**שלב 3   
חלק A :**  
כמפתחי אתר PLAY TO LEARN וכאחראים על ניהולו כל אחד מאיתנו אחראי על עבודה עם לקוחותינו באופן פתוח ומכבד.  
הקוד האתי שלנו מפרט את הסטנדרטים המצופים מאיתנו על מנת לתמוך בהתנהגות נכונה מצדנו.  
העקרונות האתיים של חברתנו הינם :  
**הציבור**: חברתנו שומרת על הקשר של המערכת עם הציבור בעל המוגבלות בקריאה ושיש להם צורך בקריאה ובלימודי שפה בין אם זה ילדים ובין אם זה מבוגרים ששפת האם שלהם אינה אנגלית.

**לקוח ומעסיק:** חברתנו תנהג בדרך שתשרת את האינטרסים של לקוחותיה ומעסיקיהם והכול עפ"י צרכי הציבור וטובתו, עם מתן השירות המיטבי והמקסימלי הדרוש לכך.

**מוצר :** חברתנו מוודאת שהמוצר(אתר האינטרנט) וההתאמות שאנחנו עורכים לאתר זה יעשו עפ"י התקנים המקצועיים הגבוהים ביותר האפשריים.

**שיפוט** : חברתנו שומרת על יושרה ועצמאות בשיפוט המקצועי שלה.

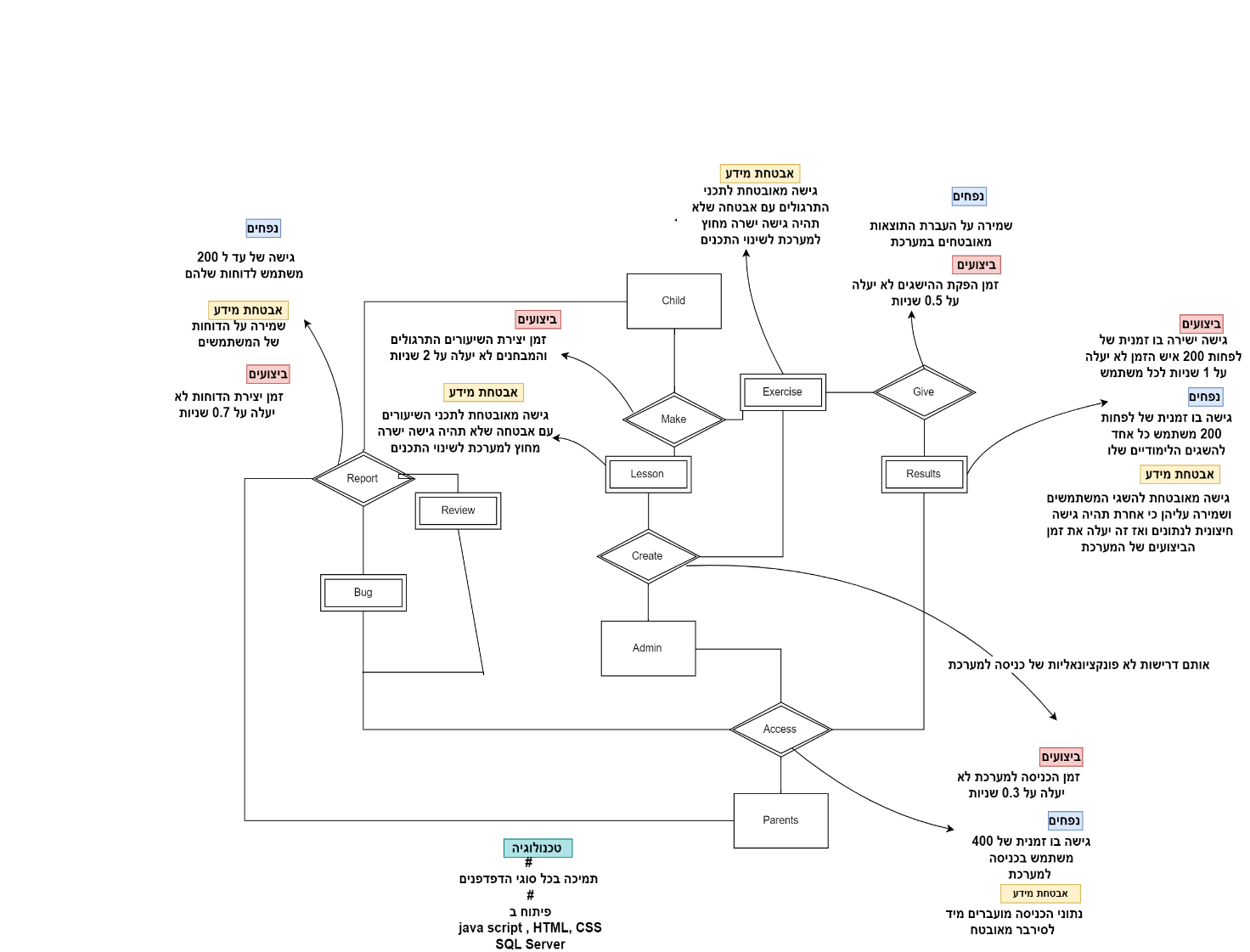
**ניהול** : השגל המנהלי של חברתנו ישמור על גישה אתית לניהול אתר האינטרנט, פיתוחו ותחזוקתו

**מקצועיות** : חברתנו תקדם את היושרה והמוניטין של האתר בהתאמה לצרכי ולטובת ציבור המשתמשים שלו .

**עמיתים**: מהנדסי חברתנו יהיו הוגנים ותומכים בעמיתיהם

**עצמי** : מהנדסי חברתנו ישתתפו לאורך חייהם המקצועיים בלימוד הפרקטיקה של האתר וינהגו בגישה אתית לפרקטיקה של האתר.

**חלק B :**



**חלק C :**   
בתהליך כתבית הייזום והאפיון התחלקנו לשתי קבוצות :  
קבוצה 1 (איתן ודוד ) שהכינו וכתבו את מסמך האפיון   
קבוצה 2 (סאמר ודן) שהכינו וכתבו את מסמך הייזום

בתהליך בניית התרשימים:

קבוצה 1 (סאמר ודוד) תיכננו, הכינו ובנו את תרשימי ה DFD והעלו אותם GITHUB

קבוצה 2 (דן ואיתן) תכננו, הכינו ובנו את תרשים ה ERD ושיתפו את זה   
בסוף דוד העלה להגשת את שלושת התרשימים

בהאקתון :

התחלקנו ל 2 קבוצות בהתחלה :  
מקרה1:  
קבוצה מס'1 (דוד, איתן, דן): חיפשה בסיס נתונים מתאים לעבודה ולבניית האתר והתחילו לממש את 4 הדרישות שבחרו ממיסטרטאסק  
קבוצה ב (סאמר) : התחיל בחלק השלישי של דרישות ההאקתון   
מקרה2 : כשסאמר סיים את כתיבת החלק השלישי הוא הצטרף לקבוצה   
דן וסאמר עבדו על בניית ועיצוב הדף (front end), דוד ואיתן עבדו על בסיס הנתונים ( back end)

מקרה 3 : דוד ואיתן התחילו בלעבוד על יצירת אובייקט של בסיס נתונים   
ולאחר שיתוף קבצים שכל אחד ממנו הכין ב github, את מה שיש לצרף ביחד צירפנו.  
  
בשלושת המקרים הנ"ל ה programing mob נתן לנו לחלק את כל העבודה לחלקים ושכל קבוצה תעבוד על חלק בודד בהנחה שהחלק שהם לא ממשים תקין לחלוטין ואז צירפנו את החלקים ביחד   
זה חסך זמן רב ועבודה אדירה, עשה סדר מצוין (70% ביצוע לעומת אם לא היינו משתמשים בתהליך זה )

**חלק D:**

בדיקת יחידות היא טכניקה המיושמת על ידי מפתחי תוכנה בודדים בכדי להבטיח כי חלקי הקוד הקטנים והעצמאיים מתפקדים כפי שתוכננו ולספק את התוצאות הנכונות. מכיוון שבדיקת יחידות ידניות היא אינטנסיבית בזמן ומאמץ, קיימים כלים רבים להפעלת בדיקות יחידה באופן אוטומטי על בסיס אלמנטים עיצוביים המקודדים ישירות במודול קוד. כלי בנייה / אינטגרציה רציפים מבטיחים כי קוד נבדק ביחידות ללא כשלים לפני ביצוע הצ'ק-אין [[1]](#footnote-1) .  
  
חסרון גדול בטכניקה זו היא שקשה ליצור סביבת בדיקה רלוונטית למודלים המפעילים אינטראקציה עם מצב מערכת מורכב או עם סביבת עבודה מורכבת  
למשל בעיית ה (Q3.2) נבחרה על ידי חברות רבות כאשור הבעייתי ביותר בבדיקות יחדיה. [[2]](#footnote-2)

1. <https://link-springer-com.ezproxy.sce.ac.il/chapter/10.1007/978-1-4842-4206-3_70> [↑](#footnote-ref-1)
2. # A survey of unit testing practices , subject : Unit Testing Weaknesses, IEEE <https://ieeexplore-ieee-org.ezproxy.sce.ac.il/abstract/document/1657935>

   [↑](#footnote-ref-2)