בית הספר הארצי להנדסאים

שם הסטודנט: בוגדנוב אלסו תאריך הגשה:

שם המנחה: יבגניה צ'רנומז מגמה וכיתה: מחשבים

נושא פרויקט הגמר:

**מערכת לניהול ומעקב אחר שיעורי בית של תלמידים**

**150 – 200 מילים, גופן 12**

תארו בקצרה ובצורה ממצא את הפרויקט שיצרתם, מה מטרתו ? מה הוא מכיל ? למי הוא מיועד ? אילו בעיות הוא אמור לפתור ? מה היתרונות שלו ? וכולי'

(התקציר הינו נגזרת של המבוא שתכתבו ) .

הפרויקט שלנו נועד לשפר את הסביבה החינוכית על ידי מתן פתרונות חדשניים לתלמידים ולמורים. מהות הפרויקט מתחלקת לשני מרכיבים חשובים:

ראשית, הפלטפורמה מספקת הזדמנות ייחודית לסטודנטים להתאמן באופן פעיל בנושאים שונים, שנלמדו כבר וחדשים. קשה במיוחד לשמור על תלמידים פעילים בימים אלה, לאור ההשפעה הנרחבת של מכשירים אלקטרוניים. אנו שואפים לפתור בעיה זו על ידי מתן פלטפורמה נוחה לסטודנטים ללמוד בעצמם, לעשות שיעורי בית ולתרגל בנושאים מעניינים.

שנית, הפרויקט שלנו נועד להקל על מורים לעקוב אחר התקדמות התלמידים. בתנאים של כיתות גדולות וזמן מוגבל, חשוב שיהיה כלי יעיל למעקב אחר הפעילות האקדמית של כל תלמיד. הפלטפורמה שלנו מספקת משוב רציף, ומאפשרת למורים לעקוב אחר התקדמות המטלות ולספק הערות מפורטות על עבודת התלמידים בזמן אמת.

הפרויקט נועד להתגבר על אתגרים מודרניים בתחום החינוך, להבטיח חיבור הדוק יותר בין התהליך החינוכי להתפתחות הטכנולוגית. צפוי כי הפלטפורמה המוצעת תשפר את איכות החינוך על ידי הגברת מעורבות התלמידים בתהליך הלמידה, וכן תבטיח אינטראקציה אפקטיבית בין מורים לתלמידים.

יש לשים כאן את הצעת הפרוייקט המקורית המאושרת , (זאת שחזרה ממזכירות– חתומה ) . אסור לשנות דבר בהצעה, מעתיקים אותה בשלמותה!

תוכן עניינים

לפי פרקים ראשיים ומשניים ( מלאו והשלימו את המבנה המוצג לפי מה שכתבתם בספר הפרוייקט שלכם ). מספור דפים מתחיל מתקציר.

מספור עמודים מתחיל בתקציר (1). מספר עמוד של מבוא = תקציר + הצעה + תוכן עניינים.

חובה למחוק שורות מיותרות בתוכן עניינים.

|  |  |
| --- | --- |
| נושא | ע"מ |
| * מבוא |  |
| * נושאים הקשורים לניתוח מערכת :   + דיאגרמת קשר   + Dfd   + מילון נתונים     - תיאור ישויות     - תיאור מאגרים     - תיאור זרימות     - תיאור תהליכים   + מבני נתונים   + חלופות שפת מימוש   + פירוט בדיקות תוכנה ואופן ביצוען   + סיכום + נספחים |  |
| * תיאור אלגוריתמים לפתרון הבעיה |  |
|  |  |
|  |  |
|  |  |
|  |  |

מבוא

מכיל את מטרת העבודה, ורקע כללי לבחירת נושא העבודה(בין ע"מ אחד לשני ע"מ) . אפשר להוסיף הסבר על הטכנולוגיה הנבחרת.

1-1.5 p

במרחב החינוכי המודרני, לצד התפתחות טכנולוגית מהירה, מתעוררים אתגרים חדשים לתלמידים ולמורים. הצורך במערכת יעילה להדרכה, שיתוף ידע ומעקב אחר ההתקדמות הופך להיות דחוף יותר ויותר. בהקשר זה, פרויקט Finnapp שואף לספק פתרונות חדשניים לשני המשתתפים בתהליך החינוכי.

**מטרת הפרויקט**

המטרה העיקרית של פרויקט אפליקציית Finns היא ליצור פלטפורמה המקדמת תרגול פעיל ופיתוח של תלמידים במגוון נושאים. הפלטפורמה גם מספקת למורים את הכלים לנטר ביעילות רבה יותר את תהליך הלמידה ולתמוך בתלמידים בלמידה שלהם.

**בחירת ערכת נושא**

בחירת נושא הפרויקט נקבעה על פי הקשיים הקיימים בתחום החינוכי הקשורים בהיקפי מידע עצומים וקושי במעקב אחר ההתקדמות. בהתחשב בכך, הוחלט לפתח מערכת שתוכל להעניק סיוע פרטני לכל תלמיד ולהקל על המורים לעקוב אחר הפעילות האקדמית.

**טכנולוגיה נבחרה**

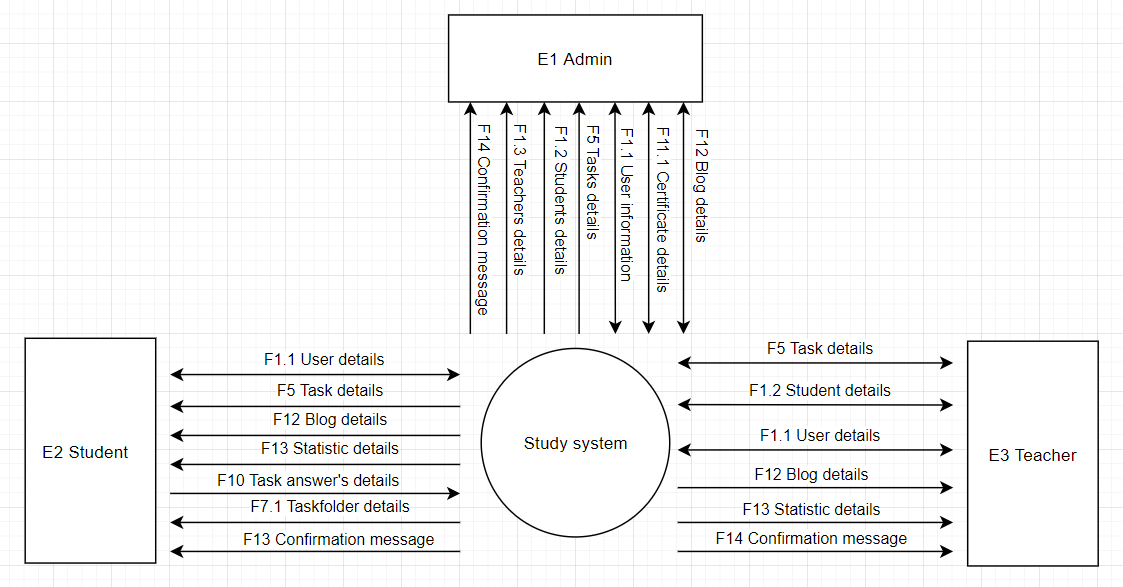
כדי ליישם את הפרויקט, נבחרה התבנית האדריכלית MVC (Model-View-Controller), המאפשרת להפריד בבירור בין ההיגיון, תצוגת הנתונים וניהול האפליקציות. Node.js משמש כקצה האחורי, המספק ביצועים גבוהים ויכולת לעבד בקשות באופן אסינכרוני. React נבחרה ל-frontend, ספרייה רבת עוצמה ליצירת ממשקי משתמש אינטראקטיביים. מסד הנתונים מיושם באמצעות MySQLi, המבטיח אחסון נתונים אמין.

הפרויקט מכסה סוגים שונים של משתמשים: מנהל, מורים ותלמידים. זה מאפשר חוויה מותאמת אישית ומבטיח שלכל קטגוריה של משתמשים נוח ללמוד ולתקשר עם הפלטפורמה. נושאים הקשורים לניתוח מערכת

## טבלת דרישות שהשתנו

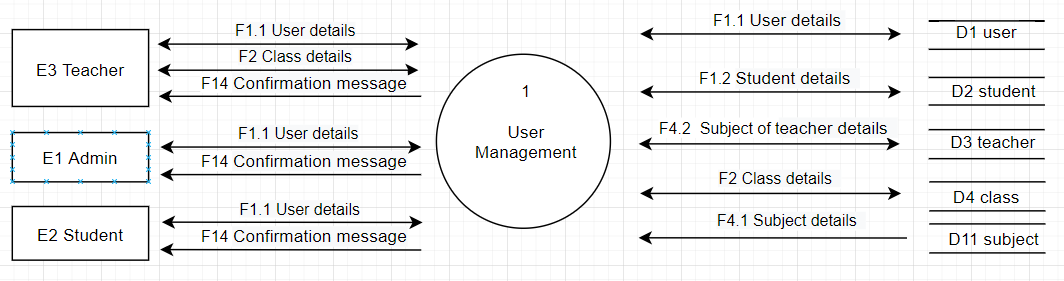
|  |  |  |  |
| --- | --- | --- | --- |
| מספר דרישה מקורית | תיאור דרישה מקורית | סיבה לשינוי  (דרישה חדשה, נוסח השתנה, דרישה בוטלה) | תיאור חדש של דרישה |
|  | המערכת תכלול אפשרות שחזור סיסמה ושם משתמש(שכחתי סיסמה), שליחת מייל לאיפוס | נוסח השתנה  כאשר יש למשתמש מייל אין צורך בשיחזור שם משתמש אלא רק שיחזור סיסמה | המערכת תכלול אפשרות שחזור סיסמה דרך שליחת מייל לאיפוס |
|  | המערכת תבדוק ששם המשתמש והמייל לא קיימים כבר במערכת, במקרה וכן תהיה הודעה מתאימה | נוסח השתנה  שם משתמש יכול לחזור על עצמו ורק מייל צריך להיות יחודי | המערכת בודקת שמייל לא קיים כבר במערכת, במקרה וכן תהיה הודעה מתאימה |
|  | משתמש לא פעיל במשך חצי שנה ימחק. | דרישה בוטלה  כמות דרישות קטנה בגלל שהפרוייקט מוגש על ידי סטודנט יחיד ולא עם סטודנט נוסף שהיה מוגדר בהתחלה |  |
|  | אפשרות הורדה של הסטטיסטיקה(יחיד או קבוצה) בקובץ אקסל בטבלה | דרישה בוטלה  כמות דרישות קטנה בגלל שהפרוייקט מוגש על ידי סטודנט יחיד ולא עם סטודנט נוסף שהיה מוגדר בהתחלה |  |
|  | המערכת תבצע גיבוי וחישוב של הסטטיסטיקות פעם ביום בשעה 12 בלילה | דרישה בוטלה  כמות דרישות קטנה בגלל שהפרוייקט מוגש על ידי סטודנט יחיד ולא עם סטודנט נוסף שהיה מוגדר בהתחלה |  |
| 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. | הבהרת הדרישה | לכל המשתמשים יש אפשרות צפייה בכיתות, תיקיות ומשימות לפי הרשאות:  **אדמין:** יכול לראות את כל הכיתות ,פרטים של תלמידים ומורים ותוכן של המשימות  **מורה:** יכול לראות ולתקן רק את הכיתות, תיקיות, קטגוריות, והמשימות שהוא יצר  **תלמיד**: יכול לראות את התיקיות והמשימות שהוא מצורף אליהם. בנוסף למשימות הקיימות שהוטלו על ידי המורה, הוא יכול לבצע משימות במקצועות שונים, ברמות קושי שונות מהמאגר הכללי של המשימות  **כל משתמש** יכול לראות ולבצע את המשימות הכלליות |
| 11 | המערכת תכלול בממשק המורה אפשרות חלוקה לפי קבוצות(כיתות). | הבהרת הדרישה | המערכת תכלול בממשק המורה אפשרות חלוקה לפי קבוצות(כיתות) ומקצועות |
| 14 | המורה יוצר את המשימות ע"י בחירת תבנית למשימה ומילוי תוכן. | הבהרת הדרישה | המורה יוצר את המשימות ע"י בחירת תבנית למשימה, מילוי תוכן והוספת תכונות של המשימה: משקל, קטגוריה, רמה ותיאור משימה. |
| 9 | למורה יש אפשרות למחוק תלמיד מהכיתה במקרה הצורך | הבהרת הדרישה | למורה יש אפשרות להוסיף תלמיד לכיתה דרך הזנה מייל של תלמיד או הוספת קובץ עם מיילים של תלמידים. ישנה אפשרות למחוק תלמיד מהכיתה במקרה הצורך |
|  | לכל המשתמשים יש גישה למדור הבלוג, שהמאמרים שלו מנוהלים על ידי אדמין | דרישה חדשה |  |

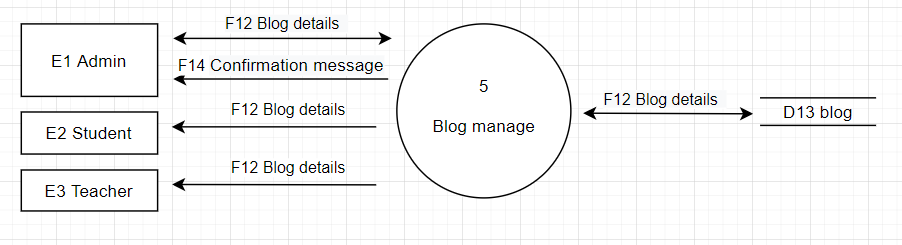
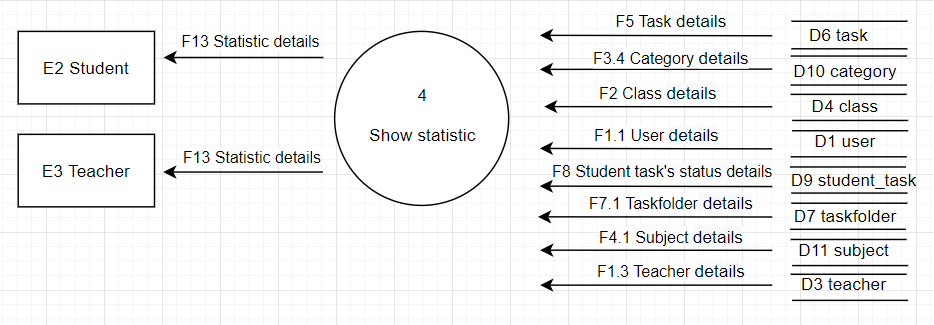
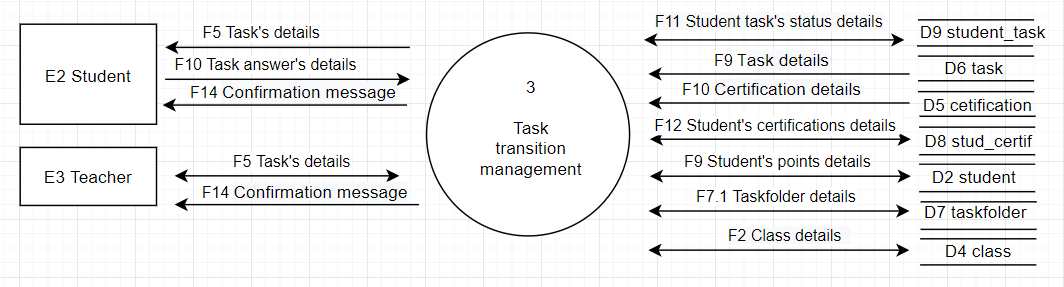
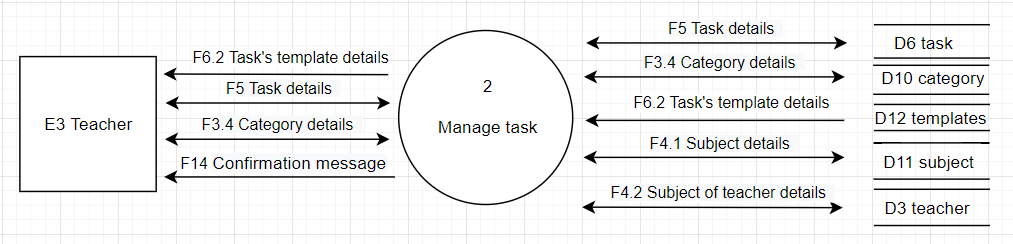
## דיאגרמת הקשר (Context Diagram).



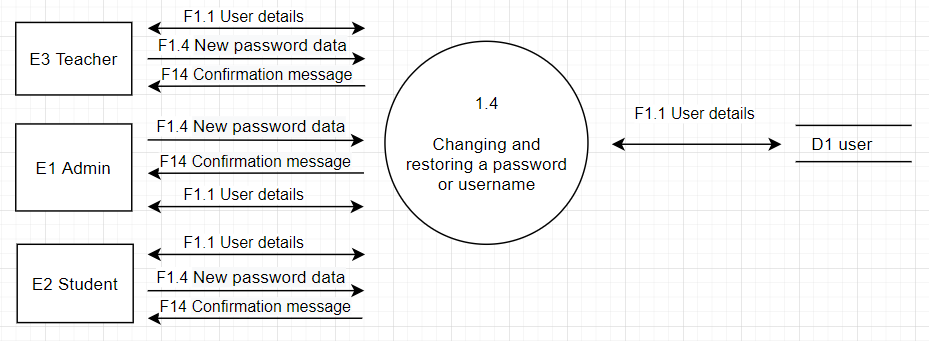
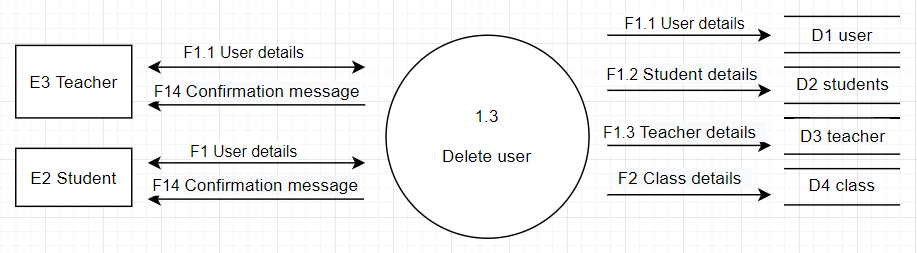
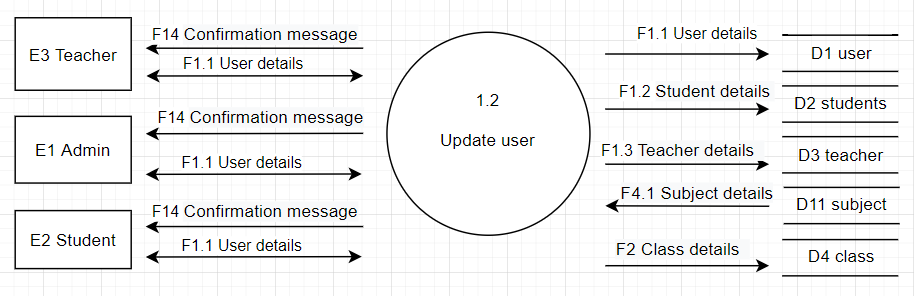
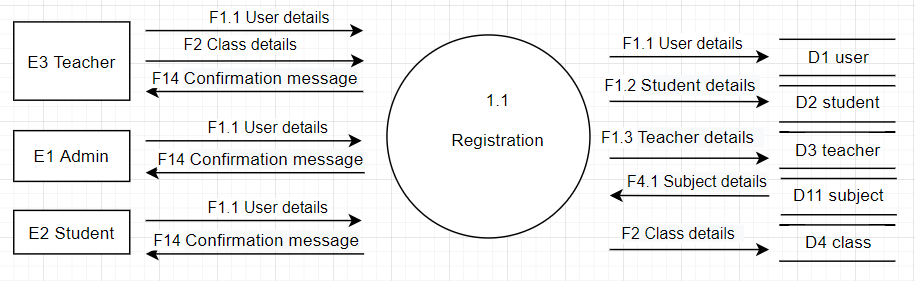
## ניתוח התהליכים העיקריים.

**DFD0**

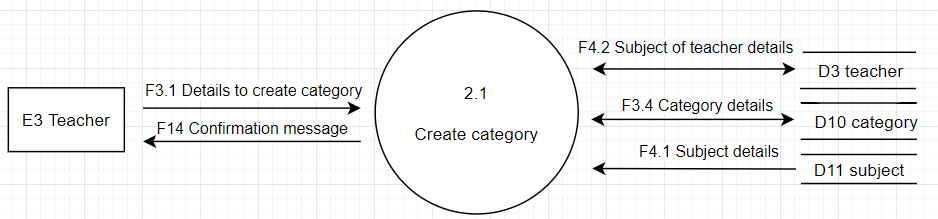


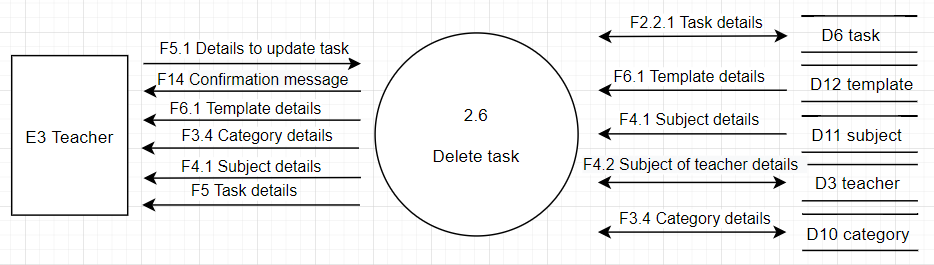
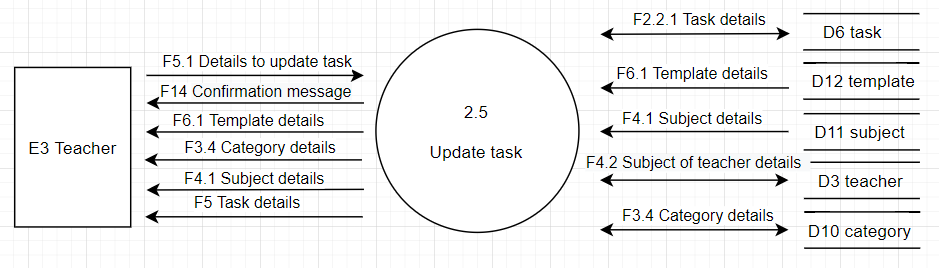
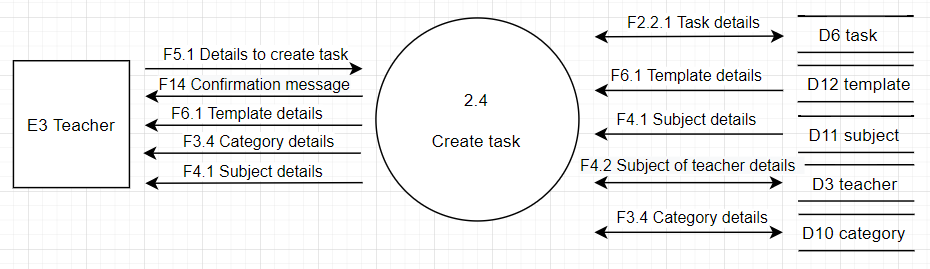
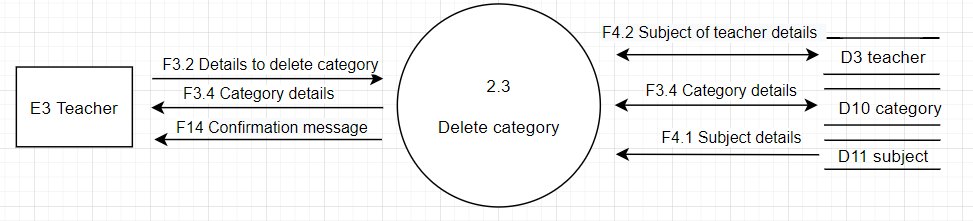
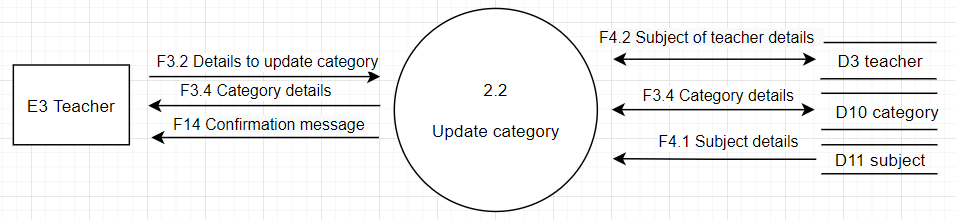


**DFD1 – User Management**

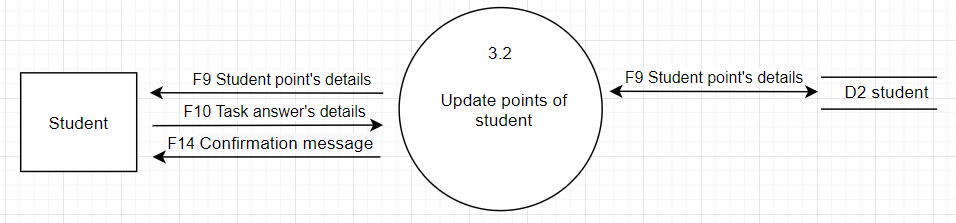
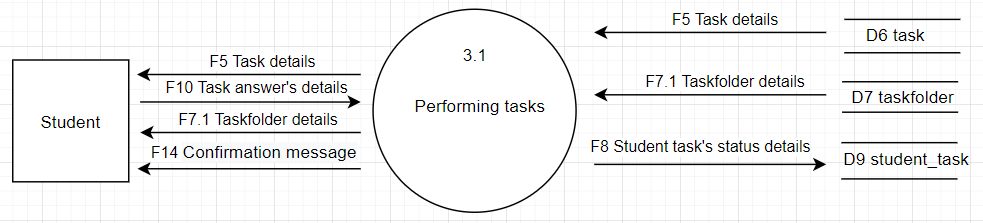


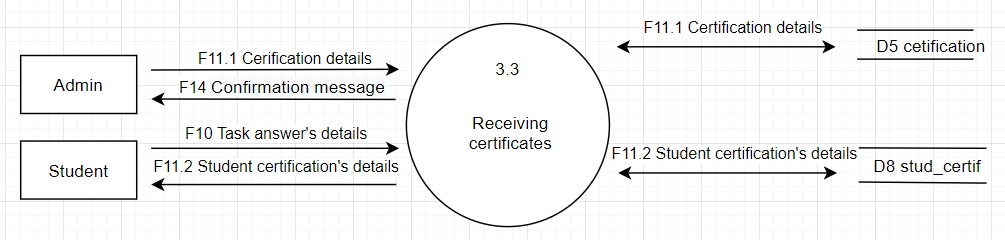
**DFD1 – Manage tasks**

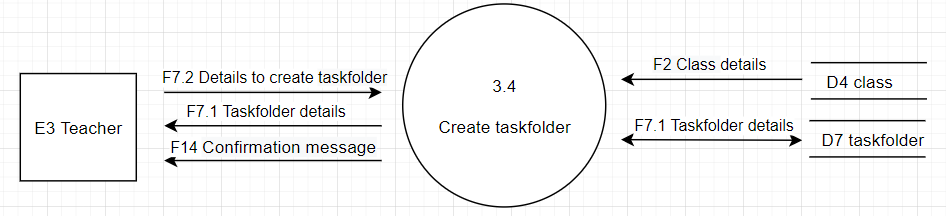


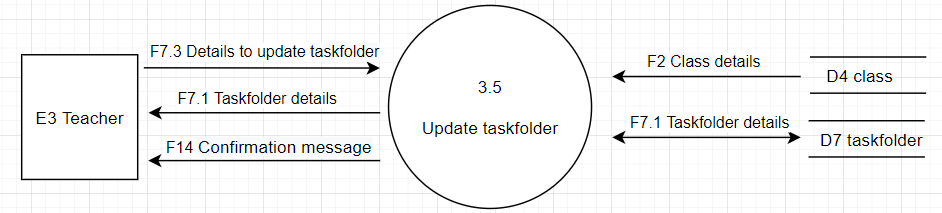


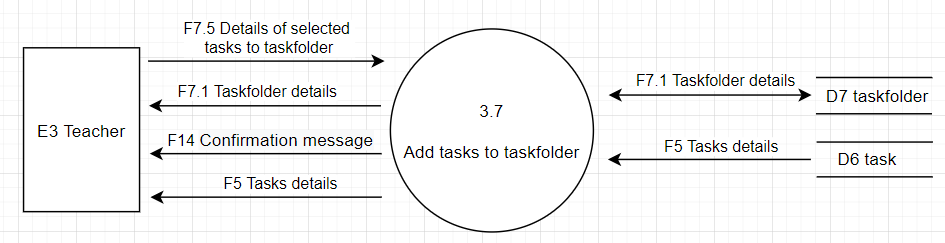
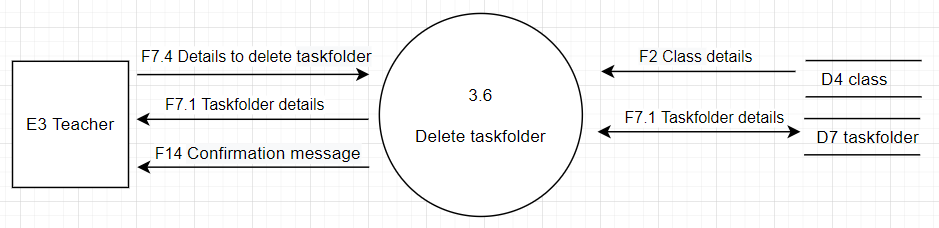
**DFD1 – Task transition management**



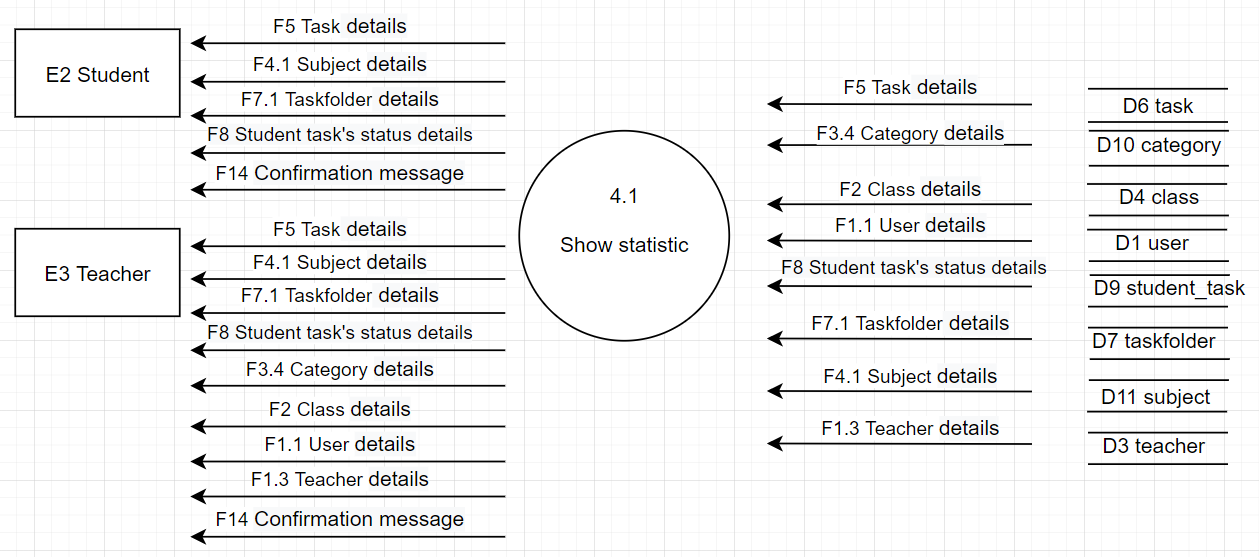




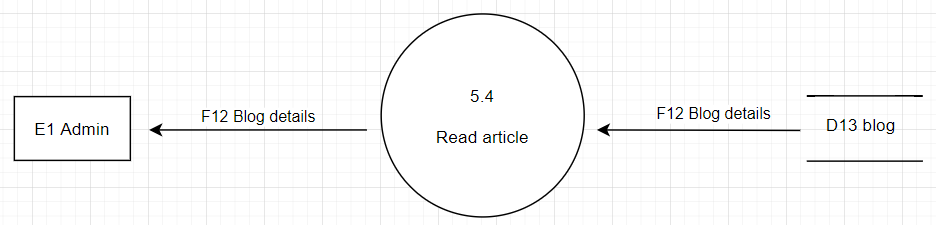
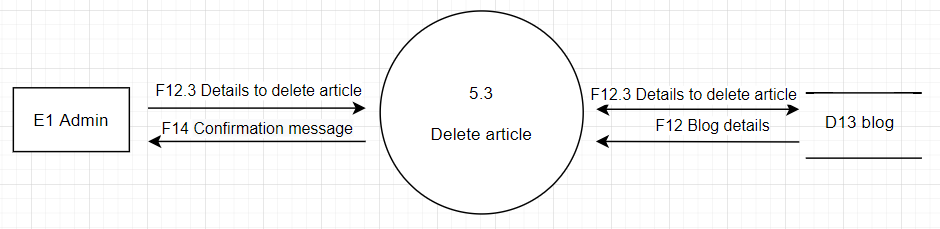
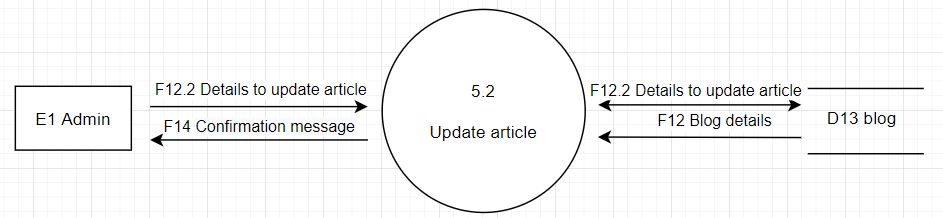
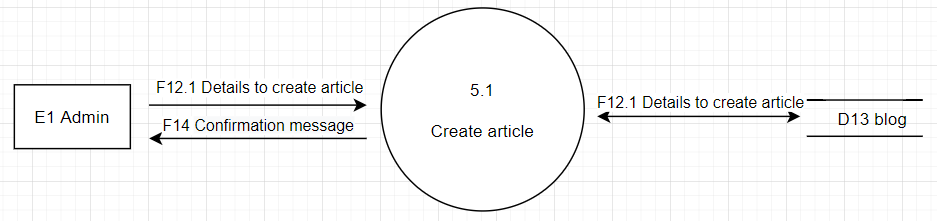




**DFD1 – Statitic**



**DFD1 – Blog management**



# **מילון נתונים**

## תאור ישויות.

|  |  |  |
| --- | --- | --- |
| **קוד** | **שם הישות** | **תיאור** |
| E1 | Admin | אחראי על יצירת משימות ורמות |
| E2 | Student | אדם המשתמש במערכת |
| E3 | Teacher | אדם המשתמש במערכת |

## תאור המאגרים.

|  |  |  |
| --- | --- | --- |
| **קוד + שם מאגר** | שם טבלה | **שדות הטבלה** |
| D1 user | user | id\_user, role, email, password, name, lastName |
| D2 student | student | id\_user ,class\_level, total\_points |
| D3 teacher | teacher\_subject | id\_user,**id\_subject** |
| D4 class | class | id\_class, class\_name, class\_level, **id\_teacher** |
| student\_class | id\_user, id\_class |
| D5 certification | certification | id\_certif, name\_certif, point, img\_url |
| D6 task | task | id\_task , task\_name, task\_text, task\_create\_date, task\_weight, task\_level, task\_img, specific\_data, **id\_teacher**, **id\_category, id\_template** |
| task\_taskfolder | id\_task, id\_tskFolder |
| D7 taskfolder | taskfolder | id\_tskFolder, tskFolder\_name, is\_publish, **id\_class, id\_subject** |
| D8 stud\_certif | student\_certification | id\_user, id\_certif, data\_get |
| D9 student\_task | student\_task | id\_user, id\_task, is\_task\_done, date |
| D10 category | category | id\_category, category\_name, date\_create, **id\_subject** |
| D11 subject | subject | id\_subject, subject\_name, subject\_icon |
| D12 template | template | id\_template, template\_name, template\_img |
| D13 blog | blog | id\_article, art\_title, art\_text, art\_img, create\_date |

## תאור זרימות.

|  |  |  |  |
| --- | --- | --- | --- |
| **קוד + שם זרימה** | **מקור** | **יעד** | **שדות הזרימה** |
| F1.1User details | E1 Admin, E2 Student, E3 Teacher | D1 user | id\_user, role, email, password, name, lastName |
| F1.2 Student details | E2 Student | D2 student | id\_user ,class\_level |
| F1.3 Teacher details | E3 Teacher | D3 teacher | id\_user, id\_subject |
| F1.4 New password data | E1 Admin, E2 Student, E3 Teacher | D1 user | id\_user, email, password |
| F2 Class details | E3 Teacher | D4 class | id\_class, class\_name, class\_level, id\_teacher |
| F3.1 Details to create category | E3 Teacher | D10 category | id\_category, category\_name, date\_create, id\_subject |
| F3.2 Details to update category | E3 Teacher | D10 category | id\_category, category\_name, date\_create, id\_subject |
| F3.3 Details to delete category | E3 Teacher | D10 category | id\_category |
| F3.4 Category details | E3 Teacher | D10 category | id\_category, category\_name, date\_create, id\_subject |
| F4.1 Subject details | D11 subject | E3 Teacher | id\_subject, subject\_name, subject\_icon |
| F4.2 Subject of teacher details | E3 Teacher | D3 teacher | id\_user, id\_subject |
| F5 Task details | E3 Teacher | D6 task | id\_task , task\_name, task\_text, task\_create\_date, task\_weight, task\_level, task\_img, specific\_data, id\_teacher, id\_category, id\_template |
| F5.1 Details to create task | E3 Teacher | D6 task | id\_task , task\_name, task\_text, task\_create\_date, task\_weight, task\_level, task\_img, specific\_data, id\_teacher, id\_category, id\_template |
| F5.2 Details to update task | E3 Teacher | D6 task | id\_task , task\_name, task\_text, task\_create\_date, task\_weight, task\_level, task\_img, specific\_data, id\_teacher, id\_category, id\_template |
| F5.3 Details to delete task | E3 Teacher | D6 task | id\_task , id\_teacher |
| F6.1 Template details | D12 template | E3 Teacher | id\_template, template\_name, template\_img |
| F6.2 Task’s template details | E3 Teacher | D12 template | Id\_task, id\_template |
| F7.1 Taskfolder details | E3 Teacher, E2 Student | D6 taskfolder | id\_tskFolder, tskFolder\_name, is\_publish, id\_class, id\_subject |
| F7.2 Details to create taskfolder | E3 Teacher | D7 taskfolder | id\_tskFolder, tskFolder\_name, is\_publish, id\_class, id\_subject |
| F7.3 Details to update taskfolder | E3 Teacher | D7 taskfolder | id\_tskFolder, tskFolder\_name, is\_publish, id\_class, id\_subject |
| F7.4 Details to delete taskfolder | E3 Teacher | D7 taskfolder | id\_tskFolder, id\_class |
| F7.5 Details of selected tasks to taskfolder | E3 Teacher | D7 taskfolder | id\_task, id\_tskFolder |
| F8 Student task’s status details | E2 Student | D9 student\_task | id\_user, id\_task, is\_task\_done, date |
| F9 Student point’s details | E2 Student | D2 student | id\_user , total\_points |
| F10 Task answer’s details | E2 Student | P3.2 Update points of student | id\_task , task\_name, task\_text, task\_create\_date, task\_weight, task\_level, task\_img, specific\_data, id\_teacher, id\_category, id\_template, id\_user |
| F11.1 Certification details | E1 Admin | D5 certification | id\_certif, name\_certif, point, img\_url |
| F11.2 Student certification’s details | 3.3 Reciving certificate | D8 stud\_certif | id\_user, id\_certif, data\_get |
| F12 Blog details | E1 Admin | D13 blog | id\_article, art\_title, art\_text, art\_img |
| F12.1 Details to create article | E1 Admin | D13 blog | id\_article, art\_title, art\_text, art\_img |
| F12.2 Details to update article | E1 Admin | D13 blog | id\_article, art\_title, art\_text, art\_img |
| F12.3 Details to delete article | E1 Admin | D13 blog | id\_article, art\_title, art\_text, art\_img |
| F12.4 Details to read article | E1 Admin, E2 Student, E3 Teacher | D13 blog | id\_article, art\_title, art\_text, art\_img |
| F13 Statistic details | P4 Show statistic | E2 Student, E3 Teacher | id\_task , task\_name, task\_create\_date, task\_img, id\_category, id\_user, id\_category, category\_name, id\_subject, id\_class, class\_name, id\_tskFolder, tskFolder\_name |
| F14 Confirmation message | Study system | E1 Admin, E2 Student, E3 Teacher |  |

2.4. תאור תהליכים.

|  |  |  |
| --- | --- | --- |
| **קוד** | **שם התהליך** | **תיאור** |
| P1.1 | Registration | * מאפשר רישום משתמש |
| P1.2 | Update user | * מאפשר עדכון משתמש |
| P1.3 | Delete user | * מאפשר מחיקת משתמש |
| P1.4 | Changing and restoring a password | * מאפשר לשנות סיסמה * מאפשר שיחזור סיסמה דרך דוא''ל |
| P2.1 | Create category | * מאפשר למורה ליצור קטגוריה |
| P2.2 | Update category | * מאפשר למורה עדכון קטגוריה |
| P2.3 | Delete category | * מאפשר למורה מחיקת קטגוריה |
| P2.4 | Create task | * מאפשר למורה ליצור משימה |
| P2.5 | Update task | * מאפשר למורה עדכון משימה בתנאי שלא נמצאת בתיקייה ציבורית |
| P2.6 | Delete task | * מאפשר למורה מחיקת משימה בתנאי שלא נמצאת בתיקייה ציבורית |
| P3.1 | Performing tasks | * מאפשר לתלמיד ביצוע משימה |
| P3.2 | Update points of student | * מאפשר עדכון נקודות של התלמיד בגין תשובות נכונות |
| P3.3 | Receiving certificates | * מאפשר לתלמיד קבלת תעודת הערכה לפי צבירת נקודות |
| P3.4 | Create taskfolder | * מאפשר למורה ליצור תיקיית משימות לכל כיתה בנפרד |
| P3.5 | Update taskfolder | * מאפשר למורה עדכון תיקיית משימות בתנאי שלא העבירה מצב התיקייה לציבורי |
| P3.6 | Delete taskfolder | * מאפשר למורה מחיקת תיקיית משימות בתנאי שהמשימה לא התבצעה על ידי אף אחד מהתלמידים |
| P3.7 | Add tasks to taskfolder | * מאפשר למורה להוסיף משימות לתיקיה בתנאי שהיא אינה ציבורית |
| P4.1 | Show statistic | * מאפשר למורה לצפות סטטיסטיקה של ביצוע משימות לפי כיתה * מאפשר למורה לצפות סטטיסטיקה כמה תלמידים ביצעו משימה מסויימת בכל כיתה בניפרד * מאפשר לתלמיד לצפות סטטיסטיקת ביצוע משימות לפי מקצועות |
| P5.1 | Read articles | * מאפשר לכל המשתמשים קריאת מאמרים מהבלוג להעשרת ידע |
| P5.2 | Create article | * מאפשר לאדמין יצירת מאמר |
| P5.3 | Update article | * מאפשר לאדמין עדכון מאמר |
| P5.4 | Delete article | * מאפשר לאדמין מחיקת מאמר |

# **מבנההנתונים**

## תרשימי ERD של מבנה הנתונים עבור בסיס נתונים רלציוני.

4. **חלופות שפת מימוש-**

במסגרת ספר הפרויקט חשוב להציג בחינה של מספר חלופות עבור שפה\ות מימוש

הפרויקט. הנ"ל צריך לכלול דרישות אותן יגדיר התלמיד בבחירת השפה המתאימה. בין יתר השיקולים ניתן לכלול:

* זמני ריצה
* היבטי אבטחה והגנה,
* הגנה על זכויות יוצרים,
* בינארי או interpreter,
* קלות במימוש,
* התאמה לממשקי משתמש או צד שרת וכו.'

בחרתי לכתוב את הממשק משתמש בעזרתספרייתReactואת צד השרת באמצעות פלטפורמה NodeJs.

Node.js הוא זמן ריצה מקורי של Javascript שנבנה על מנוע V8 Javascript של Chrome. הוא משתמש במודל קלט/פלט שאינו חוסם מונע אירועים והוא בנוי להרחבה וביצועים גבוהים.

React.js היא ספריית Javascript חיצונית חינמית, קוד פתוח, הפועלת בדפדפן ומשמשת לבניית ממשקי משתמש. הוא פותח ומתוחזק על ידי פייסבוק.

Node js היא פלטפורמה נוחה לפיתוח שרתי אינטרנט חזקים עבור יישומי React.Node js עובד לצד NPM(Node Package Manager), שהוא מאגר מקוון לפרסום פרויקטי Node.js בקוד פתוח שניתן להתקין בקלות באמצעות ממשק שורת פקודה.

Node מאגדת אפליקציית React לקובץ בודד להידור קל באמצעות מודולי צומת כמו Webpack. מנוע ה-V8 ב-Node js יכול לטפל בבקשות בכמות גדולה באמצעות פונקציות התקשרות חוזרת (V8 הוא מנוע Javascript בתוך Node.js שמנתח ומבצע את ה-Javascript).

Node js ו-React js שניהם משתמשים בשפת Javascript שניתן להפעיל הן בצד השרת והן בצד הלקוח ולכן ניתן להפעיל קוד React.js ישירות בסביבת Node.js.

בנוסף, רכיבי React DOM תוכננו במיוחד לעבוד עם Node js, מה שמפחית שורות קוד ומפשט את העיבוד בצד השרת.

חוץ מזה, שיתוף ספריות אלה מספק את היתרונות הבאים:

1. עוזר לשמור על עומס שרת גבוה

הוספת Node בפרויקט React מקלה בהרבה על שרתים לטפל בבקשות רשת מרובות ולנהל איזון עומסים בשרת.

1. עוזר לשמור על נתונים בזמן אמת

אם האפליקציה דורשת ניהול נתונים בזמן אמת או הזרמת נתונים, השילוב של React ו-Node js יהיה שימושי ביותר. הסיבה לכך היא ש-Node הוא הטוב ביותר לניהול חיבורים מתמשכים לשרת, בעוד React הוא הטוב ביותר לעיבוד נתונים חיצוניים.

1. עוזר בפיתוח יישומי עמוד בודד

באמצעות React ו-Node js, אפשר בקלות לבנות יישומי עמוד בודד. ו-React מאפשר ליצור UI/UX אינטואיטיבי, בעוד ש-Node מספק טעינת נתונים אסינכרוני באמצעות התקשרות חוזרת.

שתי הטכנולוגיות המתוארות כאן משלימות זו את זו היטב. על ידי שילובם, אנו מקבלים שילוב רב עוצמה של כלים לפיתוח אתרים אמינים.

5. **פירוט בדיקות תוכנה ואופן ביצוען -**

כולל את רשימת בדיקות התוכנה ,בדיקות יחידה, בדיקות תהליכיות- full Flow במסגרת מסמך תכנון בדיקות ובדיקות.

הנ"ל יוצג בטבלה ( השלימו והגדילו אותה לפי הצורך ):

|  |  |  |
| --- | --- | --- |
| תיאור הבדיקה | תוצאה רצויה | תוצאה מתקבלת |
| בדיקה התחברות למערכת עם פרטים שגויים | הודעת שגיאה למשתמש. | הודעת שגיאה למשתמש שאחד הפרטים לא נכון |
| בדיקה שחזור סיסמא בעזרת כתובת מייל שלא קיימת במערכת | הודעת שגיאה למשתמש על כך שהמייל שהוזן אינו רשום במערכת | הודעת שגיאה למשתמש על כך שהמייל שהוזן אינו רשום במערכת |
| בדיקה שאי אפשר להוסיף משתמש אםהמייל שהוכנס קיים כבר | המשתמש לא יוכללהירשם עד שיהיהלו מייל ייחודי | משתמש לא נרשםאלא אם יש לו מיילייחודי |
| בדיקה שכל משתמש יוכל לצפות ולעדכן פרטים אישיים בלבד | בעת לחיצה על הכפתור של עדכון נתונים,לסטודנט מופיעים שדות עם נותנים האישיים שלו,שאותם יוכל לעדכן | בעת לחיצה על הכפתור של עדכון נתונים,המשתמש מקבל רק את הנתונים האישיים שלו. |
| בדיקה שחייב להגדיר בעת הזנת משתמש חדש את הפרטים הבאים:   * סוג/תפקיד משתמש (מורה או תלמיד) * שם פרטי * שם משפחה * מייל * סיסמה   בנוסף, למשתמש מורה צריך לבחור מקצועות שהוא מלמד ולמשתמש תלמיד – רמת כיתה שהוא לומד כרגע | כאשר משתמש מסיים להאזין את הנתונים ולוחץ על כפתור "הרשמה", על המערכת לוודא שכל התונים אכן הוכנסו בצורה נכונה. | המערכת בודקת תקינות הנתונים שהוכנסו,במידה ויש נתונים שחסרים או לא תקינים,המערכת תבקש מהמשתמש להוסיף/לתקן את אותם הנתונים, ורק אז לנסות לשלוח שוב. |
| בדיקה שלא ניתן להוסיף תלמיד לכיתה בה הוא כבר קיימת, גם בהחנסת דרך קובץ של מיילים של תלמידים | הודעת שגיאה למשתמש על כך שתלמיד כבר קיים בכיתה זו | הודעת שגיאה למשתמש על כך שתלמיד כבר קיים בכיתה זו |
| בדיקה שאין אפשרות להוסיף תלמיד לכיתה שאינה רשומה במערכת”Funny app ”, גם בהחנסת דרך קובץ של מיילים של תלמידים | הודעת שגיאה למשתמש על כך שתלמיד לא קיים במערכת | הודעת שגיאה למשתמש על כך שתלמיד לא קיים במערכת |
| בדיקה שאין אפשרות להוסיף תלמיד לכיתה עם מייל שגוי, גם בהחנסת דרך קובץ של מיילים של תלמידים | הודעת שגיאה למשתמש על כך שמייל שלו שגוי | הודעת שגיאה למשתמש על כך שמייל שלו שגוי |
| בדיקה שאין אפשרות למחוק תיקיה עם משימות עם הסטטוס שלה "ציבורי" | הודעת שגיאה למשתמש על כך שאי-אפשר למחוק תיקיה שכבר הועברה ל"ציבורי" | הודעת שגיאה למשתמש על כך שאי-אפשר למחוק תיקיה שכבר הועברה ל"ציבורי" |
| בדיקה שאין אפשרות לשנות את התוכן של משימה שכבר נמצאת בתוך תיקיה עם הסטטוס"ציבורי" | הודעת שגיאה למשתמש על כך שאי-אפשר לשנות את התוכן של משימה שכבר נמצאת בתוך תיקיה עם הסטטוס"ציבורי" | הודעת שגיאה למשתמש על כך שאי-אפשר לשנות את התוכן של משימה שכבר נמצאת בתוך תיקיה עם הסטטוס"ציבורי" |
| בדיקה שתיקיות של משימות עם הסטטוס "לא פורסם" אינן מוצגות בסטטיסטיקה | לא מוצגות תיקיות של משימות עם סטטוס "לא פורסם" | לא מוצגות תיקיות של משימות עם סטטוס "לא פורסם" |
| בדיקה שאין אפשרות למחוק את הקטגוריה אם יש משימה מקטגוריה זו | הודעת שגיאה למשתמש על כך שאי-אפשר למחוק את הקטגוריה אם יש משימה מקטגוריה זו | הודעת שגיאה למשתמש על כך שאי-אפשר למחוק את הקטגוריה אם יש משימה מקטגוריה זו |
| בדיקה שבחשבון התלמיד, משימות בקטע "שיעורי בית" ייעלמו לאחר השלמתן בהצלחה | לא מוצגות משימות אחרי השלמתן בהצלחה | לא מוצגות משימות אחרי השלמתן בהצלחה |

תיאור אלגוריתמים לפתרון הבעיה

אם נעשה שימוש בשילוב של שפות תכנות שונות – יש להסביר כאן

אם נעשה שימוש בתבניות עיצוב Design Patterns)) יש להסביר כאן (כמו MVC, Singleton, Adapter)

כל אלגוריתם לא סטנדרטי או רעיון יעיל מומלץ להביא כאן ולהציג יתרונותיו, כולל חישוב סיבוכיות זמן ו/או מקום.

צריך לתאר 3 -4 אלגוריתמים מורכבים. אפשר למקם Activity Diagrams או שאילתות מורכבות ( עם תת-שאילתות – לדוגמה) עם הסבר.

Перевод слишком длинный

המערכת “Funny app” כוללת תבנית **MVC** (Model-View-Controller). להלן הסבר קצר על האופן שבו רכיבי הפרויקט ממפים לתבנית ה-MVC:

**מודל**: מייצג את הנתונים והלוגיקה העסקית של האפליקציה. בפרויקט, מודלים כוללים טבלאות מסד נתונים (למשל Student\_task, Taskfolder) וכל היגיון משויך לאחזור, עדכון ועיבוד נתונים.

**תצוגה**: התצוגה אחראית להצגת נתונים למשתמש וטיפול באינטראקציה של המשתמש. כלומר, רכיבי ממשק המשתמש שאיתם מנהל המערכת, התלמידים והמורים מקיימים אינטראקציה עם האפליקציה, בוחרים כיתות, מציגים משימות ורואים סטטוסי משימות.

**בקר**: בקרים מנהלים את זרימת הנתונים בין המודל לתצוגה. בפרויקט, רכיבי בקר מנהלים את ההיגיון לקבלת נתונים ממודלים (לדוגמה, קבלת מטלות תלמידים, עדכון סטטוס משימות) והעברת נתונים אלה לתצוגות המתאימות לעיבוד.

Singletone - השתמשנו בתבנית זו למודל שבו אנו מבצעים חיבור לבסיס נתונים. הסיבה לכך שהשתמשנו ב-singlestone היא כיוון שבעת ביצוע שאילתות שונות,אין לנו צורך ביצירת חיבור לבסיס נתונים כל פעם מחדש, בעת קריאה לכל שאילתה.זאת אומרת,שאם ישנו חיבור קיים ,אנחנו נשתמש בו,על מנת לבצע שאילתה.

אלגוריתם ב-Node:

import mysql from 'mysql';

class DatabaseSingleton {

  constructor() {

    this.connection = mysql.createConnection({

      host: 'localhost',

      user: 'root',

      password: '',

      database: 'test',

    });

  }

  query(sql, values, callback) {

    return this.connection.query(sql, values, callback);

  }

  static getInstance() {

    if (!this.instance) {

      this.instance = new DatabaseSingleton();

    }

    return this.instance;

  }

}

export const dbSingleton = DatabaseSingleton;

**אלגוריתם "יצירת משימה" על ידי המורה**

אלגוריתם זה מתאר יצירת משימה עבור תלמיד על ידי מורה. ראוי לציין שתהליך יצירת המשימה עצמה מורכב משני חלקים. 1. מילוי הנתונים המשותפים לכל משימה (נושא, קטגוריה, שם, תיאור, תמונה, רמה, משקל) 2. מילוי נתונים בהתאם לתבנית המשימה שנבחרה. בשלב זה, מרכיב התבנית מחובר, בהתאם לבחירת המורה.

A diagram of a diagram

Description automatically generated

**אלגוריתם " לראות סטטיסטיקה"** **על ידי המורה**

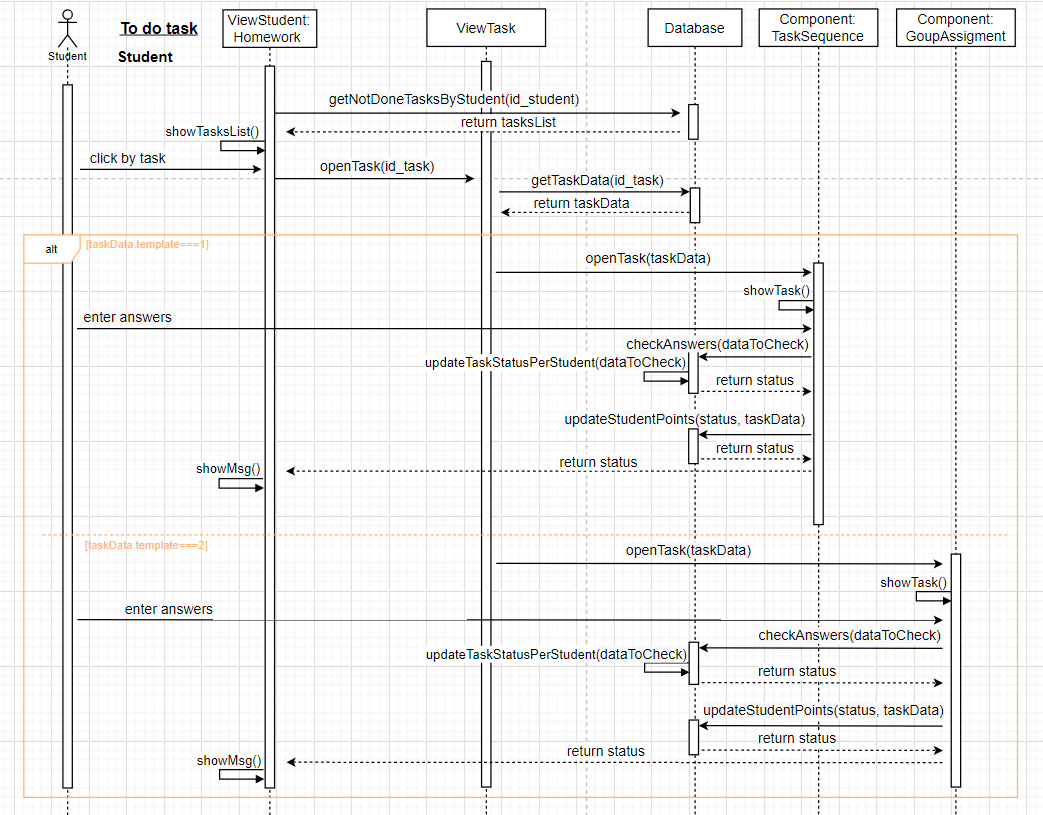
האלגוריתם הבא מתאר כיצד המורה משיג נתונים סטטיסטיים. שני סוגים של סטטיסטיקה זמינים עבור המורה: 1. לפי כיתות. כאן המורה מקבל נתונים על כל המשימות שניתנו לכיתה, רואה גם את מצב המשימה לכל תלמיד 2. במשימה. ראשית עליך לבחור את הנושא והקטגוריה של המשימה. כאן המורה רואה כיצד כל כיתה השלימה את המשימה. בנוסף, ניתן לראות סטטיסטיקות מפורטות יותר עבור כל כיתה, כלומר התלמידים שסיימו אותה

A screen shot of a computer

Description automatically generated

**אלגוריתם "לבצע משימה" על ידי התלמיד**

להלן אלגוריתם לתלמיד לבצע את המשימה שהוטלה עליו על ידי המורה. באלגוריתם זה, בהתאם לסוג המשימה, מחוברת תבנית משימה מסוימת (TaskSequence, GoupAssigment), אשר מתקשרת עם בסיס הנתונים כדי לבדוק את נכונות ביצוע המשימה.

תיאור פתרונות טכנולוגיים

פרק רשות אם יש שימוש בחומרה נוספת, כולל סכמות חיבור והתקנה

מדריך למשתמש בתוכנה

כולל צילומי מסכים והסברים עליהם – המטרה להציג את המערכת שלכם בצורה הטובה ביותר ולחסוך את אלמנט ההפתעה בהגנה על הפרוייקט .

עמוד הכניסה הוא העמוד הראשון שיופיע למשתמש עם הכניסה הראשונית לאתר.

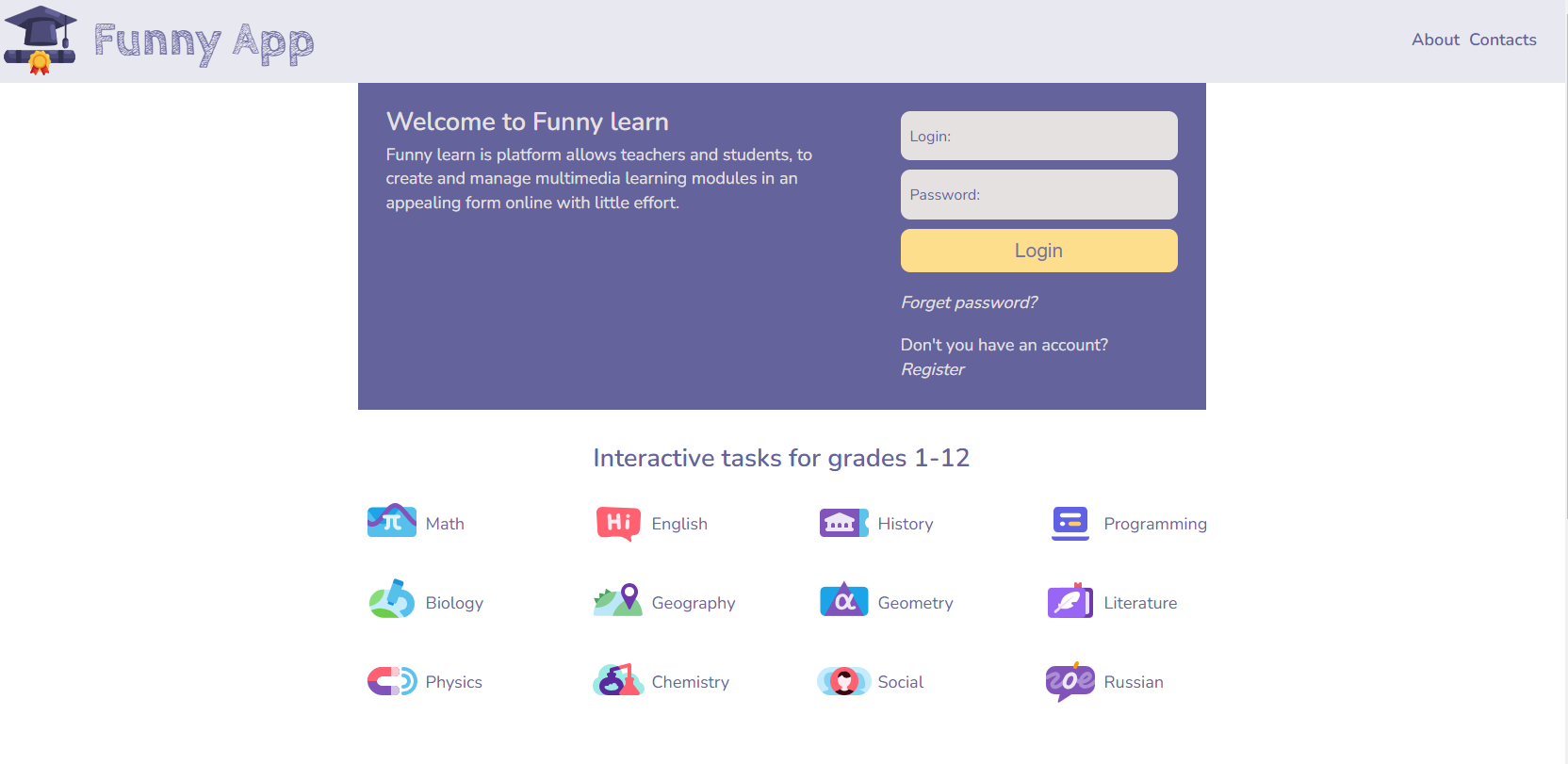
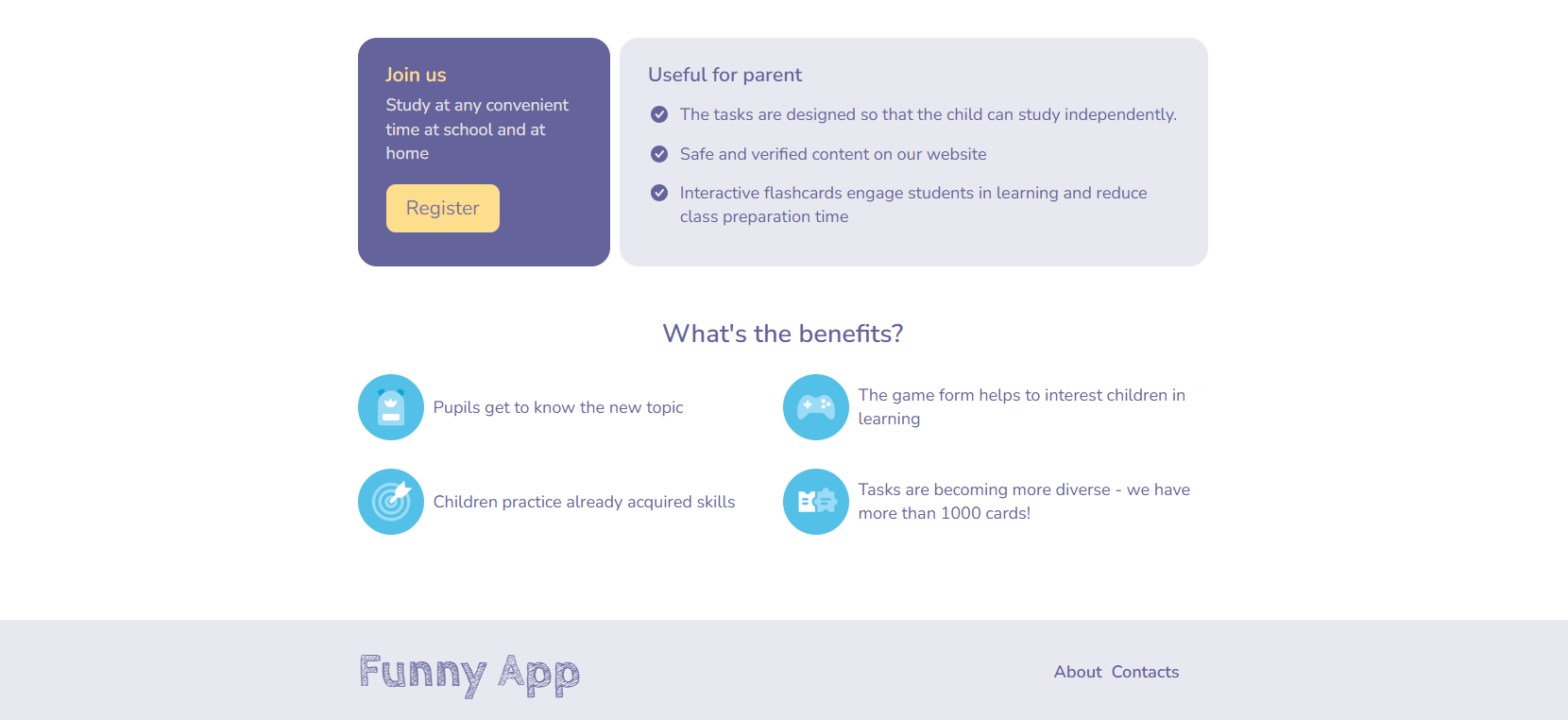
למשתמש יש מספר אפשרויות:

1. היכנס למערכת על ידי הזנת שם משתמש וסיסמה

2. השתמש באפשרות שחזור סיסמה

3. להירשם למערכת

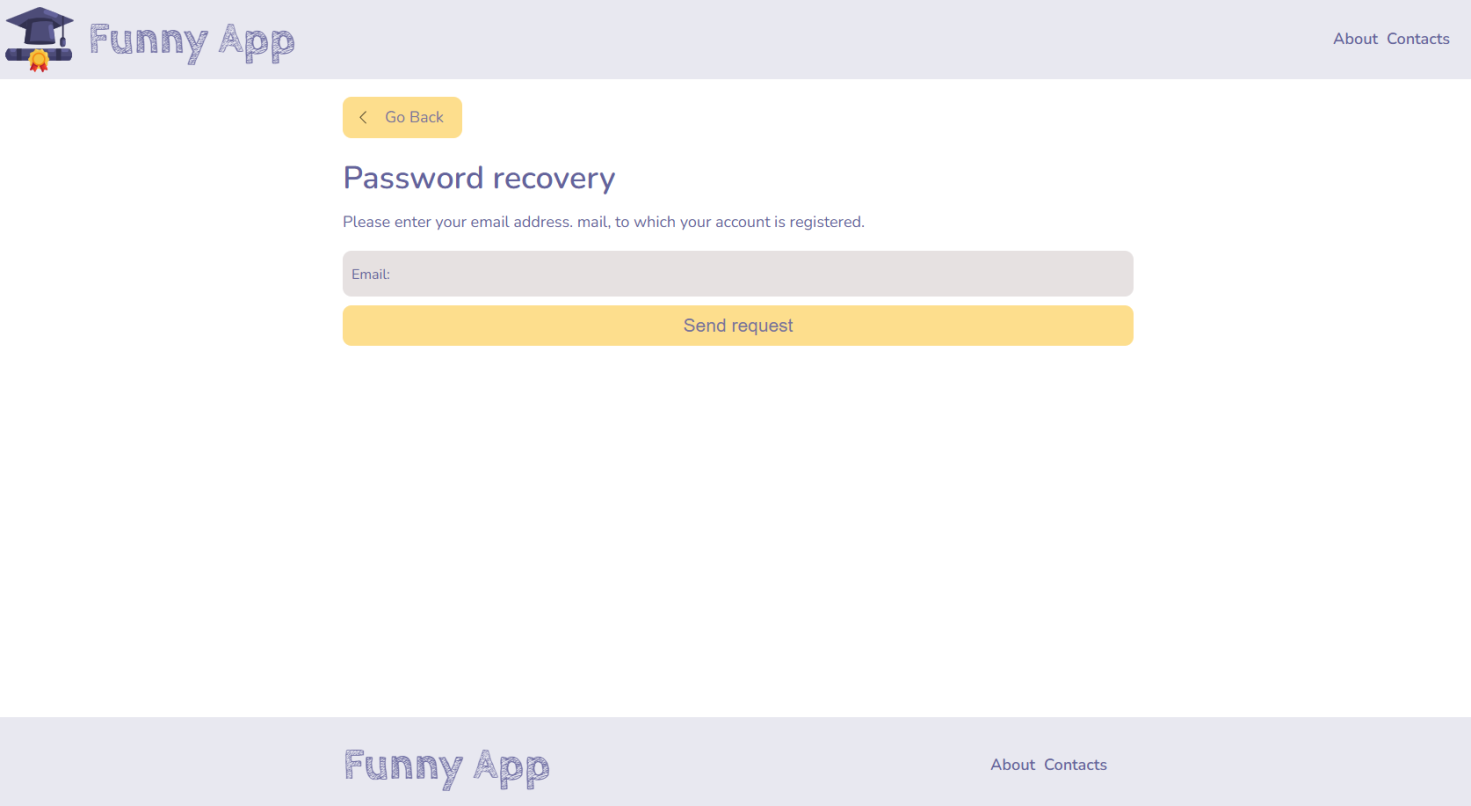
4. הכירו את המערכת

**דף איפוס סיסמה**

המסך הזה יופיע כאשר המשתמש ילחץ על כפתור איפוס הסיסמה במסך הכניסה.

במסך זה, המשתמש יכול לאפס את הסיסמה שלו באמצעות כתובת המייל איתה הוא מחובר למערכת.

לאחר שהוא מכניס כתובת מייל ולוחץ על כפתור "שלח", המערכת תבדוק האם אותה כתובת קיימת במאגר. אם זה לא קיים, המערכת תציג הודעה קופצת המציינת שאותה כתובת אימייל לא קיימת במערכת. אם הכתובת נמצאת במערכת, המערכת תציג את ההודעה המתאימה ותשלח הוראות לכתובת המייל של המשתמש. 

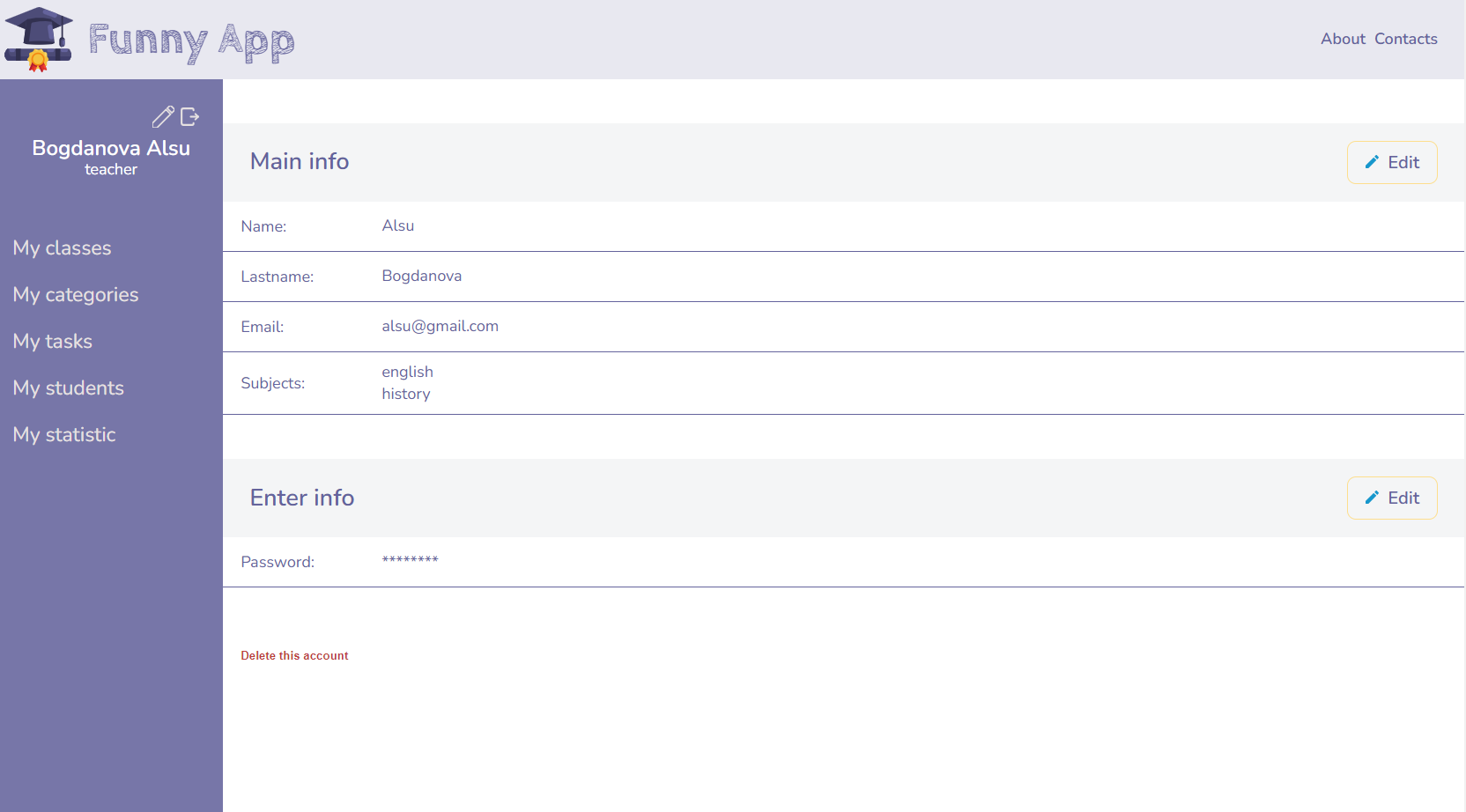
**ממשק את המורה**

**המסך הראשי של ממשק את המורה**

הדף מכיל מאמרים בבלוג. למשתמש מורה זמינות לשונית הבאות:

1. כיתות שלי
2. קטגוריות שלי
3. משימות שלי
4. תלמדים שלי
5. סטטוסטחקה שלי



**דף עריכת פרופיל**

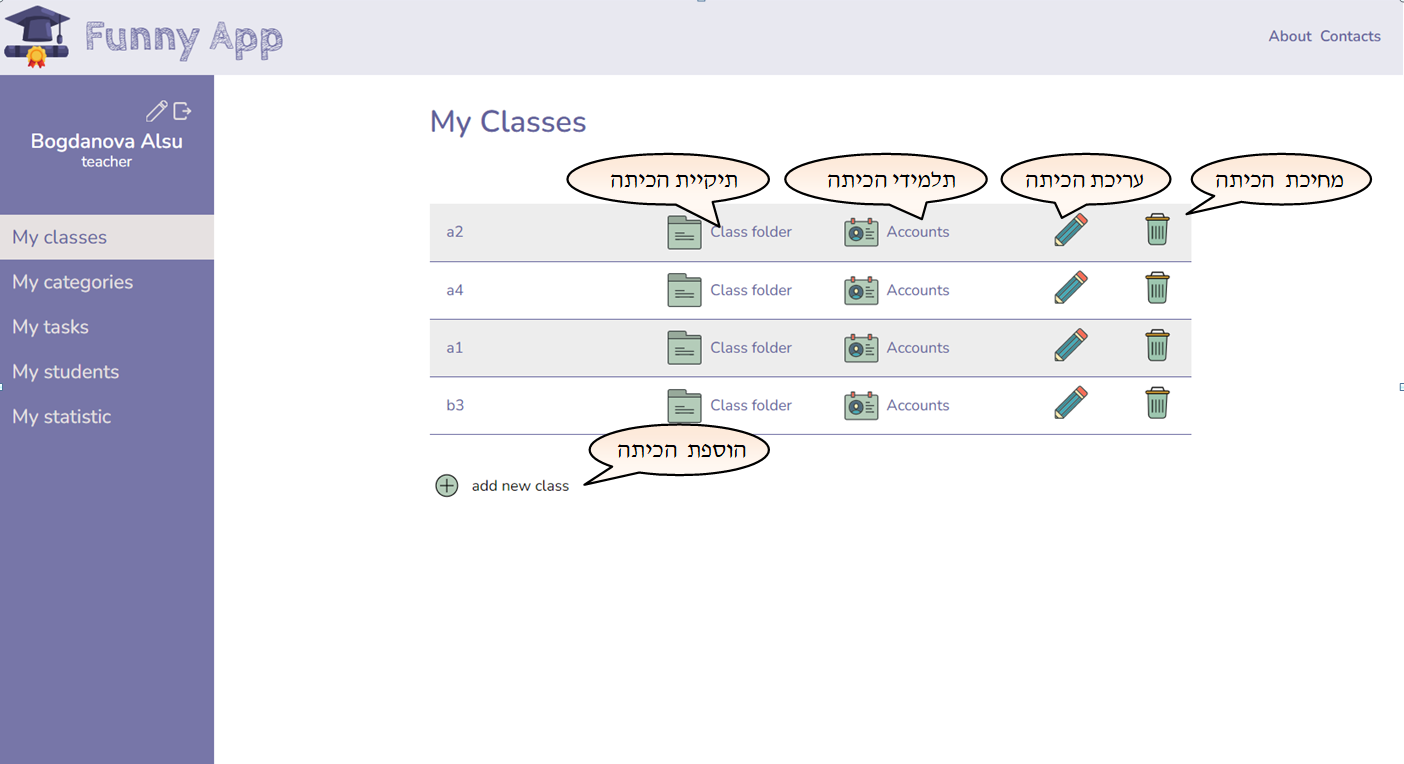
**דף הכיתות שלי**

הנה רשימה של הכיתות של המורה. האפשרויות הבאות זמינות:

1. לשנות את שם הכיתה

2. למחוק את הכיתה (רק אם אין משימות בתיקיית הכיתה ואין תלמידים בכיתה)

3. הוסף כיתה חדשה



**דף תיקיית הכיתה**

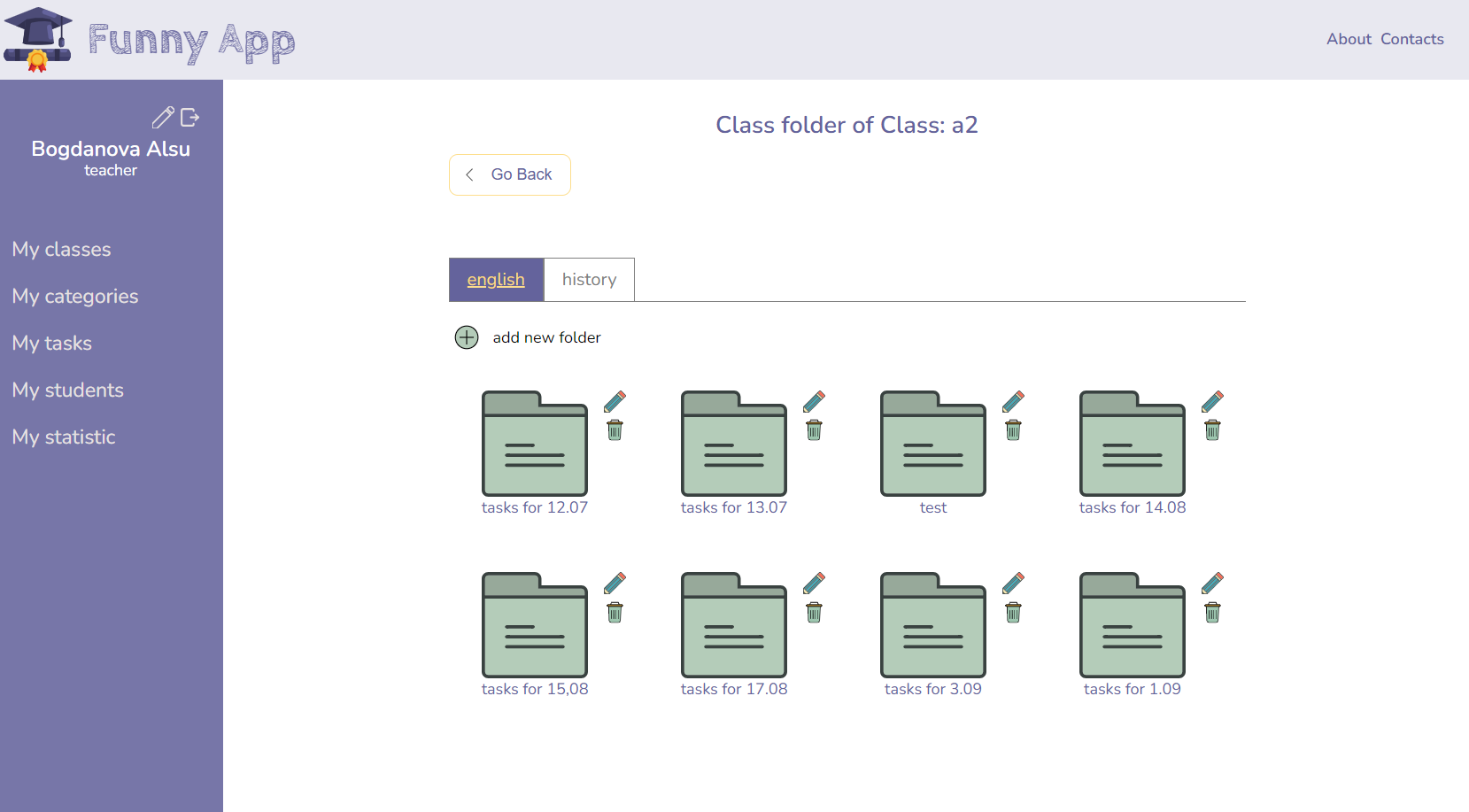
כאן מוצגות תיקיות עם משימות עבור כיתה מסוימת. אפשרויות זמינות:

1. הוספת תיקיה חדשה

2. שינוי שם של תיקיה

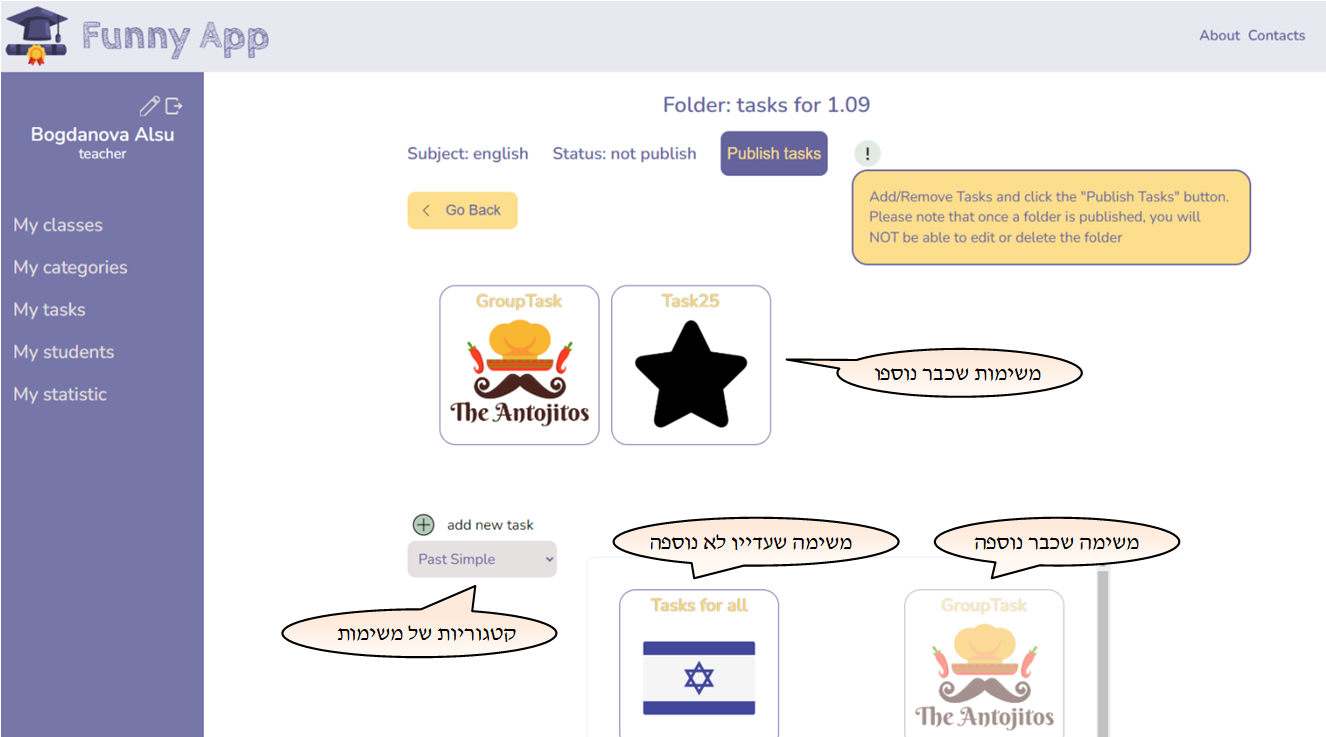
3. מחיקת תיקייה (בתנאי שהתיקייה לא מוגדרת לסטטוס "פורסם")

ניתן גם לעבור בין מקצועות הנלמדים על ידי המורה (לשונית של מקתועות)



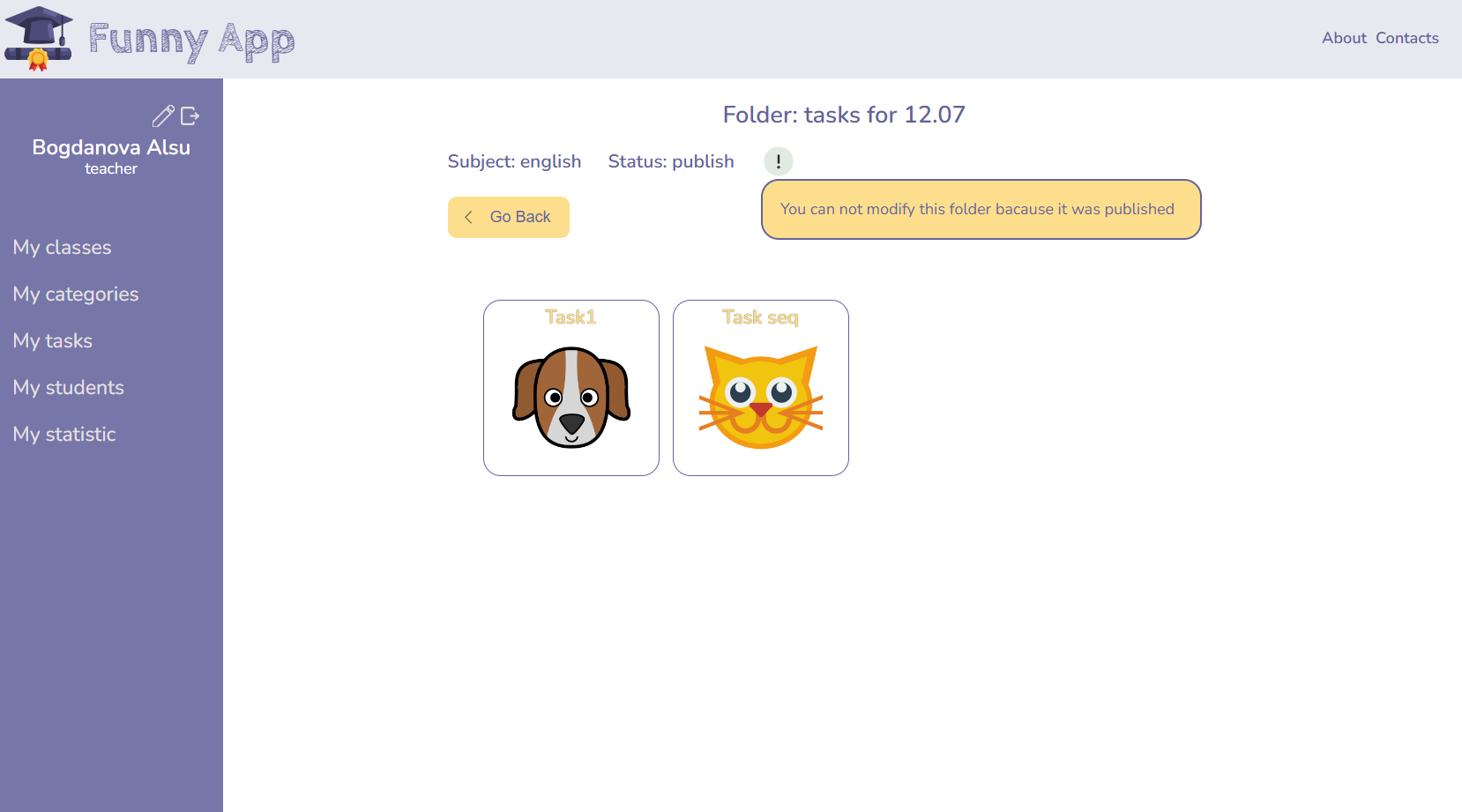
**דף תיקייה שלא פורסמה**

כל עוד התיקיה לא מוגדרת לסטטוס פורסם, ניתן להוסיף ולהסיר משימות על ידי בחירה תחילה של משימות מהקטגוריה הרצויה. אי אפשר להוסיף את אותה משימה פעמיים, היא לא ניתנת ללחיצה ושקופה בצבע

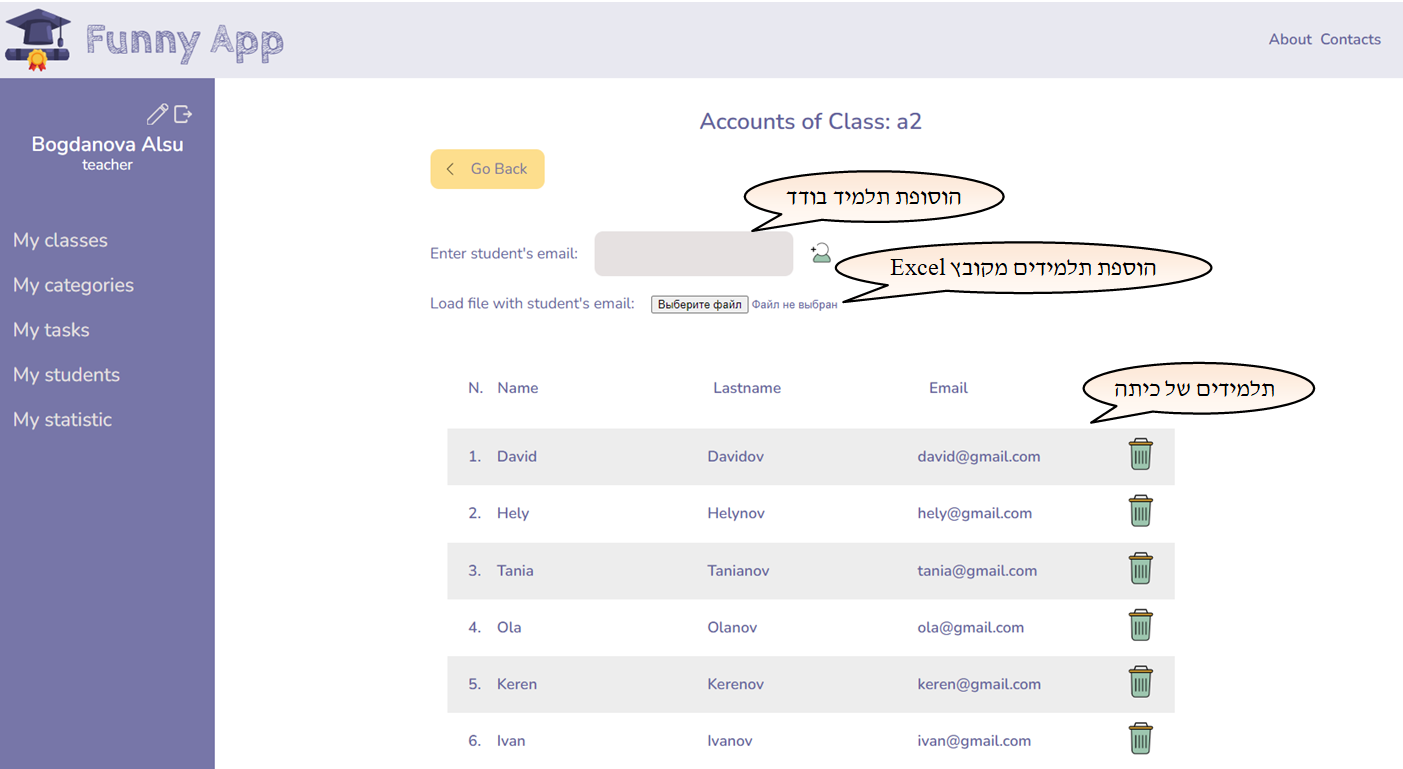


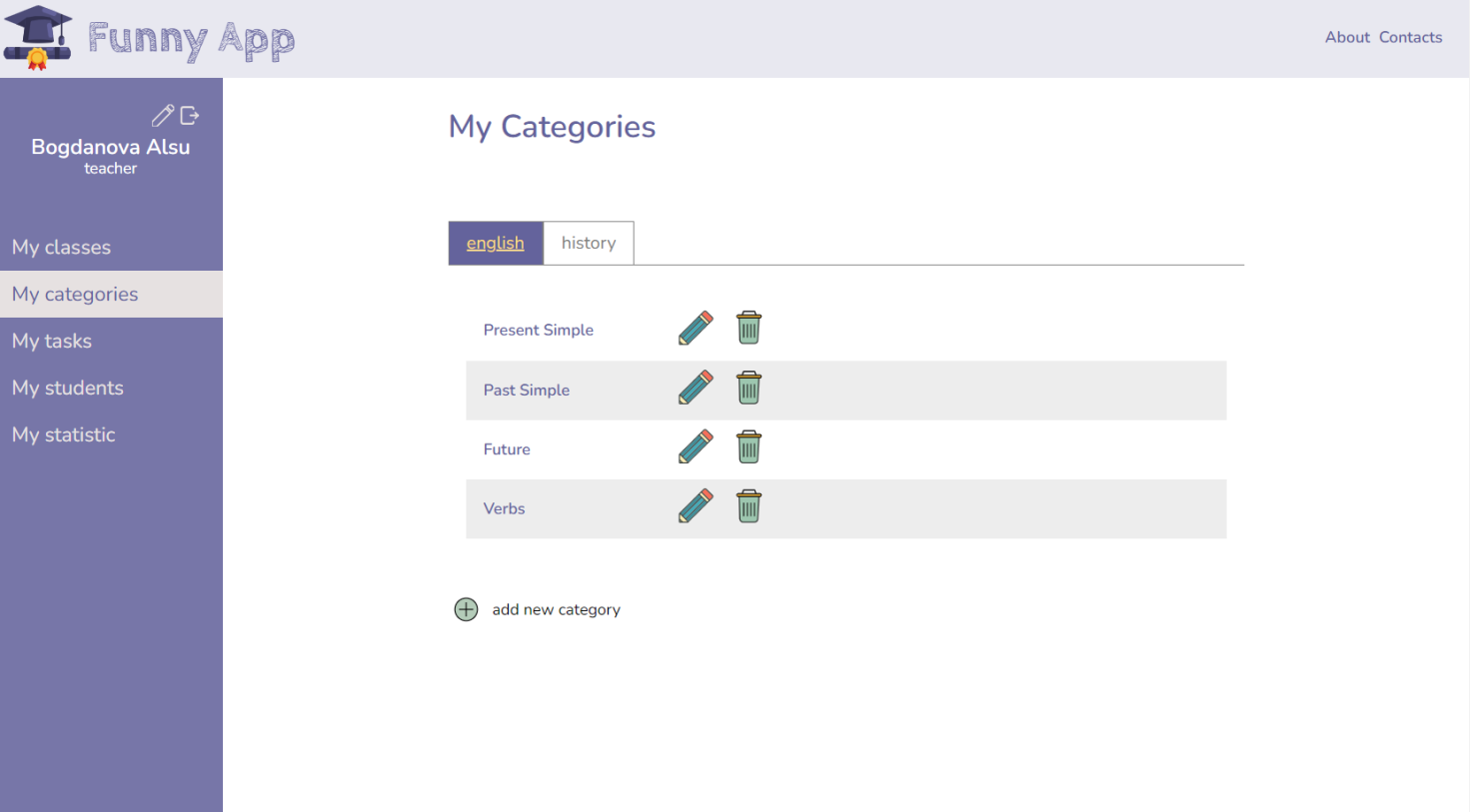
**דף תיקייה שפורסמה**

מכיוון שהתיקיה השתנתה למצב פורסם, אי אפשר למחוק ממנה או להוסיף לה משימות

**דף תלמידים של כיתה**

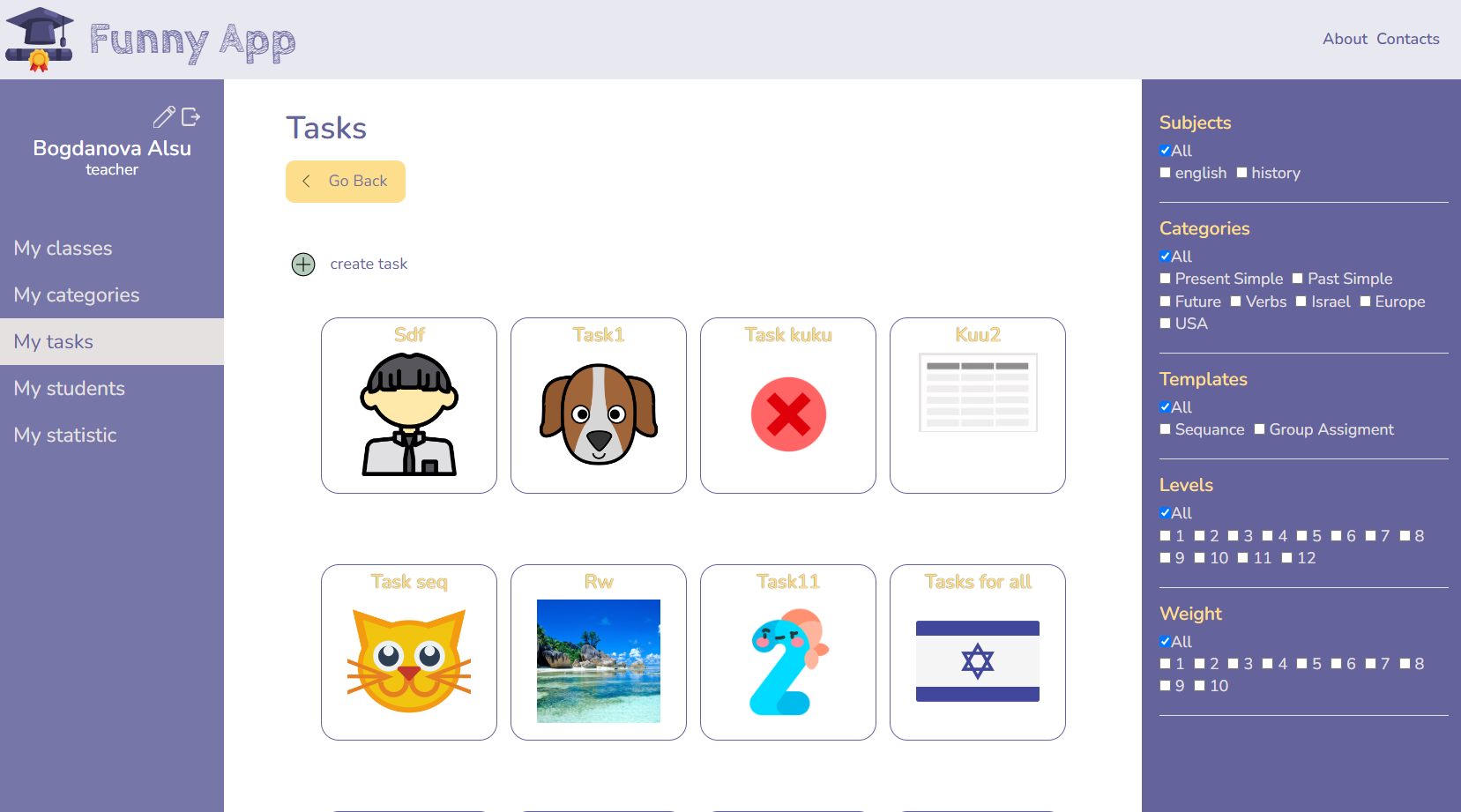
תלמידים של כיתה מסוימת מוצגים כאן. ישנה אפשרות להוסיף תלמיד על ידי הזנת המייל שלו או על ידי העלאת קובץ עם כתובות המייל של התלמידים. ראוי לציין כי התוספת מתרחשת לחשבונות קיימים, כלומר, סטודנטים כבר נרשמו מראש במערכת. אם המייל אינו תקין או שהמשתמש לא נמצא במערכת, תוצג הודעה מתאימה.

**דף קטגוריות שלי**

כאן אנו מנהלים את הקטגוריות של המשימות, עם אפשרות לחלוקה למקצועות  **דף משימות שלי**

כאן אנו רואים את כל המשימות שנוצרו על ידי המורה עם היכולת לסנן אותן לפי:

* נושא
* קטגוריה
* תבניות
* רמת המשימה
* משקל המשימה

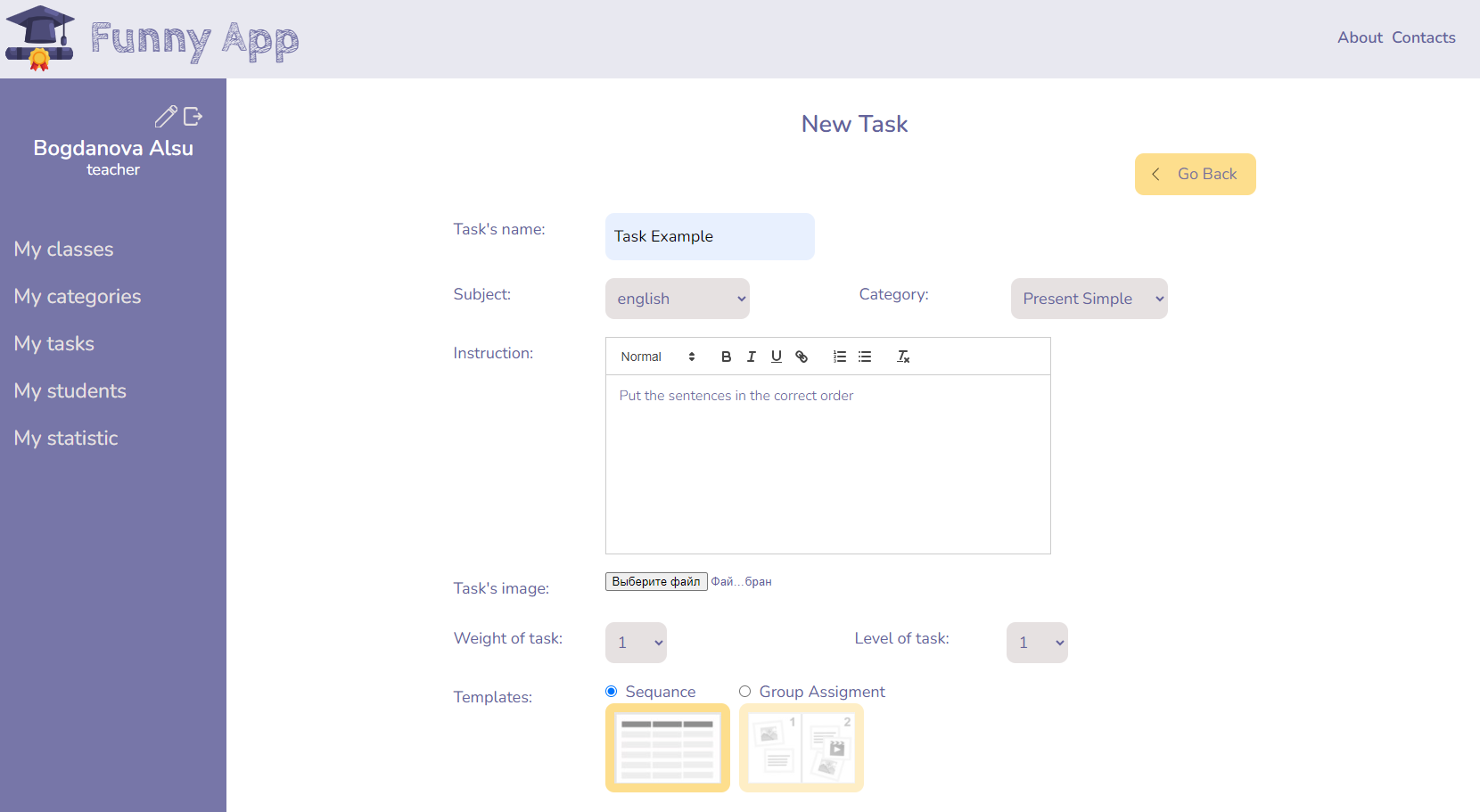
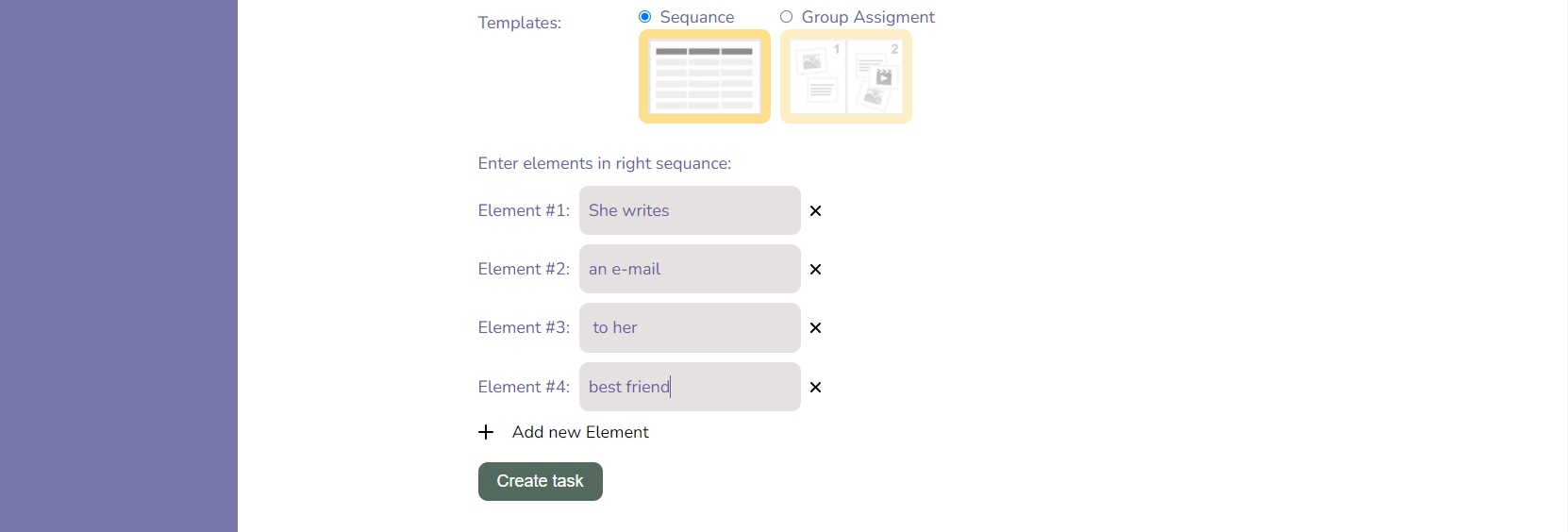
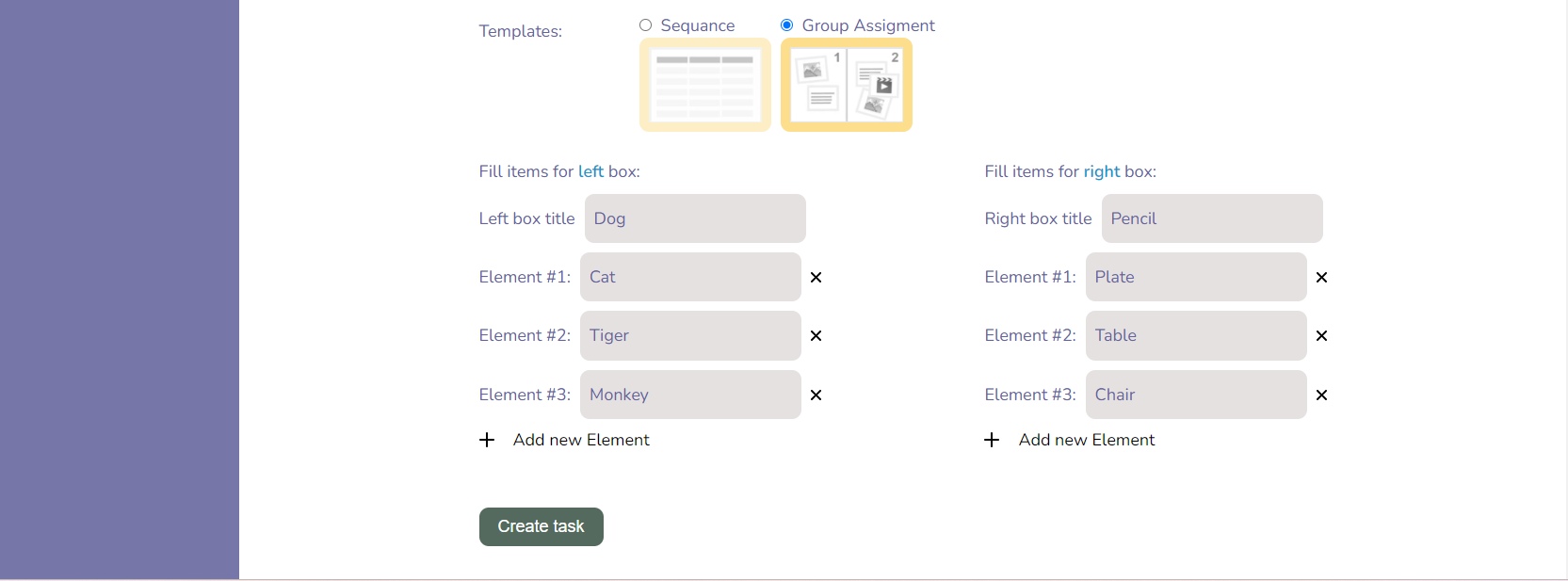
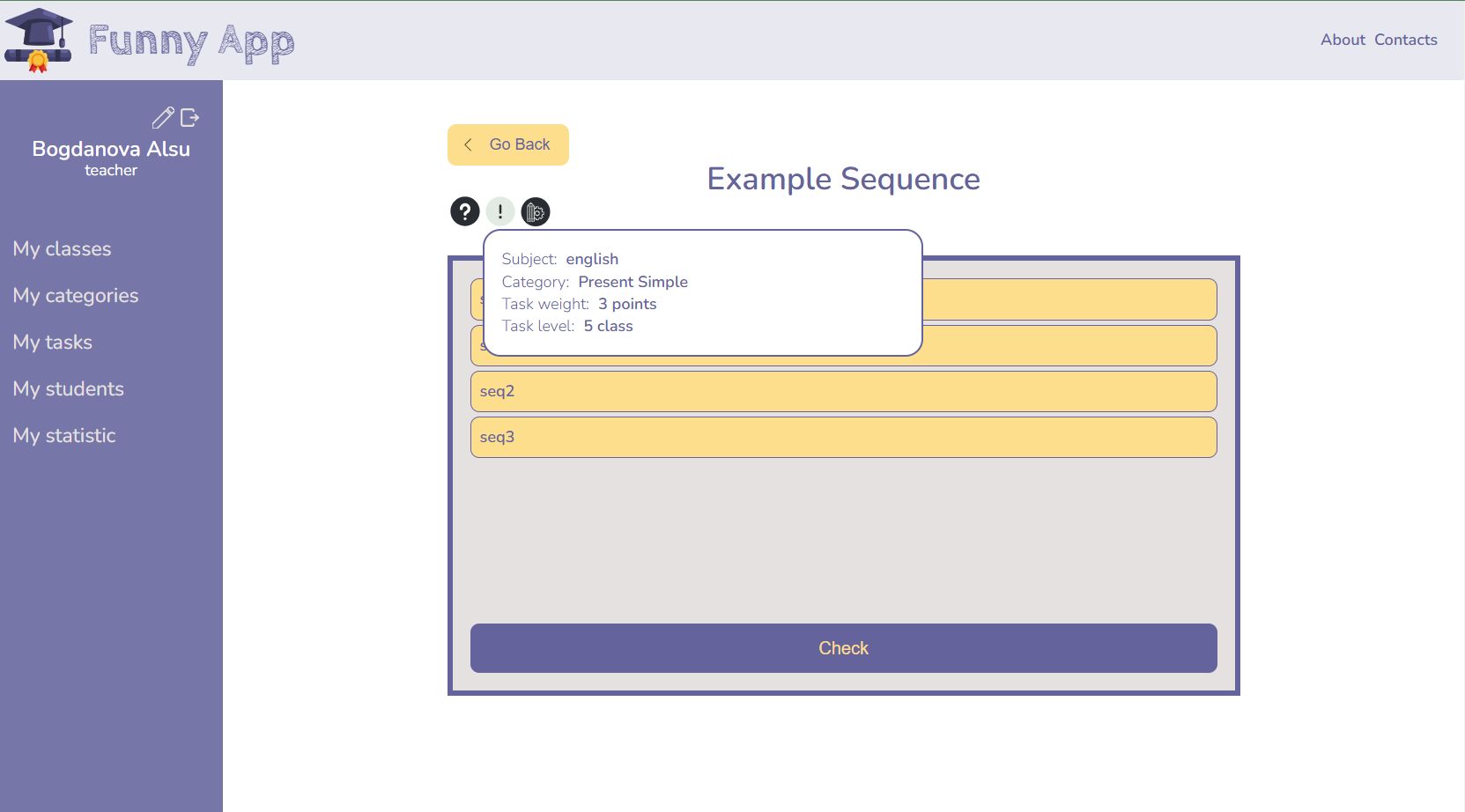
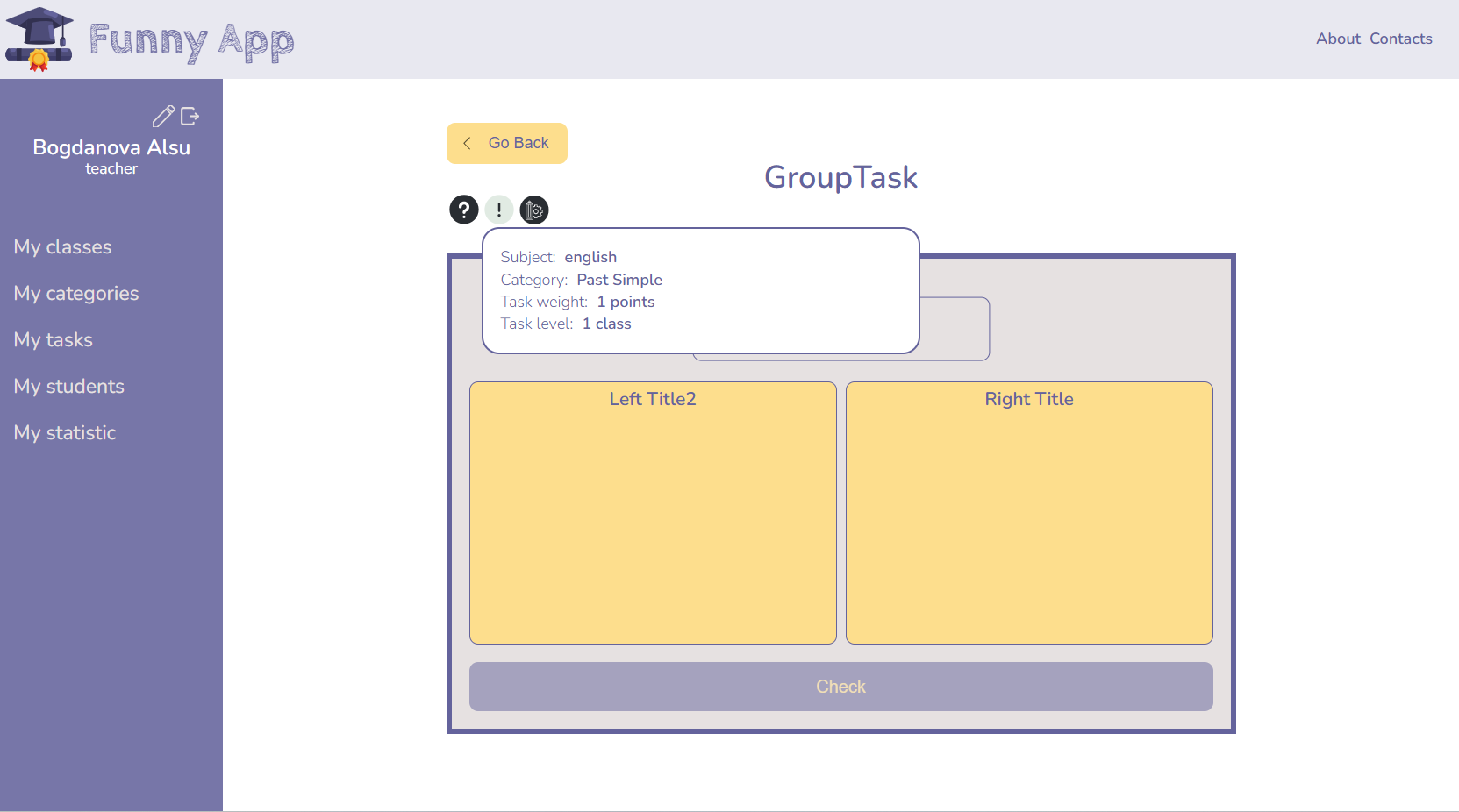
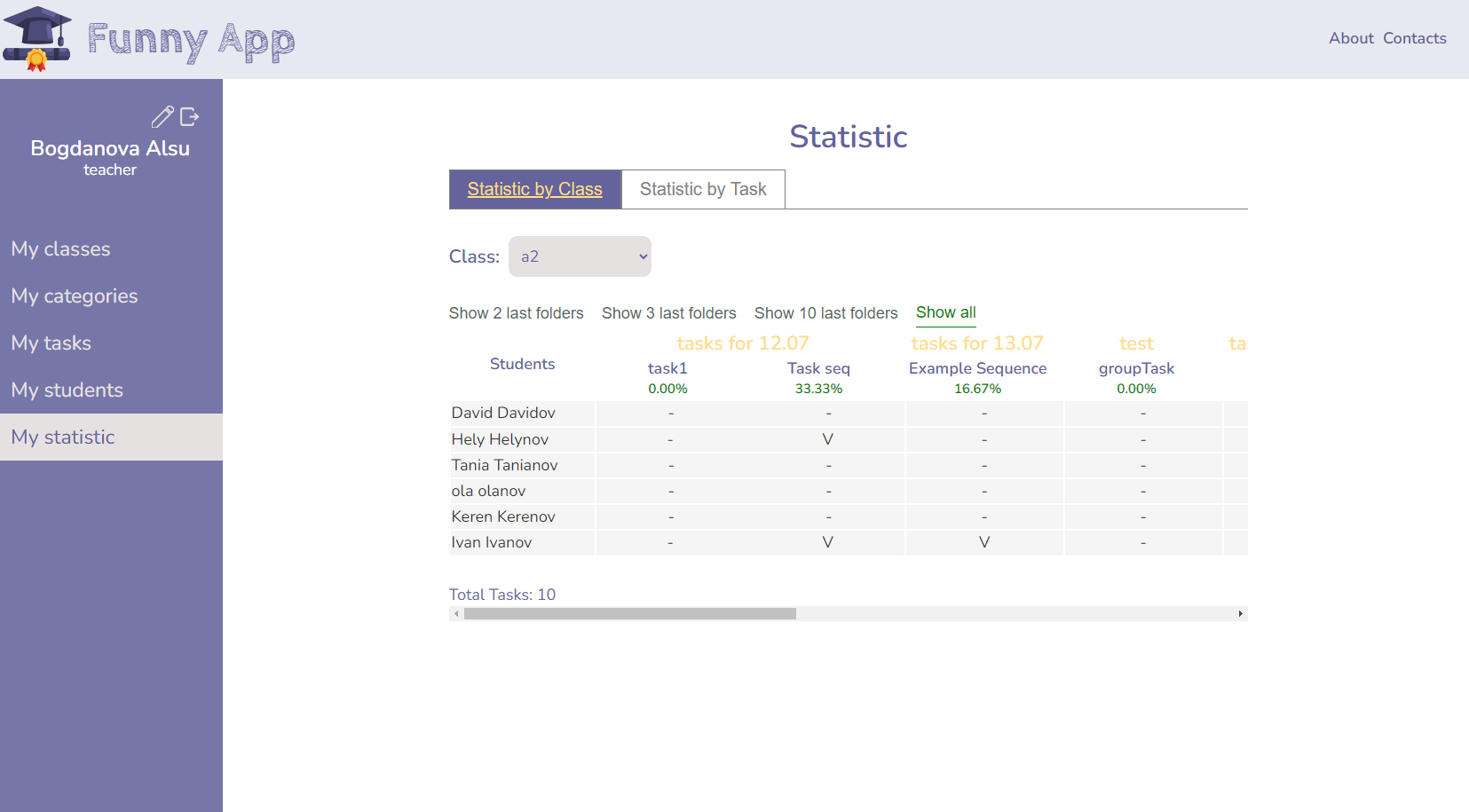
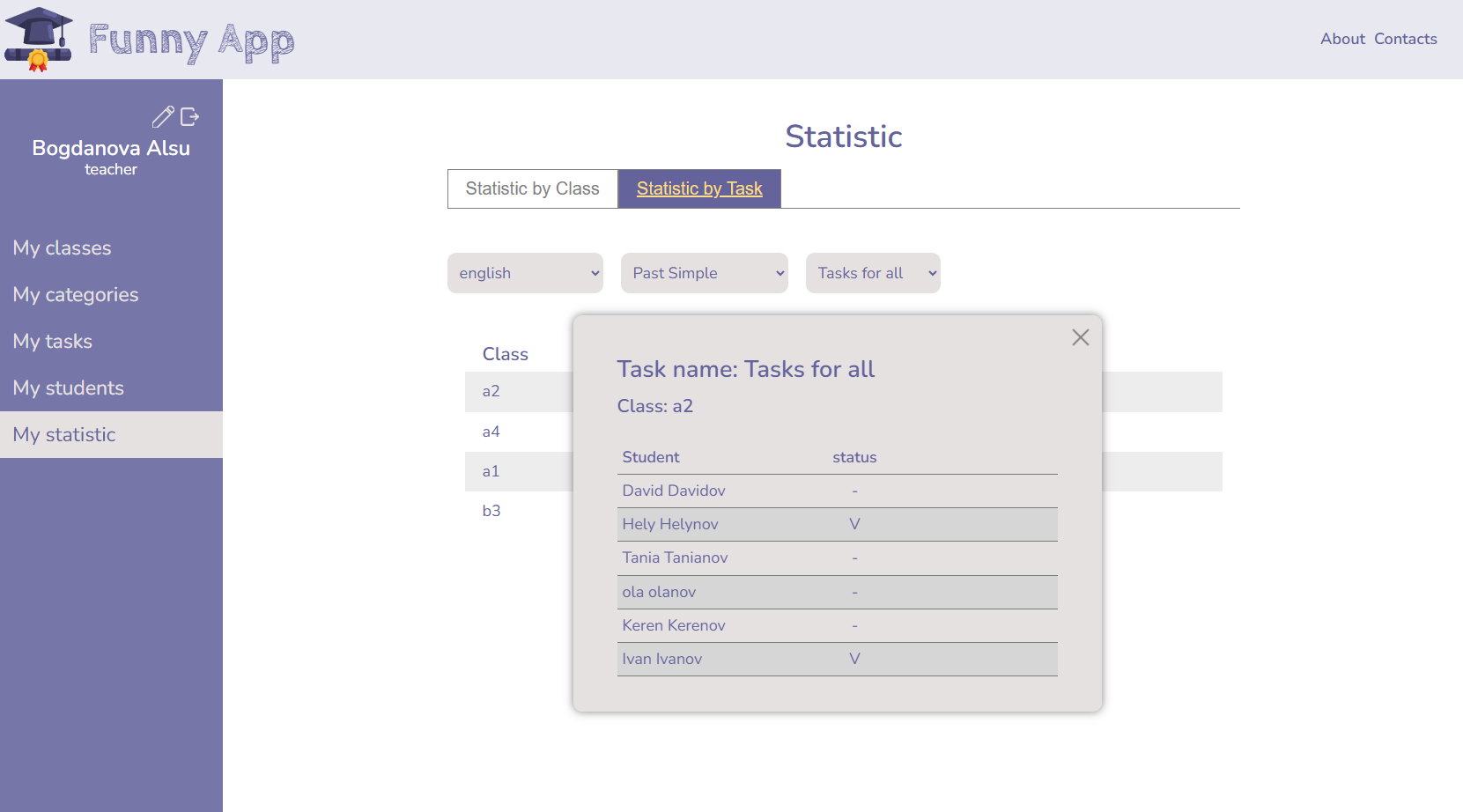


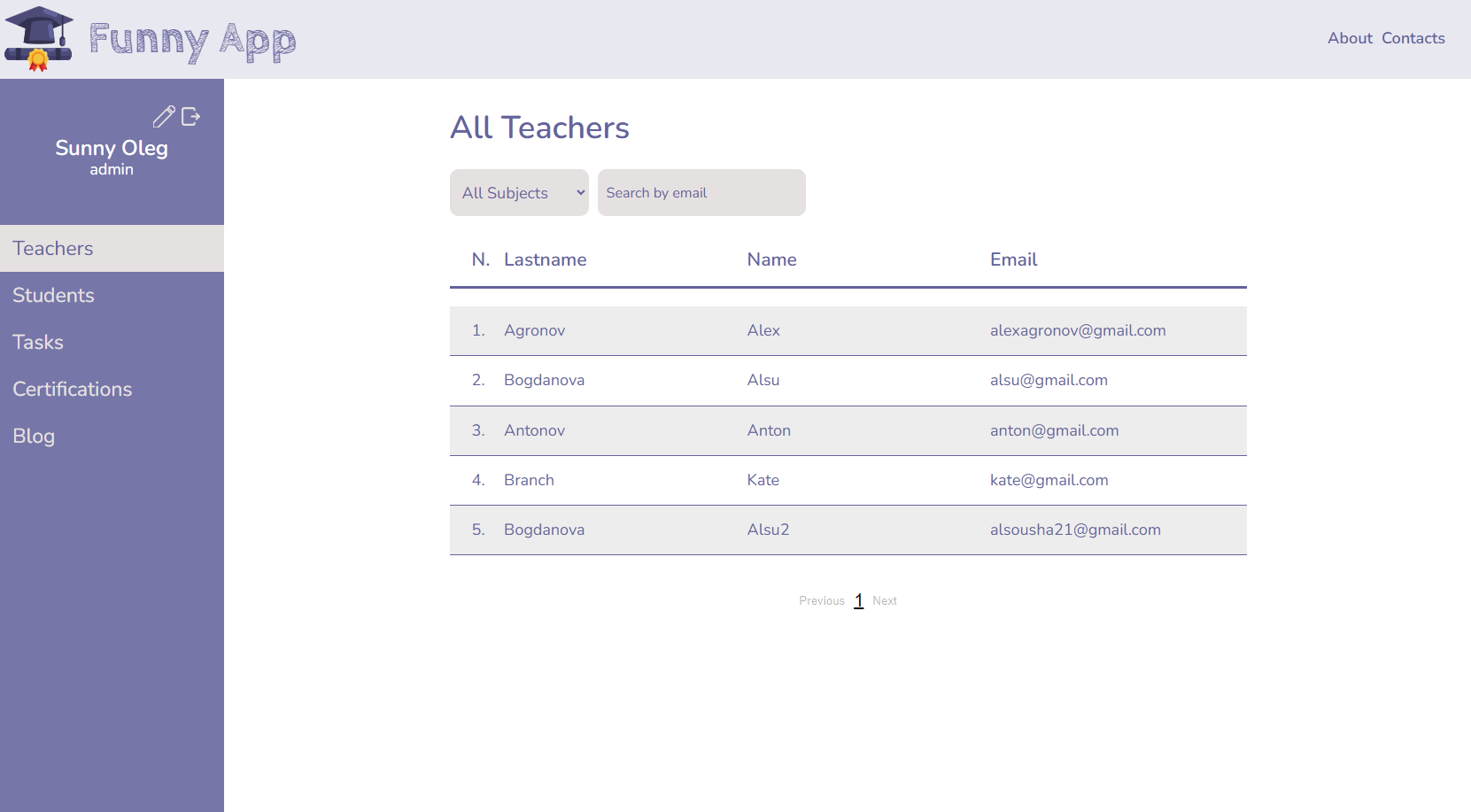
דף יצירת משימה

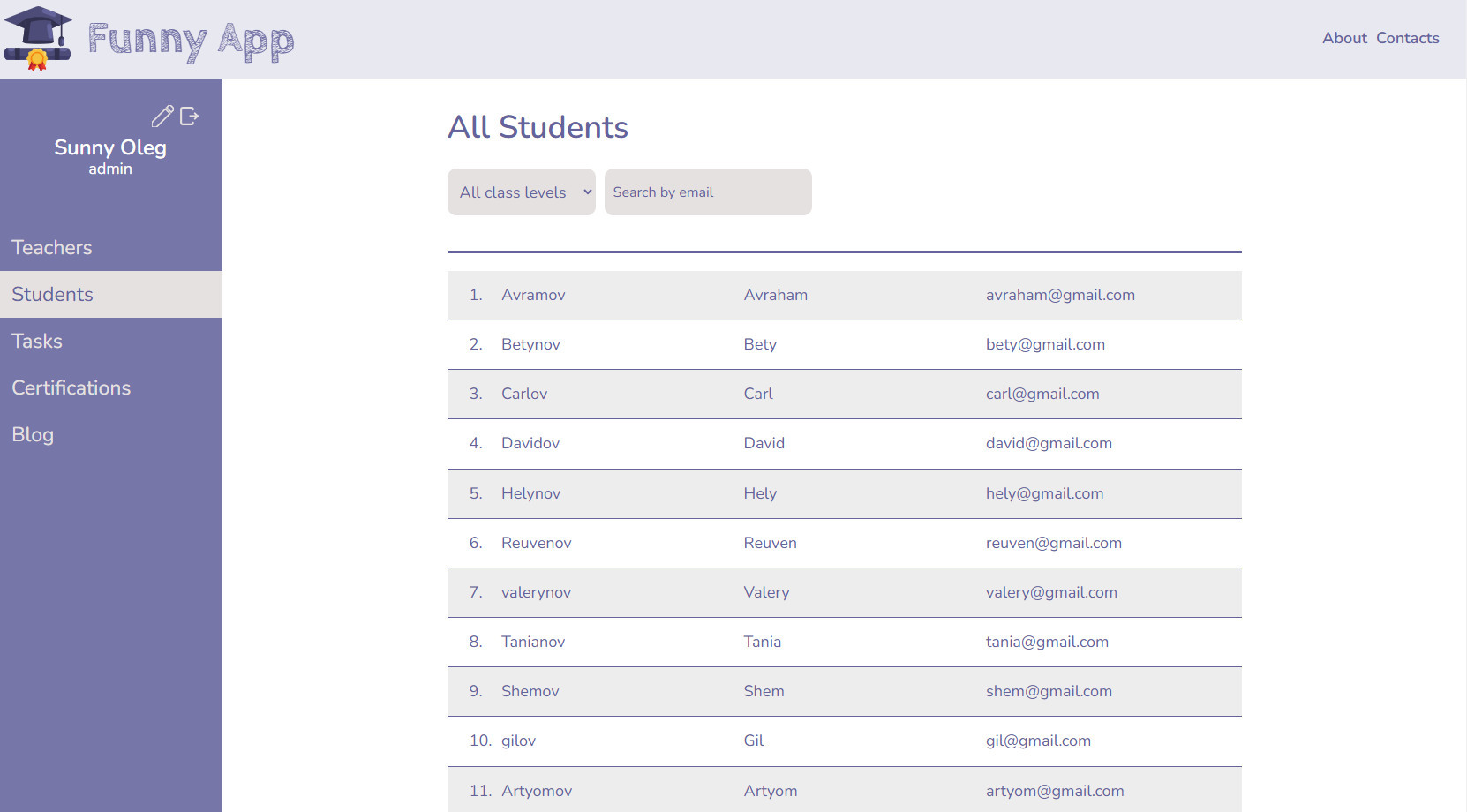
условно страницу можно поделить на две части:

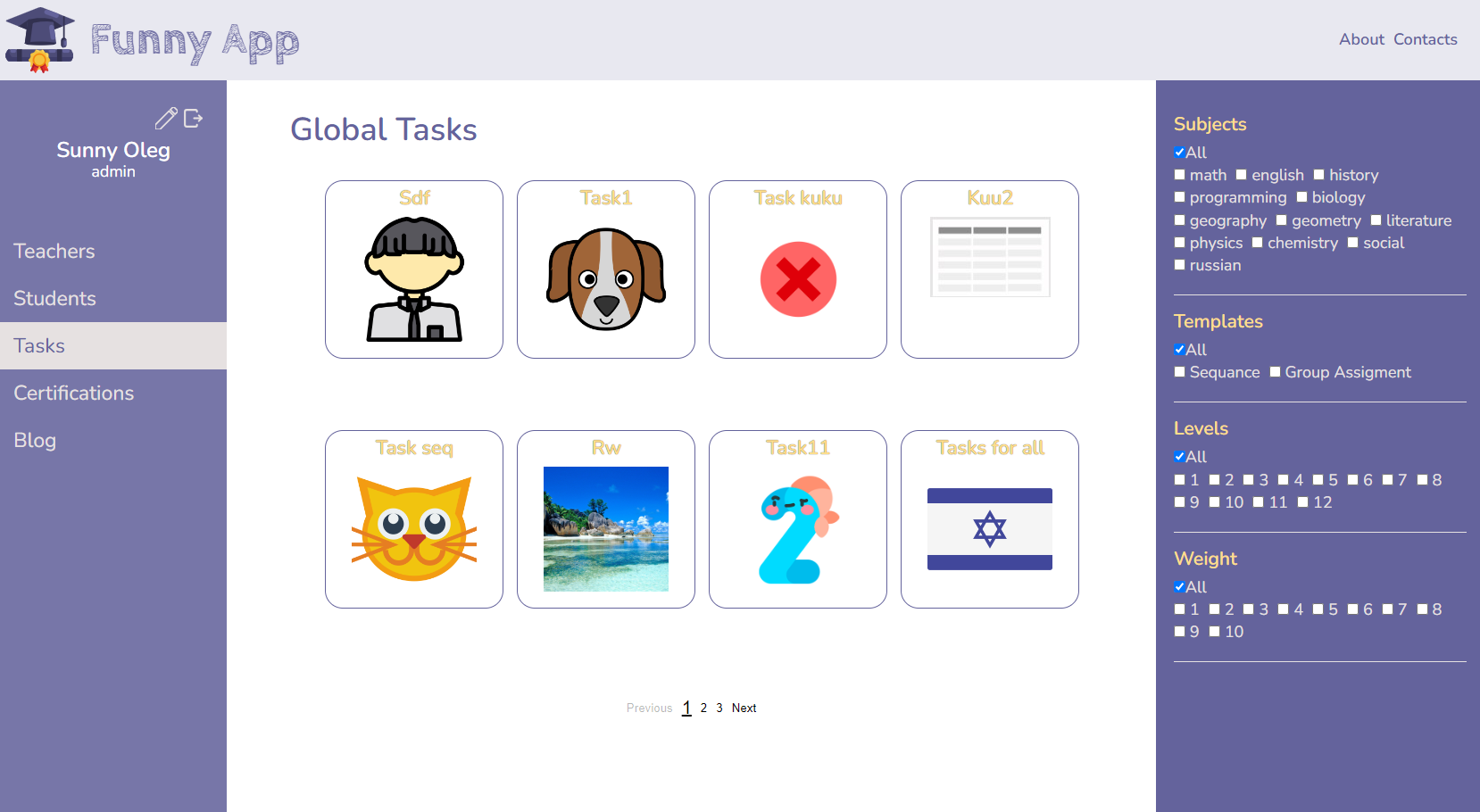
1. заполнение общих сведений о задании (название, предмет, категория, инструкция по выполнению, изображение, вес и уровень).

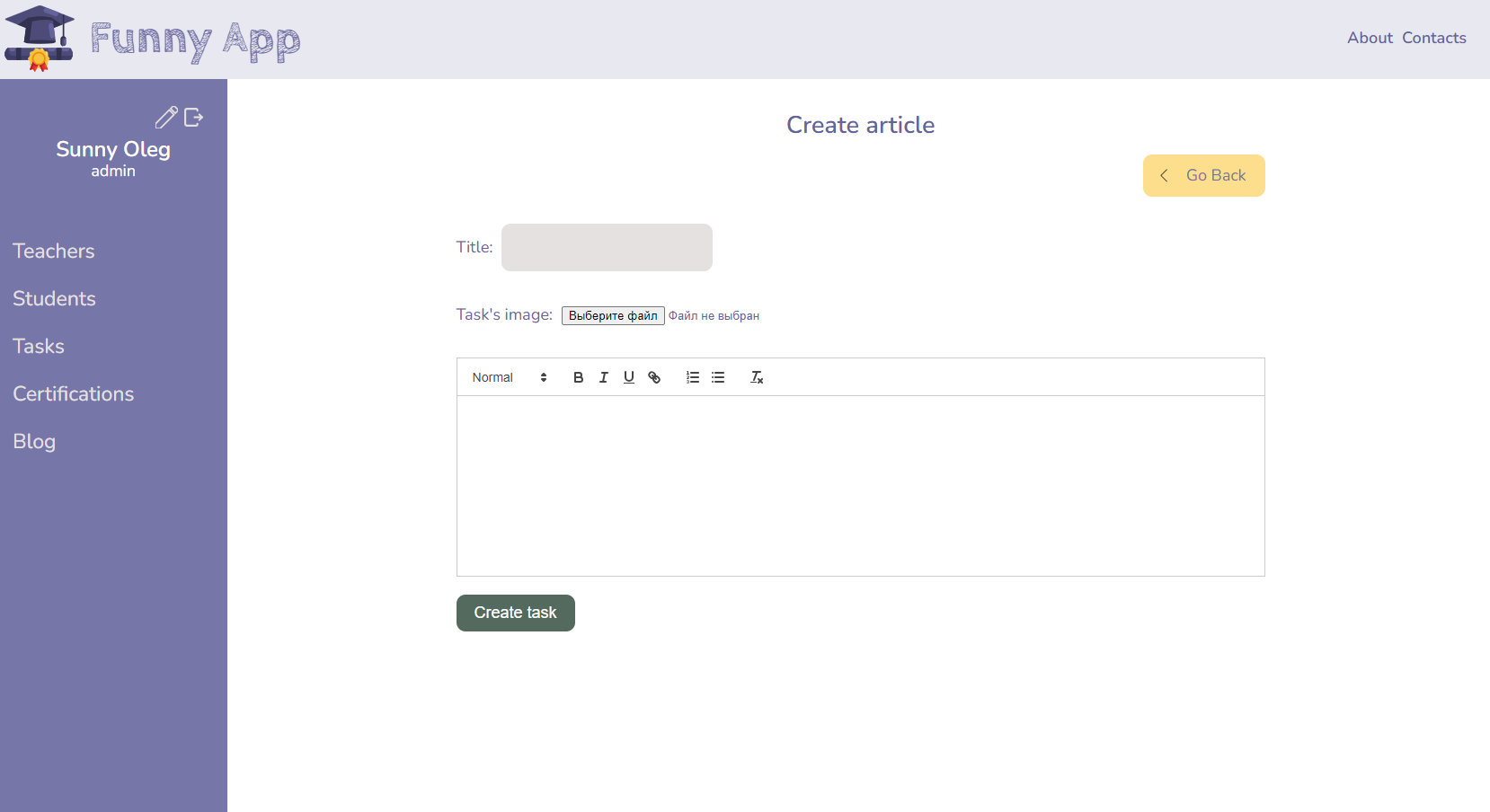
2. заполнения данных, в зависимости от выбранного шаблона. На данный момент доступны два типа заданий: расставить элементы в правильной последовательности (например, составить предложение в правильном порядке или события в хронологической последовательности); распределить элементы в д

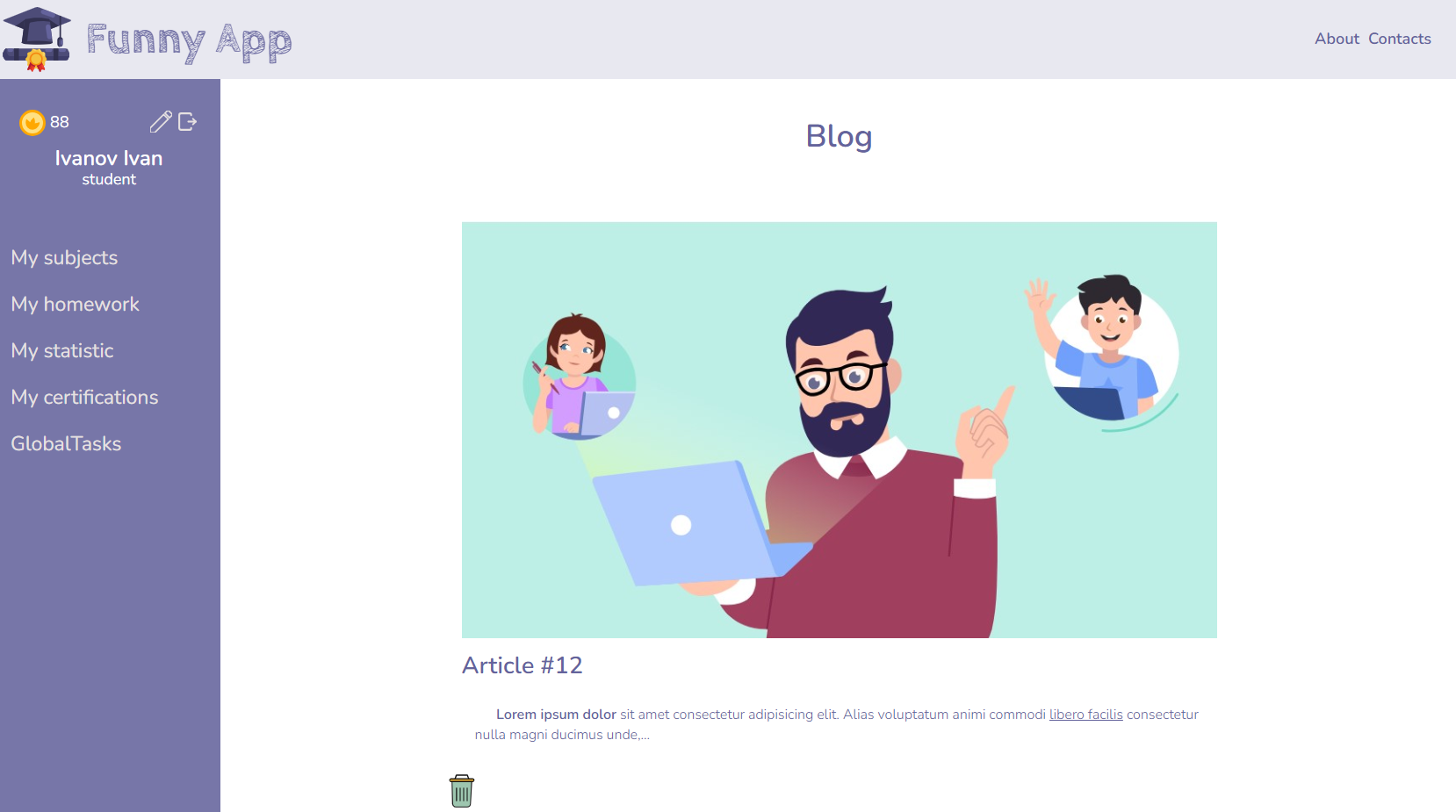
       

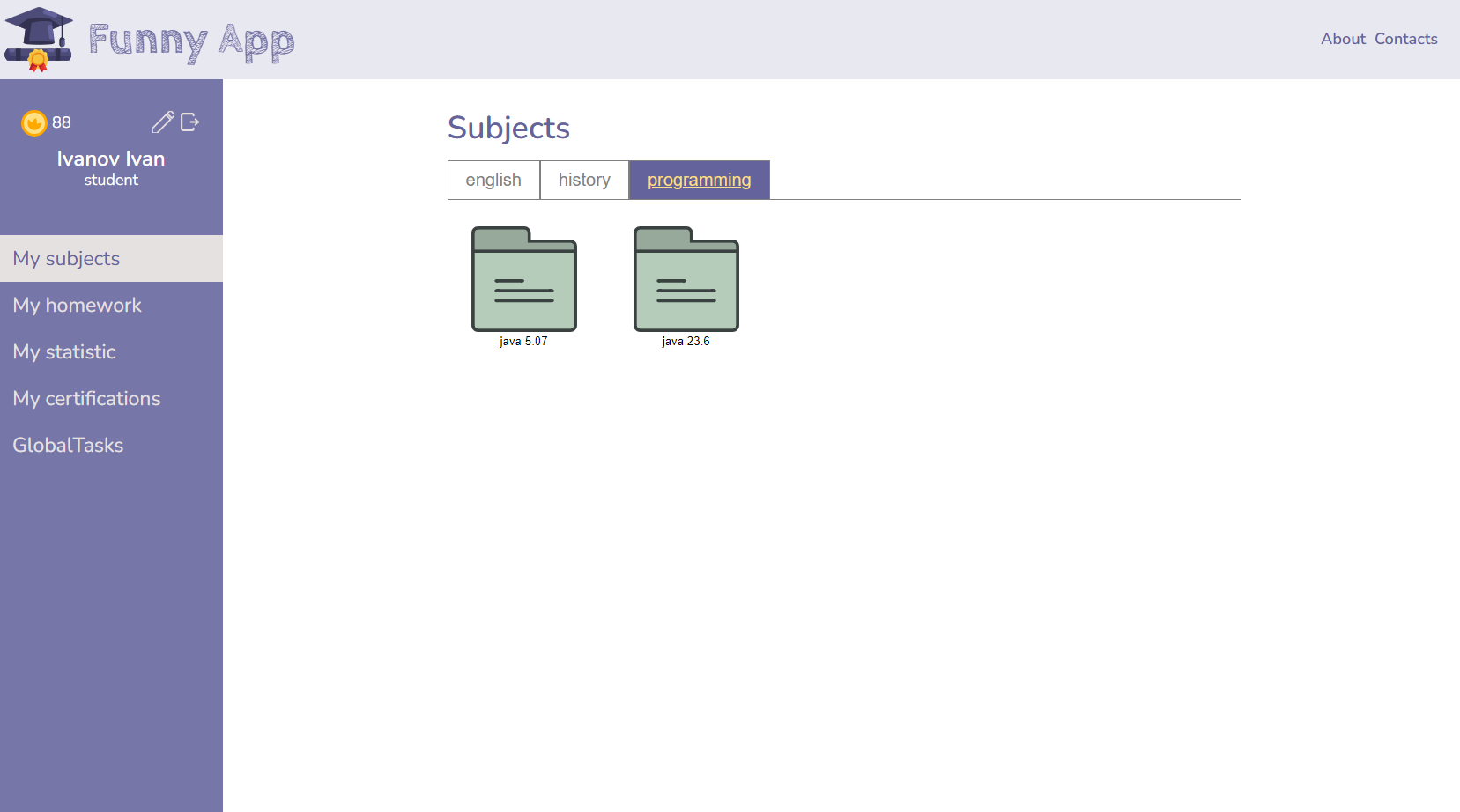


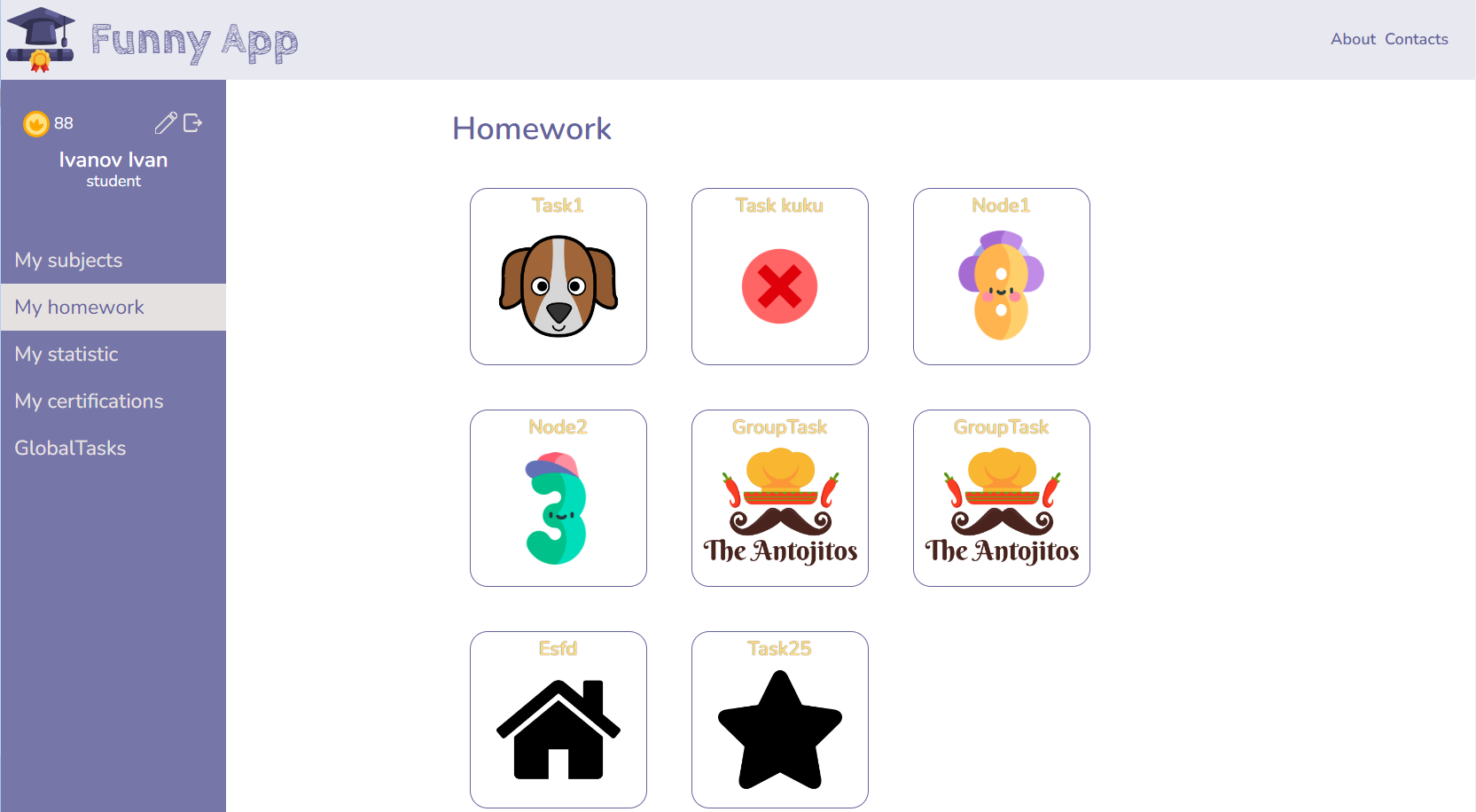


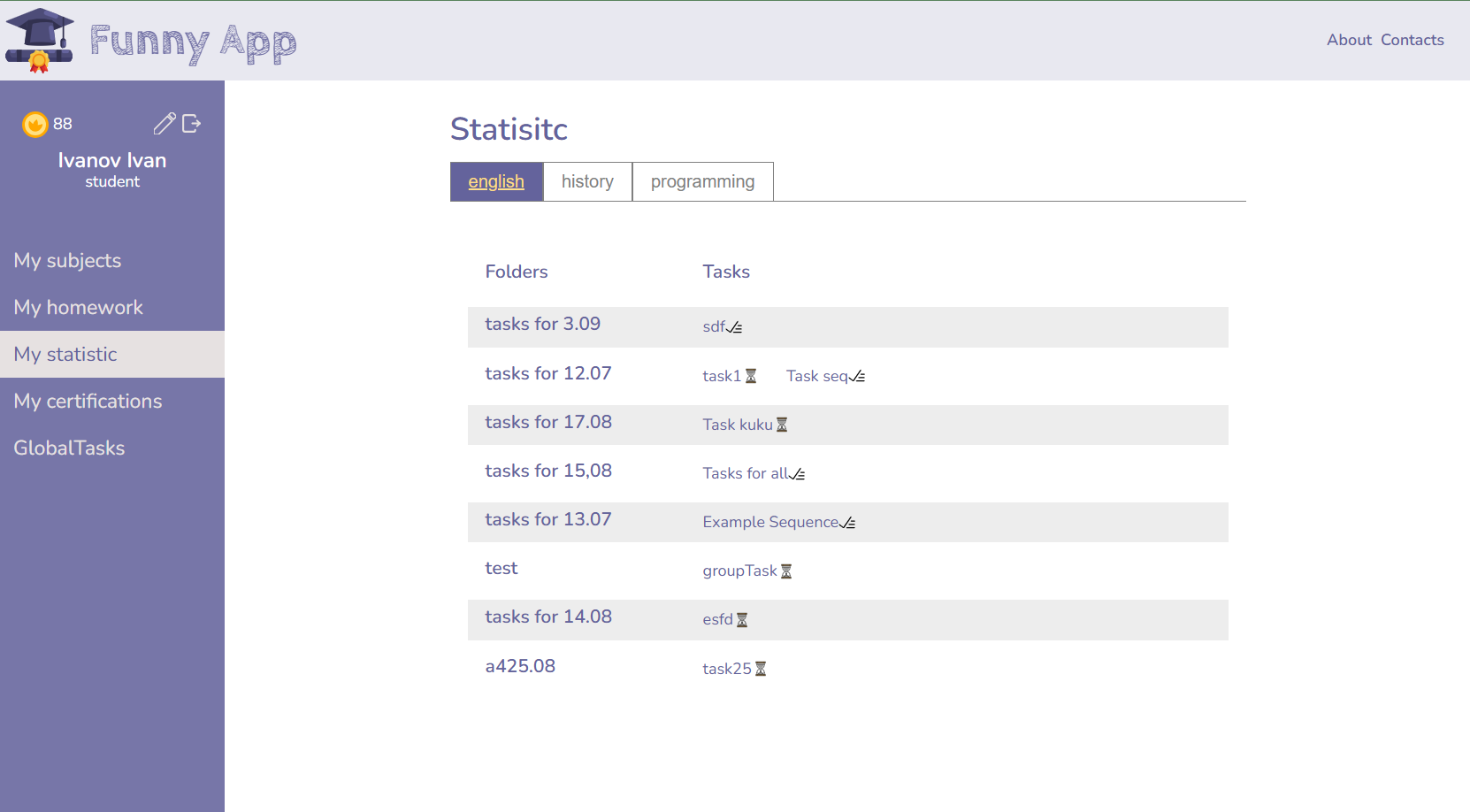












תרשים מחלקות

A diagram of a computer program

Description automatically generated

עבור ייצוג חזותי טוב יותר, אנו מחלקים את התרשים לשני חלקים:Frontend ו-Backend

**Frontend**:

A screenshot of a computer

Description automatically generated

**Backend**:

A diagram of a computer program

Description automatically generatedקוד

התוכניות(כולל דוגמאות הרצה מייצגות ).

* בתחילת כל מחלקה חובה לכתוב הסבר קצר על תפקיד של מחלקה
* חשוב להוסיף הערות לכל פונקציה וקטע קוד משמעותי, **אין צורך לתעד כל שורה .**
* סדר ההדפסה (ברוב הפרוייקטים):
* סדר הצגת מחלקות:
* מודולים לעבודה עם נתונים (Model) - BE
* מודוליםהאחרים על לוגיקה של המערכת (View-Model) - BE
* מחלקותאו מודולים של ממשק משתמש (View) - FE
  + כל מה שאינו מחלקה (פונקציות, אובייקטים נוספים ).

**אין להוסיף לפרוייקט קוד שאתם לא כתבתם .**

**שימו לב ! הקפידו על כל שלא תהייה פונקציות ארוכות ( מעל 20 שורות קוד ) – אם ישנן כאלו, פרקו אותן לפ' משנה .**

סיכום ומסקנות

* רפלקצייה אישית - מה למדתם ? האם עמדתם בציפיות של

עצמיכם ? (לא חובה )

נספחים

אם יש, פרק רשות ( לא חובה)