על כל תלמיד ועל כל כיתה והן נכונות. |
| 21 | המערכת תכלול אצל כל תלמיד את הסטטיסטיקה שלו. | דרישת מידע | כניסה לממשק התלמיד, כניסה לאזור הסטטיסטיקות, ובדיקה האם אכן קיימת סטטיסטיקה של התלמיד והיא אכן נכונה. |
| 22 | אפשרות הורדה של הסטטיסטיקה(יחיד או קבוצה) בקובץ אקסל בטבלה | דרישת ממשק | נסיון בכל אחד מהממשקים(מורה ותלמיד) להוריד את הסטטיסטיקה לתלמיד מסויים בנפרד ונסיון להוריד סטטיסטיקה לכיתה, לבדוק שאכן נוצר קובץ אקסל והוא תקין. |
| 23 | המערכת תבצע גיבוי וחישוב של הסטטיסטיקות פעם ביום בשעה 12 בלילה | דרישת מאפייני איכות | נסיון ליצור סטטיסטיקה מסויימת לפני השעה 12 ולראות האם אחרי 12 נוצר גיבוי בשרת ונוצרו הסטטיסטיקות והן תקינות. |
| 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 | בדיקה של כל אחד מהמשתמשים, נסיון כניסה עם האדמין לכיתות תיקיות ומשימות ולבדוק שהן אכן קיימות אצלו.  כניסה עם ממשק המורה ולעשות אותה בדיקה גם אצלו. בנוסף לבדוק שבאזור המשימות אכן קיימות כל המשימות שהוא יצר.  כניסה עם ממשק התלמיד ולעשות אותם בדיקות גם אצלו. |
| 25 | אצל כל משתמש תהיה אפשרות כתיבת הודעות בין תלמיד למורה. | דרישה תפעולית | כניסה לכל אחד מהממשקים ולנסות לכתוב הודעה לצד השני.  אצל התלמיד לכתוב למורה ואצל המורה לכתוב לתלמיד ולראות שאכן ההודעות מתקבלות תקינות. |

אלכס אגרונוב, אלסו בוגדנוב

A picture containing logo

Description automatically generated

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 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:\_ \_\_\_\_\_\_\_\_\_\_\_\_