



# אפליקציית Habit Tracker



מסמך תכנון

## תוכן עניינים

מבנה המערכת

תיאור בסיס הנתונים

- תיאור טבלאות

- תיאור הקשרים

תיאור המחלקות השונות

- קבצי xml

- קבצי java

- חבילת dataconnections

- חבילת utils

- חבילת drawer

- חבילת mainscreen

- חבילת settingscreen

- חבילת users

- תיאור קשרים בין מחלקות

## מבנה המערכת:

Habit Tracker היא אפליקציית אנדרואיד הכוללת פניות לשרת עם קישור לבסיס נתונים.

מטרת האפליקציה היא לאפשר למשתמש לעקוב אחרי ביצוע הרגלים יומיומיים/שבועיים בעזרת רשימת משימות אישית. (ראו מסמך תיאור פונקציונלי לפירוט על מטרת האפליקציה והשימוש בה).

המערכת בנויה משלוש שכבות:

**שכבת התצוגה** היא השכבה העליונה של היישום. היא אחראית על הצגת המידע למשתמש וכוללת בעיקרה קיבצי xml. קבצי הxml נקראים ע"י מחלקות מסוג activity או adapter משכבת היישום ובכך מונגשים למשתמש.

**שכבת היישום** היא השכבה האחראית לקשר בין שכבת התצוגה לשכבת הנתונים, בנוסף לפונקציונליות של היישום. רוב מחלקות ה JAVA ביישום שייכות לשכבה זו.

**שכבת הנתונים** היא השכבה התחתונה של היישום הבנויה משרת בסיס נתונים MySQL המחזיק את בסיס הנתונים ממנו מגיע המידע על פעולות המשתמש ביישום. התקשורת מול שכבה זו מתבצעת ע"י מחלקות connection הנמצאות בשכבת היישום.

## טכנולוגיות בשימוש המערכת

האפליקציה Habit Tracker משלבת בתוכה מספר טכנולוגיות:

- Android
- MySQL Server – שרת בסיס נתונים
- JDBC – קישור לבסיס נתונים
- XML

## תיאור בסיס הנתונים:

בסיס הנתונים בנוי משלושת הטבלאות הבאות:

### users - טבלת משתמשים

הטבלה שומרת מידע על כל המשתמשים במערכת, ויוצרת לכל משתמש מפתח ייחודי. כל שורה מורכבת מהשדות שם משתמש, סיסמא, אימייל ושדה המפתח.

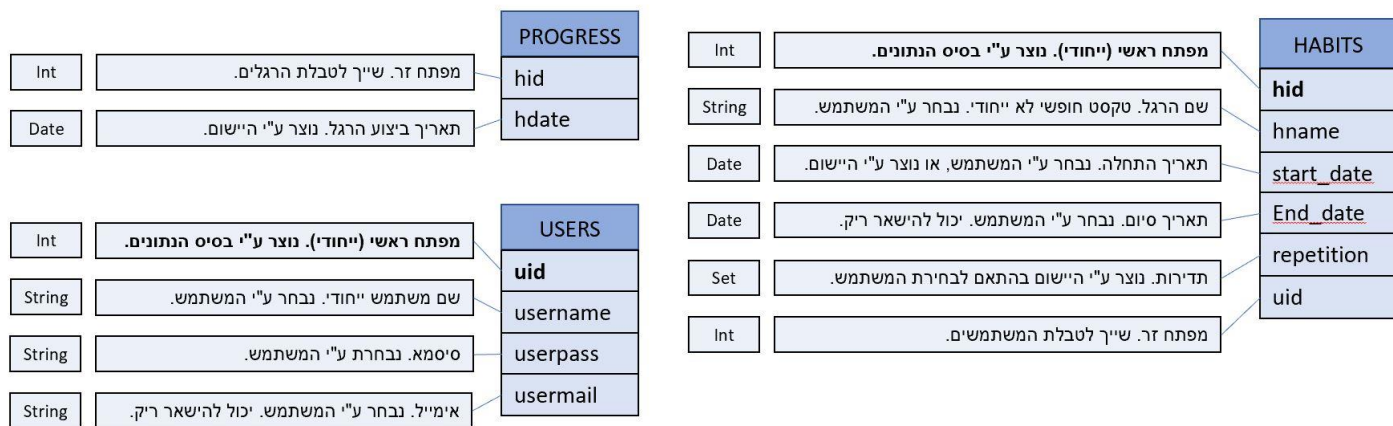
### habits - טבלת ההרגלים

הטבלה שומרת את המידע על כל ההרגלים של כל המשתמשים, ויוצרת לכל הרגל מפתח ייחודי. כל שורה בטבלה מכילה שם, שהוא טקסט חופשי הנבחר ע"י המשתמש, תאריך התחלה, תאריך סיום, תדירות ההרגל (באיזה תדירות יש לבצע אותו), שדה המפתח של ההרגל ושדה המפתח של המשתמש אשר יצר את ההרגל.

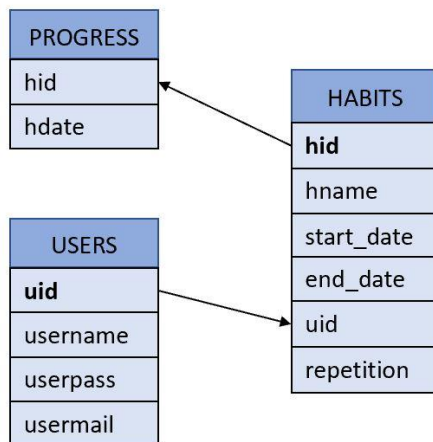
### progress – טבלת הביצועים

טבלה זו יוצרת רשומה עבור כל הרגל שמשתמש מסמן כמבוצע. כך, ובעזרת טבלת ההרגלים, ניתן לעקוב אחר ביצוע או אי-ביצוע של הרגל ע"י משתמש, בתווך מידי או רחוק, ולספק למשתמש פידבק בהתאם. כל שורה בטבלה מכילה שדה מפתח של ההרגל שהתבצע, ותאריך בו אותו הרגל התבצע.

## להלן תרשים לייצוג השדות של כל טבלה:



## להלן תרשים לייצוג הקשרים בין הטבלאות:



## תיאור המחלקות השונות:

כדי לשמור על סדר ובהירות הקוד, המחלקות השונות של היישום מחולקות לחבילות (packages) בהתאם למטרתם ביישום. נסתכל על החבילות השונות:

### קבצי xml:

ממשק המשתמש בנוי מקבצי xml.

בתוך תיקיית המשאבים (res), קיימות 4 חבילות האחראיות על ממשק המשתמש, ביניהן 3 מכילות קבצי xml :

**values** היא חבילה המכילה את הקבצים themes.xml, color.xml, strings.xml אשר קובעים הגדרות כלליות של נראות האפליקציה.

**drawable** היא חבילה המכילה קבצי xml המגדירים את כל האייקונים שמופיעים באפליקציה. בנוסף, תיקייה זו מכילה קובץ progress.xml אשר מגדיר נראות של אובייקט מסוג progressBar.

**\*\*** נציין כי שינוי נראות אובייקט זה אפשרי גם דרך חבילת layout אך שימוש מסוג זה אינו נתמך ע"י חלק מהמכשירים שנכללים בטווח המכשירים בו היישום תומך.

**layout** היא חבילה המכילה קבצי xml המגדירים את הנראות של מחלקות שונות. כל קובץ בחבילה זו מגדיר את הנראות של המחלקה אליה הוא שייך ואת האובייקטים הקבועים השייכים לממשק המשתמש של אותה מחלקה.  
נסתכל על הקבצים בחבילה זו:

שם הקובץ	המחלקה אותה הוא מגדיר
activity_calendar.xml	מסך יומן CalanderActivity.
activity_login.xml	מסך התחברות LoginActivity.
activity_main.xml	מסך ראשי MainActivity.
activity_profile.xml	מסך פרופיל ProfileActivity.
activity_register.xml	מסך יצירת משתמש חדש RegisterActivity.
activity_reset_password.xml	מסך איפוס סיסמא ResetPasswordActivity.
activity_settings.xml	מסך הגדרות SettingsActivity.
main_item_layout.xml	אובייקט המייצג משימות יחידות ברשימת המשימות במסך הראשי.
nav_drawer.xml	מגירת התפריט.
nav_toolbar.xml	סרגל התפריט.
profile_email_dialog.xml	דיאלוג עדכון כתובת אימייל במסך פרופיל.
settings_dialog.xml	דיאלוג יצירת הרגל חדש / עדכון הרגל במסך הגדרות.
Settings_item_layout.xml	אובייקט המייצג הרגל יחיד והמשתנים שלו ברשימת ההרגלים במסך ההגדרות.

## קבצי Java:

נבחן את שאר המחלקות באפליקציה לפי החבילות אליהן הן שייכות.

### dataconnections package

חבילה המחזיקה את כל המחלקות האחראיות על הקשר בין היישום לבסיס הנתונים. נסתכל על המחלקות בחבילה זו:

שם הקובץ	תיאור
DbVars.java	מחלקה שמחזיקה קבועים המגדירים את החיבור לשרת בסיס הנתונים. במידה ורוצים לשנות כתובת או משתמש שרת ליישום, יש לשנות את קובץ זה בהתאם.
ConnectionHelper.java	מחלקה מופשטת abstract ממנה יורשות כל שאר המחלקות בחבילה. המחלקה יוצרת חיבור לשרת בסיס הנתונים, מבצעת פעולות על בסיס הנתונים, וסוגרת את החיבור. הקובץ בנוי כך שהמחלקות היורשות צריכות רק לשלוח שאילתת SQL ולממש שיטה המבצעת מניפולציה על תוצאת השאילתה. בצורה זו נוכל לבצע פעולות שונות ומגוונות על תוצאות שאילתת הSQL לפני סגירת החיבור.
FindUserName.java	מחלקה המשמשת את מסך יצירת משתמש לשם בדיקה אם ברגע ההתחברות (בזמן הקריאה למחלקה) שם המשתמש הרצוי פנוי או קיים במערכת.
GetUserAuth.java	מחלקה המשמשת את מסך איפוס הסיסמא לבדיקה אם שם המשתמש והאימייל שהקליד המשתמש תואמים את אחת מהשורות בטבלת המשתמשים ומחזירה הודעה למשתמש בהתאם. (איפוס סיסמא מתבצע ע"י הקלדת אימייל המתאים למשתמש ולאחר אימות מצבע פעולה שונה מזו של התחברות, ולכן אימות זה שונה מאימות לשם התחברות למערכת)
GetLogin.java	מחלקה המשמשת את מסך ההתחברות לבדיקה את שם המשתמש והסיסמא שהקליד המשתמש תואמים לאחת מהשורות בטבלת המשתמשים. המחלקה מחזירה הודעה למשתמש בהתאם, או מבצע התחברות למערכת.
GetUserDetails.java	מחלקה המשמשת את מסך הפרופיל לשם שליפת כל פרטי המשתמש המחובר באותו רגע משרת בסיס הנתונים.
GetAllHabits.java	מחלקה המשמשת את המסך הראשי (רשימת המשימות). המחלקה מתקשרת עם שרת בסיס הנתונים לשליפת כל ההרגלים המתאימים ליום המתבקש, ויוצרת את רשימת ההרגלים בהתאם, כך שהמחלקה האחראית על המסך הראשי תוכל להציג אותם.
GetHabitSettings.java	מחלקה המשמשת את מסך ההגדרות (הגדרות הרגלים) לשליפת כל ההרגלים של המשתמש והמידע עליהם. המחלקה יוצרת את רשימת ההרגלים כך שמסך ההגדרות יוכל להציג אותם. (רשימה זו מכילה אובייקטים שונים המכילים מידע שונה מאלו שנוצרים במחלקה GetAllHabits. בנוסף לכך, השאילתה הנשלחת לחיבורים שונה מאוד בין שתי המחלקות. נעשתה בחירה ליצור רשימות אלו ע"י מחלקות חיבורים שונות בכדי לקצר את פעולות המחלקה בעת שהחיבור לבסיס הנתונים פתוח)

מחלקה המשמשת את מסך ההגדרות (הגדרות הרגלים). בעת יצירת הרגל חדש, המפתח של ההרגל החדש נוצר ע"י שרת בסיס הנתונים בהתאם למפתחות ההרגלים הקיימים המערכת באותו רגע. מחלקה זו משמשת לשליפת המפתח של ההרגל האחרון שנוצר ע"י המשתמש, בכדי לאפשר עריכה של הרגל זה מבלי לשלוח מחדש את כל ההרגלים מבסיס הנתונים (מבלי ליצור מחדש את מסך ההרגלים).	GetNewHid.java
מחלקה המשמשת לביצוע מניפולציות על בסיס הנתונים (update, insert, delete...). כלומר, מתבצע שימוש במחלקה זו כאשר יש צורך בשליחת שאילתות לשרת בסיס הנתונים שאין להן תוצאה או שהתוצאה שלהן אינה רלוונטית.	UpdateDb.java

## utils package

חבילה המחזיקה את כל המחלקות הנועדות לשימוש גלובלי, אשר אחראיות על מניפולציה של טקסט, זמן, התראות וכדומה. נסתכל על המחלקות בחבילה זו:

שם הקובץ	תיאור
DateManipulations.java	מחלקה המכילה שיטות סטטיות לשימוש בתאריכים באפליקציה. בין השיטות שקיימות במחלקה שיטות להמרת תאריך לטקסט, טקסט לתאריכים, שינוי פורמט תצוגה של תאריך ועוד.
WarningManager.java	מחלקה האחראית על התראות של תקלות נפוצות ביישום. בנוסף לפונקציות, המחלקה מכילה גם משתני String קבועים לייצוג התראות, כך שהתראות שחוזרות על עצמן יקבלו התראה אחידה מכל מקום ביישום.
DialogPopUp.java	מחלקה בעלת שימוש גלובלי ליצירת דיאלוג התראה.
ViewManager.java	מחלקה בעלת שימוש גלובלי לביצוע פעולות נפוצות על אובייקטים מסוג view. השיטות במחלקה מנהלות שינויים בקלט משתמש ומעדכנות התראות על קלט לא תקין.

## drawer package

חבילה האחראית על מגירת התפריט ועל כל הקשור למעבר בין תצוגות ביישום. נסתכל על המחלקות בחבילה זו:

שם הקובץ	תיאור
DrawerMethods.java	ממשק אותו מממשות כל המחלקות המכילות תפריט. פונקציות מעבר עמודים של התפריט מוגדרות בתוך המחלקות המכילות אותו. תפקיד ממשק זה לוודא שכל מחלקה המכילה תפריט תממש את כל השיטות הנדרשות למגירת התפריט.
DrawerManager.java	מחלקה המכילה שיטות סטטיות לפעולות על מגירת התפריט.
IntentManager.java	מחלקה המכילה שיטות סטטיות לפעולות של אובייקטים מסוג Intent. המחלקה מממשת שיטות המאפשרות מעבר עמוד (Activity), שליחת מידע לעמוד הבא ושליפת מידע מהעמוד הקודם.

מימוש של מסך במערכת מתבצע ע"י מחלקה מסוג AppCompatActivity.

כל אחת מהחבילות הבאות מכילה תת-חבילה בשם activities, אשר מכילה את מחלקות אלה.

לכל מסך שנראה באפליקציה קיימת מחלקת activity המייצגת אותו. נראה מחלקות אלו בחבילות הבאות:

## mainscreen package

חבילה האחראית על המסך הראשי של האפליקציה והפעולות המתבצעות בו. נסתכל על המחלקות בחבילה זו:

שם הקובץ	תיאור
MainActivity.java	שייכת לתת חבילה activities. מחלקה האחראית למסך הראשי וכל מה שכלול בה. המחלקה מחברת בין הקוד לקובץ הxml המגדיר את נראות המסך, מגדירה את הלוגיקה של המסך והאובייקטים שבו, ומתפעלת את המחלקות המתקשרות עם בסיס הנתונים.
ClendarActivity.java	שייכת לתת חבילה activities. מחלקה זו אחראית על מסך בחירת תאריך לתצוגה במסך הראשי ע"י בחירת תאריך מתצוגת יומן. המחלקה מחברת בין הקוד לקובץ הxml המגדיר את נראות היומן.
ProgressBarAnimation.java	מחלקה זו אחראית על נראות מד ההתקדמות (progressBar). המחלקה מנהלת את האנימציה של מד ההתקדמות ומעדכנת את ממשק המשתמש בשינויים בו.
MainItem.java	מחלקה המנהלת אובייקט משימה. רשימת המשימות במסך הראשי מורכבת מאובייקטים אינדיבידואלים המייצגים משימות. כל משימה (הרגל) מיוצגת ע"י אובייקט ממחלקה זו.
MainAdapter.java	מחלקה זו אחראית על ניהול כל מה שקשור לאובייקט משימה יחיד. המחלקה מחברת בין אובייקט משימה (MainItem) לבין קובץ הxml המגדיר את הנראות שלו, מגדירה את הלוגיקה של האובייקט ומחברת אותו למסך הראשי המכיל את רשימת המשימות.

## settingscreen package

חבילה האחראית על מסך ההגדרות של האפליקציה (My Habits) והפעולות המתבצעות בו. נסתכל על המחלקות בחבילה זו:

שם הקובץ	תיאור
SettingsActivity.java	שייכת לתת חבילה activities. מחלקה האחראית למסך ההגדרות וכל מה שכלול בה. המחלקה מחברת בין הקוד לקובץ הxml המגדיר את נראות המסך, מגדירה את הלוגיקה של המסך והאובייקטים שבו, ומתפעלת את המחלקות המתקשרות עם בסיס הנתונים.



מחלקה המנהלת אובייקט הגדרות של הרגל, נקרא לו אובייקט הרגל. רשימת ההרגלים במסך ההגדרות מורכבת מאובייקטים אינדיבידואלים המייצגים הרגלים. כל הרגל ברשימה מיוצג ע"י אובייקט ממחלקה זו.	SettingsItem.java
מחלקה זו אחראית על ניהול כל מה שקשור לאובייקט הרגל יחיד. המחלקה מחברת בין אובייקט הרגל (SettingsItem) לבין קובץ xml המגדיר את הנראות שלו, מגדירה את הלוגיקה של האובייקט ומחברת אותו למסך ההגדרות המכיל את רשימת ההרגלים.	SettingsAdapter.java
מחלקה זו אחראית על מסך הוספת הרגל חדש/עריכת הרגל. מסך זה אינו מסוג activity אלא מיוצג כדיאלוג. מחלקה זו מחברת בין דיאלוג עריכת הרגל לקובץ xml המגדיר אותו, ומגדירה את הלוגיקה של הדיאלוג והאובייקטים שבו.	SettingsDialog.java
מחלקה זו מכילה שיטות סטטיות המבצעות ולידציה לקלט המגיע מהמשתמש בעת יצירת הרגל חדש או עריכת הרגל. השיטות במחלקה מבצעות ולידציה לשדות השונים של דיאלוג עריכת הרגל ומחזירות String המתריע במידה וקיימת בעיה בקלט (או NULL אם הקלט תקין).	SettingsValidation.java

## users package

חבילה האחראית על כל המסכים שמתעסקים בניהול משתמש. החבילה אחראית על מסך ההתחברות, הרשמה, איפוס סיסמא או אימייל ומסך הפרופיל. נסתכל על המחלקות בחבילה זו:

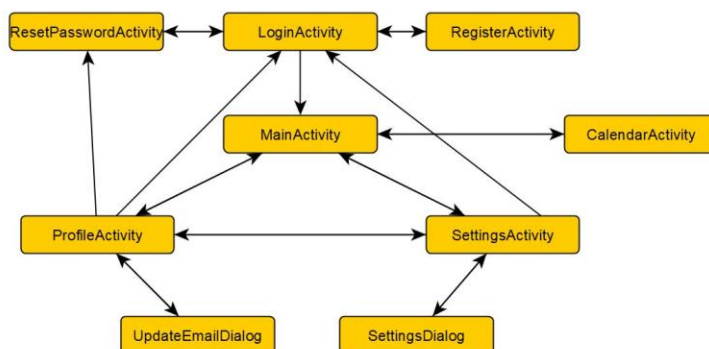
שם הקובץ	תיאור
LoginActivity.java	<b>שייכת לתת חבילה activities.</b> מחלקה האחראית למסך ההתחברות וכל מה שכלול בו. המחלקה מחברת בין הקוד לקובץ xml המגדיר את נראות המסך, מגדירה את הלוגיקה של המסך והאובייקטים שבו, ומתפעלת את המחלקות המתקשרות עם בסיס הנתונים.
RegisterActiviry.java	<b>שייכת לתת חבילה activities.</b> מחלקה האחראית למסך יצירת משתמש חדש וכל מה שכלול בו. המחלקה מחברת בין הקוד לקובץ xml המגדיר את נראות המסך, מגדירה את הלוגיקה של המסך והאובייקטים שבו, ומתפעלת את המחלקות המתקשרות עם בסיס הנתונים.
ResetPasswordActivity.java	<b>שייכת לתת חבילה activities.</b> מחלקה האחראית למסך איפוס סיסמא וכל מה שכלול בו. המחלקה מחברת בין הקוד לקובץ xml המגדיר את נראות המסך, מגדירה את הלוגיקה של המסך והאובייקטים שבו, ומתפעלת את המחלקות המתקשרות עם בסיס הנתונים. מחלקה זו מכילה שני מצבים, מצב אימות משתמש לשם אימות סיסמא ומצב אימות סיסמא. בהתאם למסך ממנו מגיעים למחלקה זו ובהתאם לפעולות המשתמש, המחלקה מציגה למשתמש את המצב המתאים.
ProfileActivity.java	<b>שייכת לתת חבילה activities.</b> מחלקה האחראית למסך פרופיל משתמש וכל מה שכלול בו. המחלקה מחברת בין הקוד לקובץ xml המגדיר את נראות המסך, מגדירה את הלוגיקה של המסך והאובייקטים שבו, ומתפעלת את המחלקות המתקשרות עם בסיס הנתונים.

מחלקה זו אחראית על מסך עדכון אימייל של משתמש. מסך זה אינו מסוג activity אלא מיוצג כדיאלוג. מחלקה זו מחברת בין דיאלוג עדכון אימייל לקובץ הxml המגדיר אותו, ומגדירה את הלוגיקה של הדיאלוג והאובייקטים שבו.	UpdateEmailDialog.java
מחלקה זו מכילה שיטות סטטיות המבצעות ולידציה לקלט המגיע מהמשתמש בעת יצירת הרגל חדש או עריכת הרגל. השיטות במחלקה מבצעות ולידציה לשדות השונים מחבילה זו ומחזירות String המתריע במידה וקיימת בעיה בקלט (או NULL אם הקלט תקין).	RegValidation.java

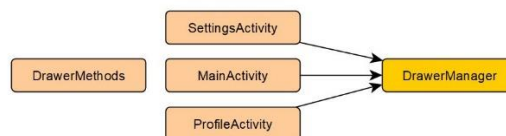
## נסתכל על הקשרים העיקריים בין המחלקות:

(נדלג על הקשרים שמכילים מחלקות מהחבילה utils או את הקובץ IntentManager מכיוון שקבצים אלו נועדו לשמש את כלל המחלקות ביישום)

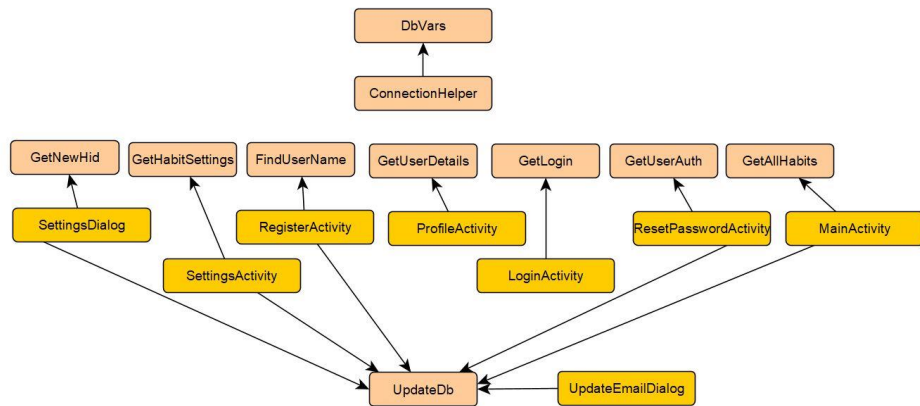
ראשית, נציג תרשים זרימה של תצוגת המסכים של היישום. המשתמש פותח את האפליקציה במסך ההתחברות, ומשם מתקדם בין מחלקות האפליקציה בהתאם לתרשים הזרימה הבא:



הדיאגרמה להלן מציגה את הקשרים בין המחלקות המנהלות את העמודים הראשיים של האפליקציה לבין המחלקות המנהלות את מגירת התפריט:



הדיאגרמה הבאה מציגה את הקשרים בין מחלקות יצירת החיבורים הנמצאות בחבילת ה dataconnections לבין המחלקות המשתמשות בחיבורים אלו:



נציג את הקשרים בין המחלקות בשאר החבילות ביישום:

