

**שם בית הספר:** ישיבה תיכונית חספין

**שם פרויקט**:BeThere

**שם התלמיד:** אברהם יצחק

**ת.ז התלמיד :** 216799296

**שם המנחה:** חגי סוויד

**שם החלופה:** תכנות טלפונים חכמים

**תאריך הגשה:** 13.6.2025



**תוכן עניינים**

**מבוא**

**הרקע לפרויקט**

**שם הפרויקט:**

BeThere

**תיאור קצר של הפרויקט:**

הפרויקט מהוה אפליקציה שמאפשרת סימון נוכחות למדריכי פנימייה בצורה פשוטה ובסיסית, ומעקב על ידי המורים ומנהל הפנימיה.

באמצעות האפליקציה המדריכים יכולים לבדוק מי נוכח בכל ערב ולדאוג לבטיחותם של התלמידים בצורה נוחה ומהירה, המורה יכול להתעדכן מי לא שמר על חוקי הפנימיה או לאפשר היעדרות מהפנימייה, ומנהל הפנימיה יכול לוודא שכל התלמידים נוכחים.

האפליקציה כוללת התחברות והרשמה לפי סוג משתמש (מדריך\מורה\מנהל פנימייה).

כל הנתונים נשמרים ומנוהלים על ידי Firebase – כולל מדריכים, מורים, וכיתות.

**קהל היעד של הפרויקט:**

קהל היעד של האפליקציה הוא פנימיות\מוסדות שמאפשרים לינה.

הפנימיות יכולים להיעזר באפליקציה בשלושה סוגים עיקריים – מדריכי פנימייה, מורים, מנהלי פנימייה.

מדריכי פנימייה שאחראים על נוכחות, משמעת, ציוד אישי (טלפונים וכו'), צריכים כלי פשוט ונגיש לדיווח מהיר בסוף כל ערב.

מורים נעזרים באפליקציה כדי לזהות תלמידים שנעדרים לעיתים קרובות או לעדכן את המדריך באמצעות האפליקציה על היעדרות מאושרת של תלמיד.

מנהלי פנימייה נעזרים באפליקציה מכיוון שהם זקוקים לתמונה כוללת על נוכחות ומשמעת, ומעוניינים ביכולת לזהות דפוסים ובעיות רוחביות בזמן אמת.

**הסיבות לבחירת הנושא:**

הסיבה העיקרית שגרמה לי לבחור בנושא הנוכחות בפנימיה לאפליקציה שלי, היא התפקיד שלי כמדריך בישיבה שבה אני לומד, והחוסר בדרך סימון נוחה ופשוטה שמתעדכנת כל 24 שעות.

הרגשתי רצון לעזור לבאים אחריי, להנגיש להם את העבודה כמדריך פנימיה, ולהקל עליהם את המלאכה העיקרית כמדריך פנימיה – סימון נוכחות ודאגה לביטחונם של כלל התלמידים מבלי לפספס אחד בטעות (בעקבות מקרי עבר שקרו לתלמידים שהמדריכים שכחו לבדוק איפה הם).

**תהליך המחקר**

**נקודות עיקריות במחקר לצורכי הפרויקט:**

במהלך העבודה על האפליקציה, נדרשתי ללמוד תחומים רבים שלא הכרתי מראש. כדי להבין וליישם בצורה נכונה את הכלים והטכנולוגיות, נעזרתי באתרים ובמקורות מידע שונים.

אלו היו האתרים המרכזיים שהשתמשתי בהם בעבודה על הפרויקט:

- Stack Overflowפורום של מפתחים מכל העולם שבו מצאתי תשובות לשאלות רבות, ובעיקר לשגיאות ובעיות שעלו תוך כדי תכנות. חיפשתי שם פתרונות קיימים ולמדתי איך לגשת לקוד בצורה נכונה.

YouTube - צפיתי במדריכים שסייעו לי להבין איך לבנות מסכים, ליצור תרשים UML, לעבוד עם Firebase, ולבצע פעולות כמו התחברות, הרשמה והצגת רשימות. סרטוני ההדגמה עזרו לי ליישם תהליכים בקוד.

ChatGPT - עזר לי ללמוד דברים חדשים ששימשו אותי בעיצוב המסכים, ולהסביר נושאים תכנותיים.

Android Developers - הסביר לי בצורה פשוטה איך לכתוב את הקוד בצורה נכונה, חיפשתי שם פעמים רבות את השגיאות בהם נתקלתי, וקיבלתי תשובה שנתנה לי מענה.

קיימות מספר אפליקציות דומות לאפליקציה אותה אני בניתי:

**משו"ב** – מערכת "משוב" של משרד החינוך היא אפליקציה חינוכית המיועדת לחיזוק הקשר בין בית הספר, ההורים והתלמידים באמצעות שקיפות ובקרה שוטפת. האפליקציה מאפשרת למורים להזין ציונים, נוכחות, הערות התנהגות ושיעורי בית, בעוד הורים ותלמידים יכולים לעקוב בזמן אמת אחר ההישגים הלימודיים, מערכת השעות, התנהלות ומשימות. מדובר בכלי נגיש ודיגיטלי שמטרתו לשפר את המעקב והמעורבות בתהליך החינוכי.

**Priority TimeSheet** - מערכת Priority TimeSheet היא כלי לניהול ודיווח שעות עבודה, המיועדת לארגונים ולעובדים כחלק ממערכת Priority ERP. האפליקציה מאפשרת לעובדים להזין שעות עבודה יומיות או שבועיות, לנהל פרויקטים ומשימות, ולעקוב אחרי ניצול זמן ומשאבים. המנהלים יכולים לאשר דיווחים, לנתח נתוני שעות ולייעל תהליכים תפעוליים. המערכת מתממשקת עם שכר, פרויקטים ותפעול, ומשמשת כפתרון יעיל לדיווח זמין, מדויק ומבוקר מכל מקום.

סקירת המצב הקיים בשוק:

ישנן אפליקציות דומות לאפליקציה שלי, אך הייחודיות של האפליקציה שלי בכך שהיא מיועדת דווקא לפנימיות או למוסדות שמאפשרים לינה, ומאפשרת למנהל הפנימייה מבט ניהולי וכולל של כל הפנימייה. בנוסף באפליקציה שלי הנתונים מתאפסים כל 24 שעות, אבל ישנם מודדים (סוכמים) של הנתונים לכל תלמיד.

האפליקציה BeThere דואגת לביטחונם של התלמידים, ובשונה משאר האפליקציות שמתעסקות בעיקר בסדר היום של התלמיד, האפליקציה BeThere מתמקדת בשעות הלילה ובביטחונו של התלמיד.

**אתגרים מרכזיים**

הפרויקט עונה על הצורך לסמן נוכחות בפנימיה (למדריכים שאינם חלק ממערכת של משרד החינוך ולכן אינם יכולים להשתמש במשו"ב), בצורה קלה ומהירה דרך אפליקציה נוחה לשימוש. פיתוח האפליקציה דרש ממני להתמודד עם שני אתגרים עיקריים : עבודה עם Firebase, הפרדה בין סוגי המשתמשים (מדריך\מורה\מנהל פנימיה).

1. **עבודה עם Firebase -**

לפני תחילת הפרויקט, לא היה לי הרבה ניסיון בעבודה עם מסדי נתונים, ובפרט לא עם Firebase. נדרשתי להבין וללמוד כיצד שומרים מידע אונליין בענן, ואיך מקשרים בין ממשק המשתמש לבין הנתונים האלו - כל זאת באמצעות קוד.

במהלך העבודה למדתי לבצע מספר פעולות עיקריות באמצעות Firebase –

יצירת משתמשים חדשים באמצעות signinWithEmailAndPassword ובדיקת הסוג שנבחר (מדריך או מורה או מנהל פנימייה), שליפת נתונים והוספת נתונים, ועדכון נתונים. הפרויקט כולו תלוי בFirebase ולכן יש הרבה עבודה איתו.

על מנת לבצע את כל הפעולות האלו, ולוודא שהקוד מתקשר באופן חלק עם מסד הנתונים, היה עליי לחקור וללמוד על Firebase, להבין כיצד עובד מסד הנתונים ואילו פעולות ניתן לבצע בעזרתו.

1. **הפרדה בין סוגי משתמשים – מורה, מדריך, מנהל פנימייה –**

הפרויקט שלי דרש הפרדה בין שני סוגי משתמשים שונים – עם תפקידים ופעולות שונות לגמרי. זה דרש ממני לבנות שלוש חוויות משתמש שונות מקבילות באפליקציה, שבנויות על קוד משותף אבל פועלות באופן שונה. מדריך יכול לערוך נתונים מסויימים שהמורה לא יכול, המורה יכול לערוך נתונים מסויימים שהמדריך לא יכול, ומנהל הפנימיה לא יכול בכלל לערוך נתונים, יש לו רק תצוגת צופה.

במסך הכניסה נדרש המשתמש להכניס כתובת מייל – לפיה נכנס למסכים שונים אם הוא מורה\מדריך\מנהל פנימיה. כדי לשמור על הפרדה מסודרת, השתמשתי במחלקות שונות (Student, Guide , Teacher , Maneger) עם מבנה נתונים מותאם. היה עליי לשמור על סדר והפרדה כדי למנוע בלבול בין שתי חוויות המשתמש.

**תיאור תחום הידע – הפרק המילולי**

**אובייקטים נחוצים, ייצוגם ופעולות על המידע שלהם**

**אובייקטים נחוצים**

1. **Student – (תלמיד):**

אובייקט המייצג פרטי משתמש של תלמיד.

תכונות:

name – שמו של התלמיד. טיפוס:String.

isPresent – האם התלמיד נוכח. טיפוס: Boolean.

isLate – האם התלמיד איחר. טיפוס: Boolean.

isInHome – האם התלמיד בבית באישור. טיפוס:Long.(מכיוון שFirebase לא הצליח להכיל כמות גדולה של Boolean עשיתי מטיפוס Long).

isPhone – האם התלמיד הביא טלפון. טיפוס: Boolean.

viewType – מאיזה זווית מסתכלים (מורה או מדריך). טיפוס:Long.

Id – כתובת ייחודית לתלמיד. טיפוס:String.

count - סוכם איחורים של התלמיד. טיפוס:Long.

1. **Guide (מדריך פנימיה):**

אובייקט המייצג פרטי משתמש של מדריך.

תכונות:

name – שמו של המדריך. טיפוס:String.

Id – כתובת ייחודית למדריך. טיפוס:String.

ArrayList<Student>students – רשימה של תלמידים של המדריך. טיפוס: ArrayList<Student>.

Teacher- אובייקט מסוג Teacher שמייצג את המורה שעובד במקביל עם המדריך.

1. **Teacher (מורה):**

אובייקט המייצג פרטי משתמש של מורה.

תכונות:

name – שמו של המורה. טיפוס:String.

Id – כתובת ייחודית למורה. טיפוס:String.

ArrayList<Student>students – מערך של תלמידים של המורה. טיפוס: ArrayList<Student>.

Guide – אובייקט מסוג Guide שמייצג את המדריך שעובד במקביל עם המורה. טיפוס:Guide.

1. **Manager (מנהל פנימיה):**

אובייקט המייצג פרטי משתמש של מנהל פנימיה.

תכונות:

name – שמו של מנהל הפנימיה. טיפוס:String.

Id – כתובת ייחודית למנהל הפנימיה. טיפוס:String.

ArrayList<Guide>guides – רשימה של כל המדריכים שתחת מנהל הפנימיה. טיפוס: ArrayList<Guide>.

**סוגי נתונים**

String - לייצג טקסט (שמות, זהויות ייחודיות).

Long – לייצג מספרים (כמות איחורים)

ArrayList<T> - לייצג רשימות(מורים, תלמידים).

Boolean – לייצג תנאים (האם התלמיד נוכח, האם הביא טלפון, האם איחר).

**ייצוג מידע:**

**תלמיד(Student):**

אובייקט עם שם, מזהה ייחודי, פרטים הקשורים להתנהגות התלמיד, וסוכם אירועי התנהגות של התלמיד.

מאפשר לשמור את פרטי התלמיד בצורה נוחה ונגישה.

**מדריך (Guide):**

אובייקט עם שם, מזהה ייחודי, רשימת תלמידים ששייכים למדריך, כולל בתוכו אובייקט מסוג מורה של המורה שעובד איתו.

מאפשר לערוך את נתוני התלמידים בצורה נוחה ונגישה.

**מורה(Teacher):**

אובייקט עם שם, מזהה ייחודי, רשימת תלמידים ששייכים למורה, כולל בתוכו אובייקט מסוג מדריך של המדריך שעובד איתו.

מאפשר לערוך את נתוני התלמידים בצורה נוחה ונגישה ולנהל מעקב על התלמידים.

**מנהל פנימיה(Manager):**

אובייקט עם שם, מזהה ייחודי, רשימת מדריכים שעובדים תחתיו.

מאפשר לנהל מעקב אחרי כלל המדריכים בצורה נוחה ונגישה.

**תיאור פעולות על המידע:**

1. **הוספה:**

הוספת משתמש תלמיד חדש:

הזנת הפרטים שפירטתי למעלה, יצירת מזהים ייחודים, יצירת אובייקט חדש ושמירתו בFirebase.

1. **עדכון:**

משתמש מדריך יכול לעדכן את פרטי התלמיד (האם נוכח, האם הביא טלפון, האם איחר), ולשנות את הפרטים של התלמיד (חוץ מהשם ומהמזהה הייחודי).

עדכון רשימת תלמידים ופרטיהם בכל הוספה של תלמיד ובכל שינוי פרטים שהמדריך מבצע.

1. **קריאה:**

שליפת נתוני תלמידים\נתוני מדריכים, שליחה חזרה ל Firebase**.**

**עץ תרשים זרימת מסכים**

LoginActivity

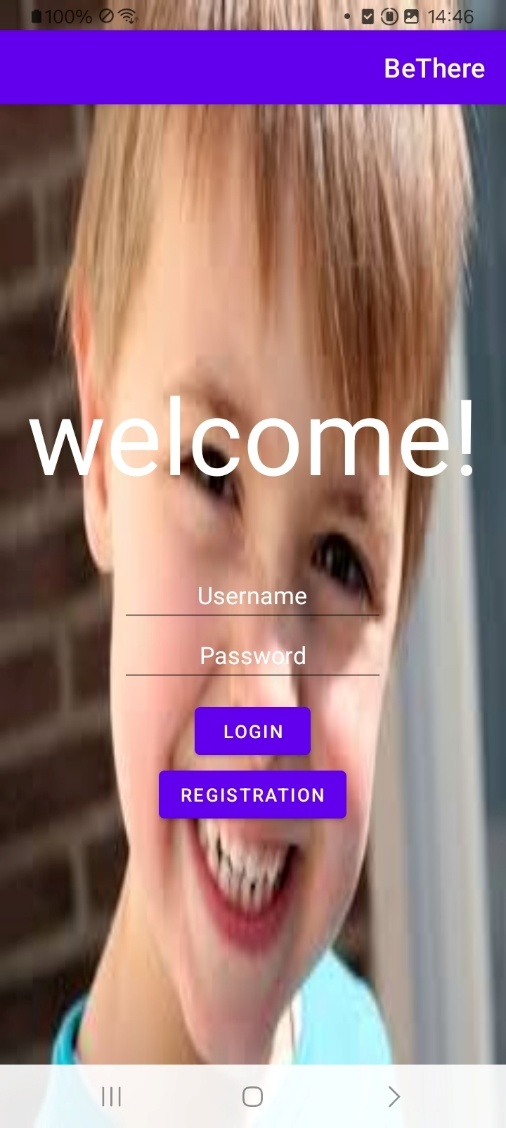
ManagerActivity

GuideActivity

**ארכיטקטורת הפרויקט**

**שם המסך: LoginActivity**

מסך ראשי, מאפשר כניסה והרשמה לאפליקציה. המשתמש בוחר האם לבצע Sign או registration. המשתמש מכניס Email, password.



Button

כפתור ההרשמה

Button

כפתור ההתחברות

TextView

הודעת כניסה

EditText

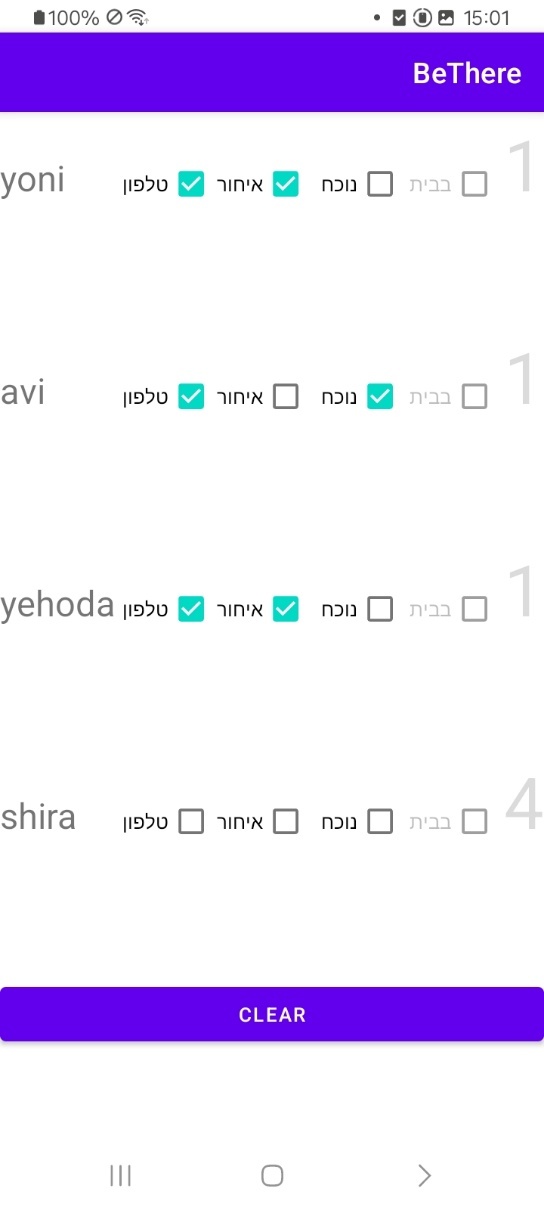
קלט הסיסמה

EditText

קלט האימייל

שם המסך: GuideActivity

מסך שבו מופיעים רשימת התלמידים ונתוניהם.



RecycleView

רשימת תלמידים

Button

ניקוי של הנתונים

TextView

סוכם איחורים

CheckBox

האם התלמיד בבית

CheckBox

האם התלמיד נוכח

CheckBox

האם התלמיד איחר

CheckBox

האם התלמיד הביא טלפון

TextView

שם התלמיד

מימוש הפרויקט

שם המחלקה: LoginActivity

מטרת המחלקה:

מסך ראשי, מאפשר כניסה והרשמה לאפליקציה. המשתמש בוחר האם לבצע Sign או registration. המשתמש מכניס Email, password.

תכונות המחלקה:

|  |  |  |  |
| --- | --- | --- | --- |
| שם התכונה | טיפוס | תחום הכרה | הסבר |
| etname | EditText | private | אובייקט בו המשתמש מכניס את  האימייל שלו על מנת להירשם  למאגר הנתונים. |
| etpassword | EditText | Private | אובייקט בו המשתמש מכניס את  הסיסמה שלו על מנת להירשם  למאגר הנתונים. |
| bRegistration | Button | Private | קולט את פרטי המשתמש ויוצר  user חדש בבסיס הנתונים. |
| bsubmit | Button | private | בודק האם הuser רשום כבר, אם כן מעביר למסך הבא. |
| auth | FirebaseAuth | private | אובייקט המאפשר גישה למאגר  הנתונים של הפרויקט השמור  בfirebase . |

פעולות המחלקה:

|  |  |
| --- | --- |
| שם הפעולה | הסבר |
| onCreate | קישור לבסיס נתונים מרוחק, קישור למסך הxml המתאים,  קישור האובייקטים, מפעיל בתוכו את פעולות onClick |
| onClick | מפעיל את הפעולה registration() כאשר לוחצים על הכפתור registration |
| onClick | מפעיל את הפעולה sign() כאשר לוחצים על הכפתור login |
| registrtaion | הקוד מנסה **לרשום משתמש חדש** (באמצעות אימייל וסיסמה) ב־Firebase. אם הרישום **הצליח**, מוצגת הודעת "Authentication success". אם הוא **נכשל**, מוצגת הודעת "Authentication failed". |
| sign | ביצוע התחברות באמצעות Firebase לפי אימייל  וסיסמה, בדיקת האם המשתמש קיים, ומעביר למסך GuideActivity. |

שם המחלקה: GuideActivity

מטרת המחלקה:

מסך שבו מופיעים רשימת התלמידים ונתוניהם, ניתן לערוך\לצפות בנתוני התלמידים במסך זה.

תכונות המחלקה:

|  |  |  |  |
| --- | --- | --- | --- |
| שם התכונה | טיפוס | תחום הכרה | הסבר |
| rcShowStudents | RecycleView | private | רכיב תצוגה שמציג את רשימת  התלמידים השייכים למשתמש המחובר.(מורה\תלמיד) |
| auth | FirebaseAuth | Private | אובייקט המאפשר גישה למאגר  הנתונים של הפרויקט השמור  בfirebase . |
| students | ArrayList<Student> | Private | רכיב פנימי שמזין את אובייקט הרשימה (RecycleView) - רשימת התלמידים ופרטיהם |
| Bclear | Button | Private | לחצן שמפעיל את פעולת clear |

פעולות המחלקה:

|  |  |
| --- | --- |
| שם הפעולה | הסבר |
| onCreate | קישור לבסיס נתונים מרוחק, קישור למסך הxml המתאים,  קישור האובייקטים, יצירת אירועים לאובייקטים נבחרים, מפעיל את פעולת start |
| Start | יוצרת רשימה ריקה של תלמידים, שולף מבסיס הנתונים את נתוני התלמידים ומוסיף את התלמידים לרשימה, מפעיל בסוף את פעולת createList |
| Clear | מאפסת את הערך של כל השדות של התלמידים (לא מוחקת) |
| createList | מעדכנת בכל לחיצה את מסד הנתונים, ושולחת כל לחיצה על כפתור רשימה מעודכנת של פרטי התלמידים |

שם המחלקה:

Student

מטרת המחלקה: לייצג אובייקט של תלמיד – כולל כל פרטיו.

תכונות המחלקה:

|  |  |  |  |
| --- | --- | --- | --- |
| שם התכונה | טיפוס | תחום הכרה | הסבר |
| Name | String | private | שם התלמיד |
| isPresent | boolean | Private | האם התלמיד נוכח |
| isPhone | boolean | Private | האם התלמיד הביא טלפון |
| isLate | boolean | Private | האם התלמיד איחר |
| Home | Long | Private | האם התלמיד בבית |
| viewType | boolean | Private | זווית הסתכלות – מדריך או מורה |
| Count | Long | Private | סוכם איחורים של התלמיד |
| id | String | private | מזהה ייחודי של התלמיד |

פעולות המחלקה:

|  |  |
| --- | --- |
| שם הפעולה | הסבר |
| Student(String name , String id, boolean isLate , boolean isPresent , boolean isPhone, Long home ,boolean viewType , Long count) | בנאי מלא - יוצר אובייקט student עם כל הנתונים הדרושים. |
| incNum() | מוסיף לcount 1 |

שם המחלקה:

Guide

מטרת המחלקה: לייצג אובייקט של מדריך – כולל כל פרטיו.

תכונות המחלקה:

|  |  |  |  |
| --- | --- | --- | --- |
| שם התכונה | טיפוס | תחום הכרה | הסבר |
| Name | String | private | שם המדריך |
| isFinish | boolean | Private | האם המדריך סיים את עבודתו(כל תלמידי הכיתה נוכחים\בבית) |
| grade | String | Private | שם הכיתה שהמדריך אחראי עליה |
| Id | String | Private | מזהה ייחודי  למדריך |
| students | ArrayList<Student> | private | רשימת תלמידים של המדריך |

פעולות המחלקה:

|  |  |
| --- | --- |
| שם הפעולה | הסבר |
| Guide(String name ,boolean isFinish,String grade, String id ,ArrayList<Student> students ) | פעולה בונה היוצרת אובייקט Guide עם כל הנתונים הדרושים. |

שם המחלקה: Teacher

מטרת המחלקה: לייצג אובייקט של מורה – כולל כל פרטיו.

תכונות המחלקה:

|  |  |  |  |
| --- | --- | --- | --- |
| שם התכונה | טיפוס | תחום הכרה | הסבר |
| Name | String | private | שם המורה |
| Id | String | Private | מזהה ייחודי למורה |
| students | ArrayList<Student> | private | רשימת תלמידים של המורה |
| guide | Guide | Private | המדריך שעובד עם אותו המורה (על אותה הכיתה) |

פעולות המחלקה:

|  |  |
| --- | --- |
| שם הפעולה | הסבר |
| Teacher(String name , String id ,ArrayList<Student> students , Guide guide ) | פעולה בונה היוצרת אובייקט Teacher עם כל הנתונים הדרושים. |

שם המחלקה:Manager

מטרת המחלקה: לייצג אובייקט של מנהל פנימיה עם כל פרטיו.

תכונות המחלקה:

|  |  |  |  |
| --- | --- | --- | --- |
| שם התכונה | טיפוס | תחום הכרה | הסבר |
| Name | String | private | שם המנהל |
| Id | String | Private | מזהה ייחודי למנהל |
| guiders | ArrayList<Guide> | private | רשימת מדריכים של המנהל |

פעולות המחלקה:

|  |  |
| --- | --- |
| שם הפעולה | הסבר |
| Manager(String name , String id , AraayList<Guide>guiders) | פעולה בונה היוצרת אובייקט Manager עם כל הנתונים הדרושים. |

בסיס נתונים

בפרויקט זה השתמשתי בFirebase, בסיס נתונים שיתופי על פי צרכי הפרויקט. בFirebase

השתמשתי בשתי תכונות:

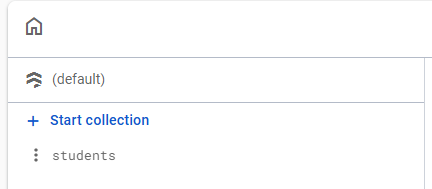
Authentication .1

Firestore Database .2

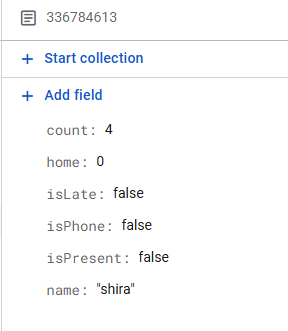
בAuthentication נשמרים אימייל וסיסמה באופן ייחודי (UID מונע כפילויות).

ב – Firestore Database נשמרים יותר פרטים.

האובייקטים המתבקשים בפרוייקט:



פירוט על כל אובייקט:



Count – סוכם איחורים של התלמיד.

Home – האם התלמיד בבית.

isLate - האם התלמיד מאחר

isPhone - האם התלמיד הביא טלפון

isPresent – האם התלמיד נוכח

Name – שם התלמיד

פעולות על / עם בסיס הנתונים:

1. רישום משתמש חדש - הוספת משתמשים חדשים לאפליקציה בעזרת פרטים.
2. התחברות משתמש קיים - אימות פרטי המשתמש כדי לאפשר גישה לחשבון שלו.
3. שליפת נתונים מהמסד - הצגת תלמידים, או נתוני תלמידים שנשמרו במסד הנתונים.
4. הוספת פריט חדש למסד - יצירת תלמיד חדש או הוספת תלמיד לרשימת התלמידים.
5. עדכון מידע קיים - שינוי פרטים בתלמיד\מדריך שכבר נוצר, כמו כמות איחורים או האם התלמיד בבית.

מדריך למשתמש

מסך הכניסה לאפליקציה – לפניך 2 אפשרויות – להירשם לאפליקציה או להתחבר לאפליקציה (אופציה זו מיועדת למשתמש רשום).

אם ברצונך ליצור משתמש חדש – הכנס את כתובת המייל שתבחר ואת הסיסמה שתבחר, לחץ על REGISTRATION , ולאחר מכן לחץ על Login

אם ברצונך להתחבר למשתמש קיים, יש למלא את הפרטים: אימייל וסיסמת המשתמש, ולאחר מכן ללחוץ על לחצן LOGIN.

הערות:

o אם ההרשמה תצליח תקפוץ הודעה "registered Successfully".

o אם ההרשמה לא תצליח תקפוץ הודעה "Error Registration".

**אם התחברת כמדריך:**

לאחר הלחיצה על LOGIN, תועבר באופן אוטומטי למסך המרכזי של המדריך.

במסך זה תראה רשימה של כל התלמידים שתחת אחריותך.

* כל תלמיד מוצג כשורה נפרדת
* בכל שורה מוצגים לפניך פרטי התלמיד (שם , האם נוכח, האם איחר, האם הביא טלפון, האם הוא בבית, כמות האיחורים).

תיבת הסימון "בבית" איננה ניתנת לשינוי למשתמש שלך, רק למשתמש "מורה".

אם תיבת הסימון "בבית" לחוצה כבר (מסומנת בV), לא תוכל לערוך את שאר תיבות הסימון של אותו תלמיד – מכיוון ששאר פרטי התלמיד לא רלוונטים כי הוא לא נוכח בפנימיה.

שאר תיבות הסימון ניתנות לשינוי בכל לחיצה שלך ונשמרות עד ללחיצה הבאה שתשנה את הסימון.

בצד ימין מופיע לך מספר – הוא בעצם כמות האיחורים הקודמים שיש לתלמיד, הסוכם מאפשר מעקב על תלמידים בעייתים בכיתתך.

למטה מופיע כפתור "CLEAR" – כפתור שמנקה את כל השדות (מלבד סוכם האיחורים) – מומלץ ללחוץ על הכפתור כל 24 שעות כדי שהנתונים יתאפסו כל ערב.

כאשר תלחץ על הכפתור – תיפתח חלונית עם אפשרות לבחור האם אתה בטוח שאתה רוצה למחוק, יש אפשרות שכן ויש אפשרות שלא.

המטרה שלך היא למלא את הנתונים של כל התלמידים ולבדוק אם הם שומרים על כללי הפנימיה.

**אם התחברת כמורה:**

לאחר הלחיצה על LOGIN, תועבר באופן אוטומטי למסך המרכזי של המורה.

במסך זה תראה רשימה של כל התלמידים שתחת אחריותך.

* כל תלמיד מוצג כשורה נפרדת
* בכל שורה מוצגים לפניך פרטי התלמיד (שם , האם נוכח, האם איחר, האם הביא טלפון, האם הוא בבית, כמות האיחורים).

תיבות הסימון "איחור", "נוכח", אינם נתונות ללחיצה עבורך, רק למשתמש "מדריך" (כי המורה איננו נוכח בפנימיה בערב).

תיבות הסימון "בבית", "טלפון", ניתנות לשינוי עבורך – מי שקיבל ממך אישור להיות בבית – עליך לסמן את תיבת הסימון "בבית", כדי שהמדריך יידע לא לחפש אותו בערב.

תיבת הסימון "טלפון" – רלוונטית למי שיש אישור להשאיר טלפון בערב – כאשר אתה מאשר לתלמיד להשאיר את הטלפון, עליך ללחוץ על תיבת הסימון "טלפון", כדי שהמדריך יידע לא לקחת לו את הטלפון בערב.

בצד ימין מופיע לך מספר – הוא בעצם כמות האיחורים הקודמים שיש לתלמיד, הסוכם מאפשר מעקב על תלמידים בעייתים בכיתתך.

המסך מאפשר לך לעקוב אחרי תלמידי הכיתה – לדעת מי מקיים את כללי הפנימיה ומי לא.

למטה מופיע כפתור "CLEAR" – כפתור שמנקה את כל השדות (מלבד סוכם האיחורים), הכפתור מיועד למדריך שאחראי על כיתתך.

המטרה שלך היא לעקוב אחרי תלמידי כיתתך, לוודא שהם שומרים על כללי הפנימיה, ולסמן תלמידים עם שיחרורים חריגים.

**אם התחברת כמנהל פנימיה:**

לאחר הלחיצה על LOGIN, תועבר באופן אוטומטי למסך המרכזי של מנהל הפנימיה.

במסך זה תראה רשימה של כל המדריכים שתחת אחריותך.

* כל מדריך מוצג כשורה נפרדת
* בכל שורה מוצגים לפניך פרטי המדריך(שם, כיתה שעליה המדריך אחראי, האם סיים את עבודתו, מערך תלמידים).

המסך מיועד לצפייה בלבד –

מדריך שכל כיתתו נוכחת\בבית תהיה בשורתו תיבת טקסט "Finish!".

אם תלחץ על שמו של המדריך – תוצג לפניך רשימת התלמידים של אותו מדריך.

המטרה שלך היא שכל המדריכים יסיימו לבדוק את הנוכחות של תלמידיהם.

**רפלקציה:**

את האפליקציה שלי בחרתי לפתח סביב נושא שמדבר אליי – הדרכה של פנימיה וניהול. בשנה האחרונה יצא לי להיות מדריך בפנימיה, רציתי לעזור למדריכים שיבואו אחריי שיהיה להם יותר פשוט, יותר נוח ויותר מהיר.

התחלתי את הדרך עם חששות - לא היה לי הרבה ניסיון קודם עם חיבור למסדי נתונים או Firebase. למדתי לבנות מסכים בצורה מסודרת, להפריד בין תפקידים באפליקציה, להאזין לאירועים במסד הנתונים, ולנהל את זרימת המשתמש – מיצירת משתמש, דרך התחברות ועד לשינוי פרטים ומעקב.

לאורך הדרך למדתי גם לתכנן ממשק משתמש ברור - ניסיתי להיכנס לראש של המשתמש ולחשוב מה הוא ירצה לראות בכל שלב, ואיך השימוש באפליקציה יהיה נוח ואינטואיטיבי עבורו. יצרתי היררכיה בין המסכים, והודעות טקסט מותאמות כדי שהמשתמש יבין אם הפעולה הצליחה או נכשלה.

ככל שהתקדמתי בפרויקט, הבנתי שיש בו מורכבות רבה - בעיקר מבחינת הפרדה בין שלושה סוגי משתמשים: מורה מדריך ומנהל פנימיה. נדרשתי לתכנן את האפליקציה כך שכל סוג משתמש יעבור מסכים שונים, יראה רק את התוכן שרלוונטי אליו, ויקבל חוויית שימוש מותאמת.

לדוגמה: מורה יכול לעדכן את תיבת הסימון "בבית", ולעומתו המדריך יכול לסמן תיבות סימון אחרות אבל לא את תיבת הסימון "בבית" – כי אין לו סמכות לאשר שחרורים הבייתה.

במהלך העבודה גיליתי כמה חשוב לשאול שאלות, לבדוק שגיאות, ולנסות שוב. אחד הלקחים המרכזיים שלמדתי הוא שכל בעיה פותרים בצעדים קטנים, ובסבלנות. בהמשך, **אולי הייתי מוסיף פיצ'רים נוספים לאפליקציה, מנגיש את המסכים ומעצב אותם בצורה שמושכת את העין. הייתי מוסיף אפשרות להיכנס כתלמיד ולצפות בנתונים של עצמך. הייתי מוסיף סוכמים של נוכחות ושל הבאת טלפון, יוצר ממוצעים ושעון זז ברקע.**

לסיכום, הפרויקט הזה היה בשבילי תהליך מאתגר אך משמעותי, חוויה של למידה, יצירה והתמודדות. אני גאה במה שהצלחתי לבנות, ומרגיש שצמחתי ממנו גם ברמה המקצועית וגם ברמה האישית, והכי חשוב - נהניתי.

**ביבליוגרפיה**:

- אתר Stackoverflow

- אתר Geeksforgeeks

- Chat gpt

Youtube –

**נספחים**

**קטעי הקוד**