

הבדלים בין דיאגרמה ראשונית לסופית

UserController:

- ביוזר קונטרולר נוסף מהגרסא הראשונית לסופית שדה ובו נשתמש בשביל לסמן id לכל יוזר, כיוון שהבנו שבשביל לתת id ייחודי לכל יוזר, נוסיף שדה נוסף המיועד לכך.
- הוספנו את פעולה GetUser() בשביל גישה מהירה מה Service Layer לכלל היוזרים.
- IsValidEmail החליפה את CheckEmail(), החלטנו שהפעולה תהיה void ונתפוס קלט לא תקין וחריגות.
- GetUserTop נוספה בשביל לגשת לשדה המתאים. שינינו את Register ל-void כי הבנו שאין צורך להחזיר יוזר.

User:

- הוספנו שדה id ו-boardTop בשביל לייצר id ספציפי לכל לוח.
- הוספנו שדה LoggedIn בשביל למנוע ממשתמשים לא מחוברים לבצע פעולות.
- הוספנו פעולות get לשדות של יוזר.
- הוספת פעולות כניסה ויציאה עבור כל משתמש בעצמו, והוספת לוח ומחיקה כך שכל יוזר מנהל את הלוחות של עצמו.
- בנוסף, נוספה פונקציית עזר לוידוא סיסמא בשביל ChangePassword() ו-Login.

Board:

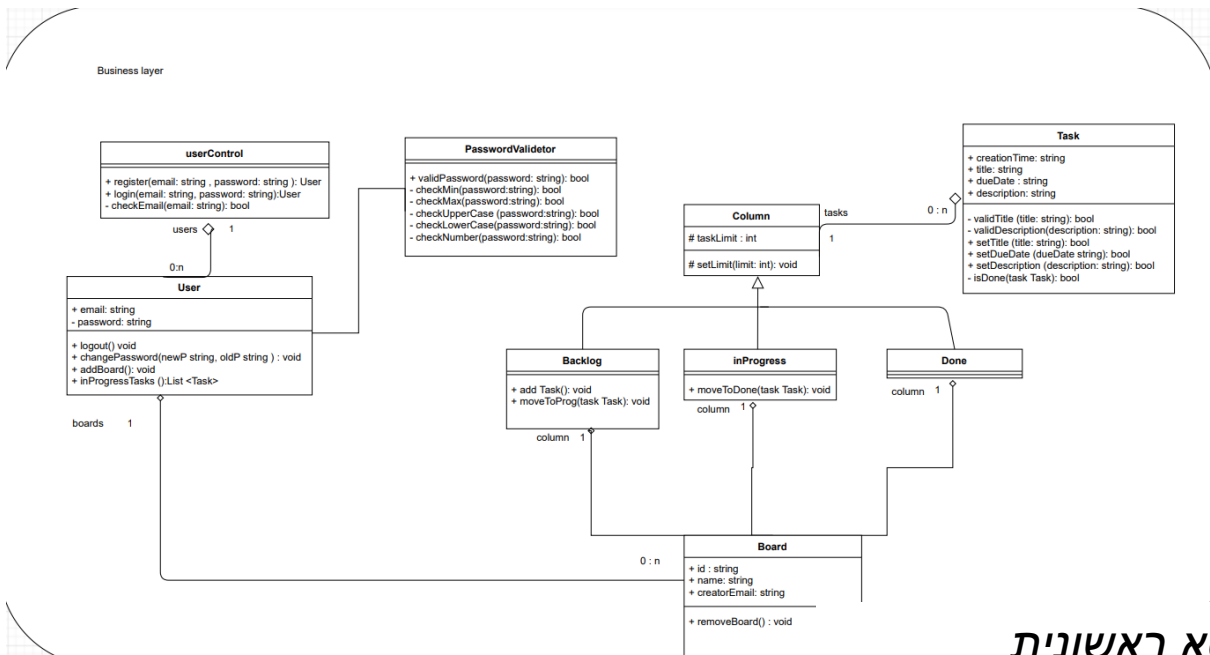
- הוספנו שדות done, in progress, backlog - במקום ליצור 3 אובייקטים שונים המייצגים עמודה שונה, ישנם 3 שדות שכולם עמודות, מכיוון שאין פונקציונליות כל כך שונה ביניהן, החלטנו לשנות ולא לייצג כל אחד כאובייקט נפרד.
- כיוון ששהחלטנו לייצג עמודה כשדה, הוספנו פונקציות בבורד, בהן ניתן לקבל את תוכן העמודה ואת שמה.
- כמו ביוזר קונטרול ויוזר, הוספנו שדה id בשביל לתת id ייחודי לכל משימה.
- בנוסף, הבנו כי כל פעולה של ניהול המשימות בלוח, צריכות להיות בבורד (הוספה, מחיקה ועדכון).

Column

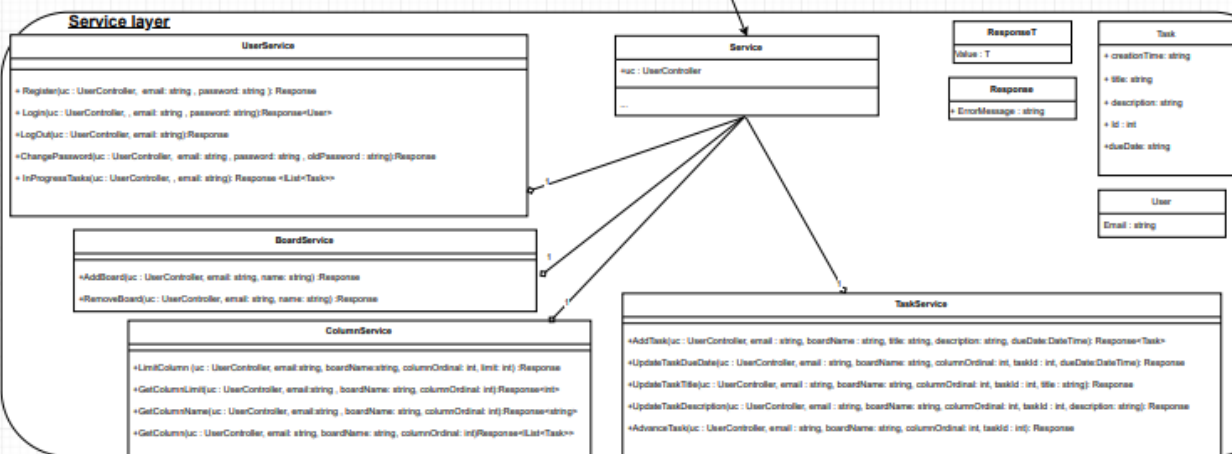
- הגדרנו אובייקט עמודה אחד, מהסיבה שנמקה לעיל בBoard.
- הוספנו פעולות get set לTaskLimit, TaskList בשביל שיהיה אפשר לגשת לשדות.
- הבנו שבשביל לממש פעולות על Task, נצטרך להעביר אותן דרך העמודה שהמשימה נמצאת בה (הוספה, מחיקה ועדכון).
- הוספנו פעולה Contains כפונקציית עזר לפונקציות אחרות.

Task:

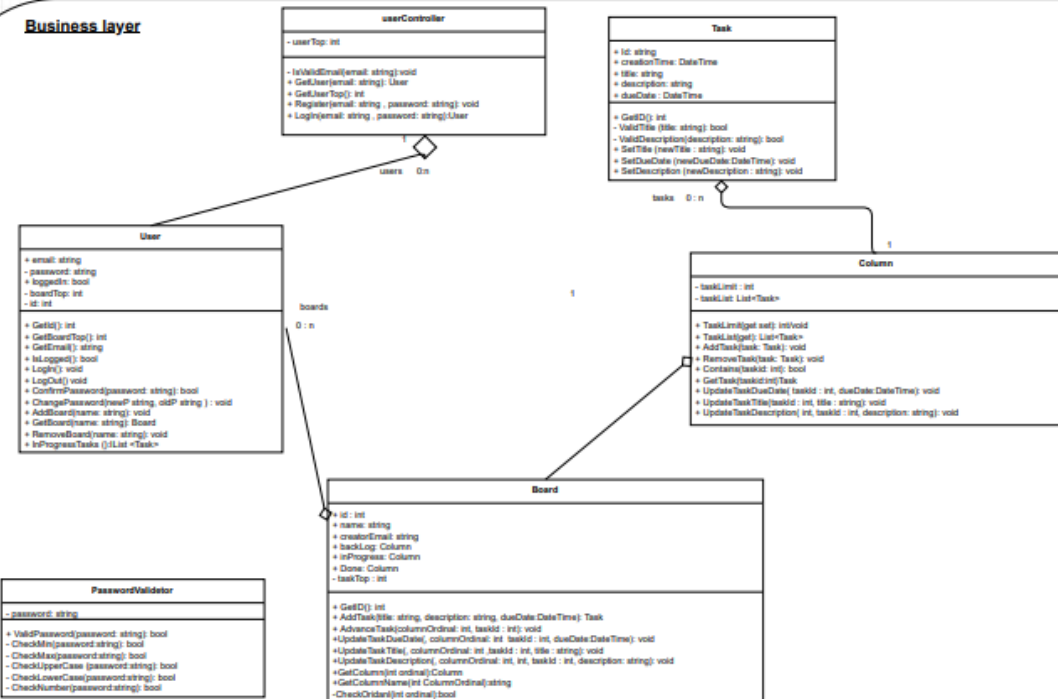
- ההבדל מינורי בין הגרסא הראשונית לסופית באובייקט משימה, הוספנו לגרסא הסופית שדה id הנחוץ עבור כל משימה.
- הסרנו את הפונקציה IsDone שבדקת האם המשימה נמצא בעמודה האחרונה (ממנה אי אפשר לבצע פעולות – פעולה לא נחוצה כי הבדיקה האם משימה יכולה להערך מתבצע באובייקט אחר)



גרסה ראשונית



Business layer



גרסה סופית