

Natali braids

עבודת גמר תכנון ותכנות מערכות

מגישה: נטלי מיכאלהשוילי

תעודת זהות: 326284791

שם המורה: אירנה

שם ההתמחות: תכנון ותכנות מערכות בטלפונים
ניידים תחת מערכת הפעלה Android

בית ספר: תיכון קריית שרת



תוכן עניינים

מבוא.....	3
מטרת המערכת.....	3
יכולות המערכת.....	3
תרשים זרימה.....	4
המסכים באפלקיציה.....	6-10
פירוט מסכים.....	11-18
הרשאות.....	19
גרסאת אנדרואיד מינימלית.....	19
מכשירים עליהם נבדקה המערכת.....	19
אתגרים מרכזיים.....	19
תרשים uml.....	20
Activities.....	21-29
Resources.....	30-31
Firebase.....	32
אלגוריתם מרכזי.....	33-35
סרוויס.....	36
ברודקאסט.....	37

38.....	Drawable
39.....	Custom List View
40.....	Dialog box
41.....	Shared preferences
42.....	רעיון לפיתוח עתידי
42.....	רפלקציה

מדריך משתמש

מבוא

בחרתי דווקא בפרויקט הזה כי אני כבר כמה שנים עושה צמות באופן מקצועי ואנשים קובעים אצלי תור דרך הטלפון ומבקשים ממני לשלוח להם תמונות של העבודות שלי. כתוצאה מכך החלטתי לפתח אתר שגם הלקוחות וגם אני נוכל להרוויח ממנו ולהקל על תהליך קביעת התור.

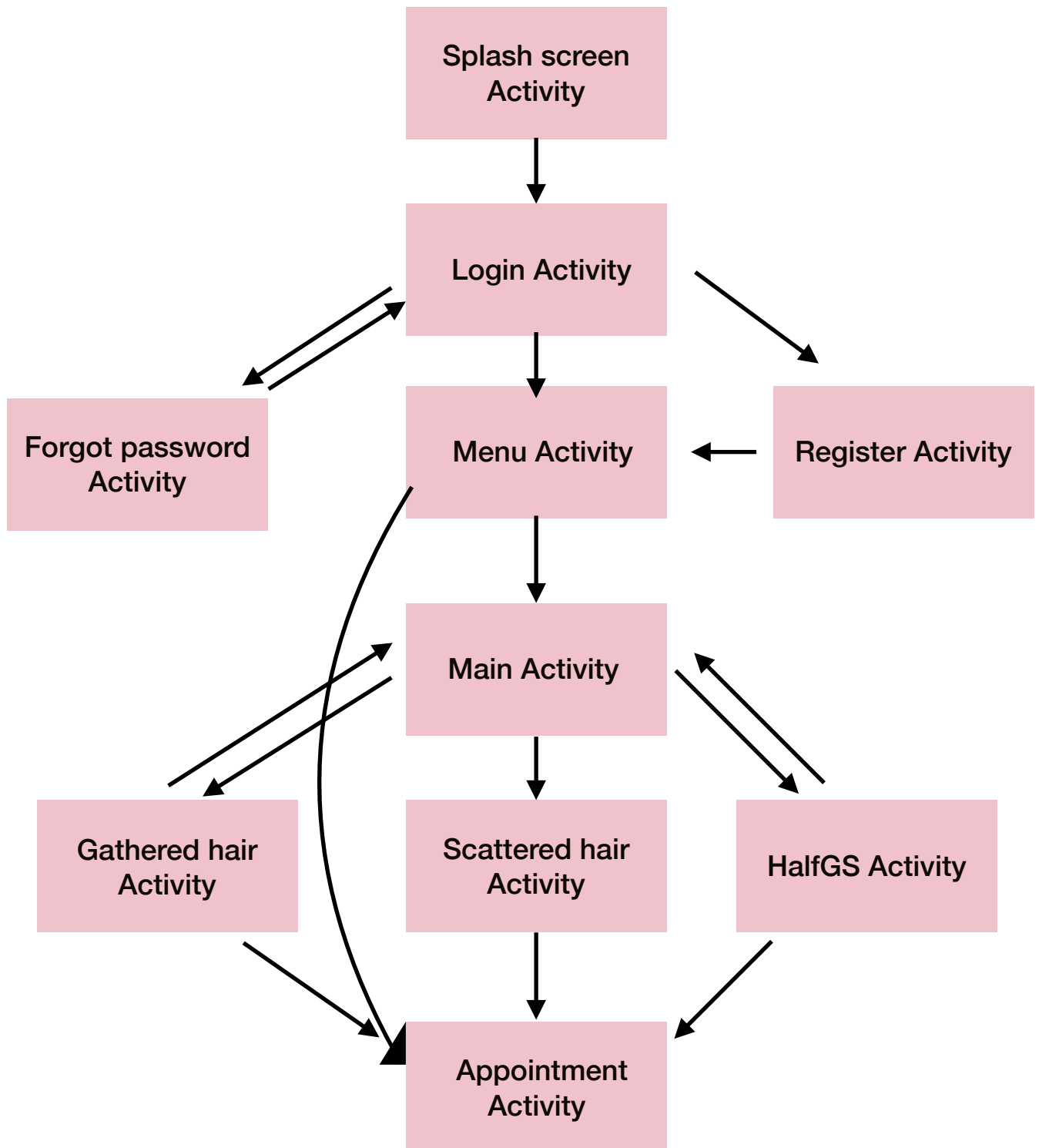
מטרת המערכת

המערכת נותנת ללקוח אפשרות לקבוע תור לצמות בקלות ביום ובשעה הנוחים לו, לראות את העבודות שלי בצורה הנוחה ביותר, לסמן לב על התסרוקות האהובות או על הקטגוריה האהובה ולפי התמונה, משך העבודה והמחיר להחליט מהי התסרוקת המושלמת בשבילו.

יכולות המערכת

1. אפשרות להתחבר כלקוח חוזר.
2. אפשרות להרשם כלקוח חדש.
3. אפשרות לשנות סיסמה אם המשתמש שכח אותה.
4. אפשרות להסתכל על התמונות של כל התסרוקות, שמן, מחירן וכמה זמן הן לוקחות.
5. אפשרות לסמן בלב סוגי תסרוקות אהובים(קטגוריות אהובות).
6. אפשרות לסמן בלב תסרוקת ספציפית אהובה.
7. אפשרות לקבוע תור ביום ובשעה המתאימים למשתמש.

תרשים זרימה



המסכים באפליקציה

מסך פתיחה:

NataliBraids

Natali Braids

EMAIL

PASSWORD

LOGIN

Create new user

Forgot password?

מסך הרשמה:

NataliBraids

Natali Braids

FULL NAME

EMAIL

PASSWORD

PHONE NUMBER

REGISTER

Already Registered? Click To Log In

מסך שכחתי סיסמה:

NataliBraids

Natali Braids

Enter Email


EMAIL

SEND PASSWORD TO EMAIL

מסך תפריט:

NataliBraids

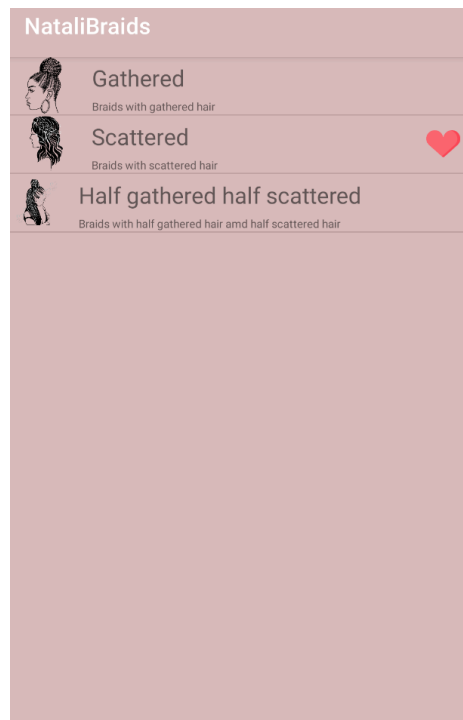
Natali Braids



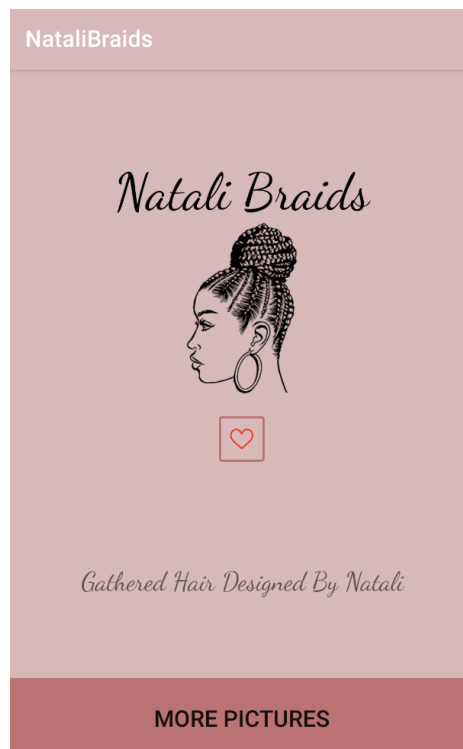
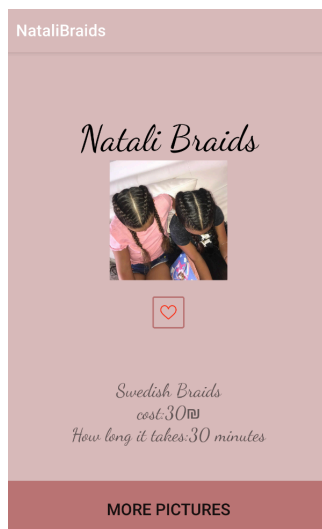
CHOOSE YOUR HAIRSTYLE

SET AN APPOINTMENT

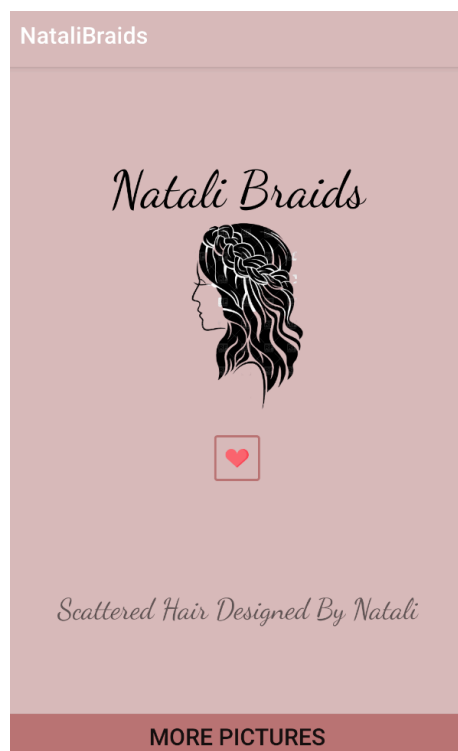
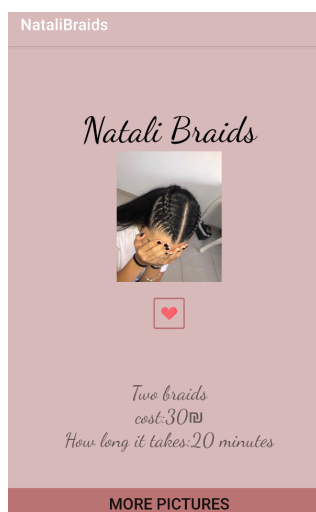
מסך בחירת סוג תסרוקת:



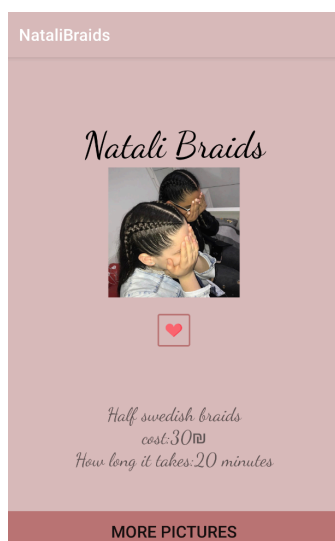
מסך צמות עם שיער אסוף:



מסך צמות עם שיער פזור:




מסך צמות עם חצי שיער אסוף וחצי שיער פזור:



מסך צמות קביעת תור:

NataliBraids

Natali Braids




Day Hour

Sunday 15:00

BOOK NOW

מסך דיאלוג:

Natali Braids



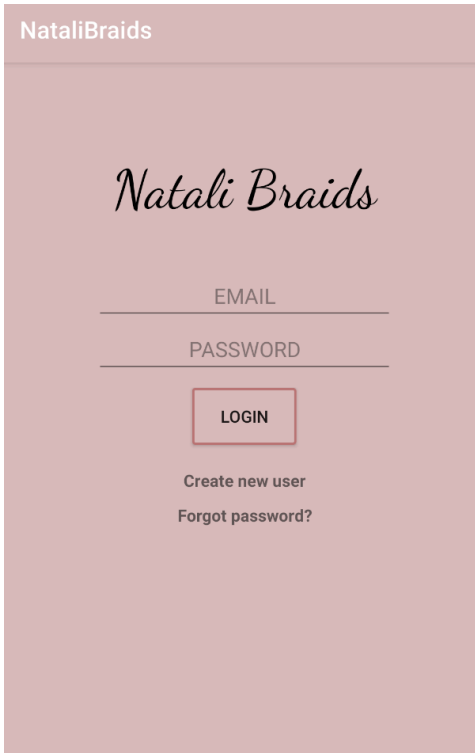
Like this Hair Style?
Would you like to book an appointment?

CANCEL YES

*Headband braid
cost: 30₪
How long it takes: 15 minutes*

פירוט מסכים

מסך פתיחה:



במסך הפתיחה שלושה
כפתורים-

login - כפתור התחברות
המיועד ללקוחות שנרשמו
בעבר בלבד.

כפתור - Create new user
הרשמה ללקוח חדש.
הכפתור מפנה למסך
הרשמה דרכו נרשמים ע"י
הזנת פרטים אישיים.

כפתור - Forgot password
המיועד ללקוחות שנרשמו
בעבר ושכחו את הסיסמה.
הכפתור מפנה למסך של
שינוי הסיסמה ע"י הזנת
כתובת האימייל.

מסך הרשמה:

NataliBraids

Natali Braids

FULL NAME

EMAIL

PASSWORD

PHONE NUMBER

REGISTER

Already Registered? Click To Log In

מסך ליצירת משתמש חדש
ע"י הזנת פרטיו האישיים- שם
מלא, אימייל, סיסמה וטלפון.

במסך זה שני כפתורים-

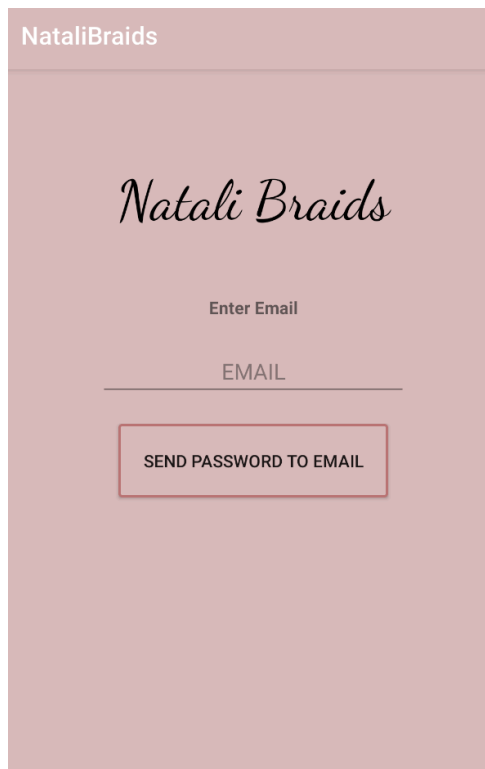
Register- כפתור
הרשמה לאחר הזנת
הפרטים האישיים.

Already registered?

-Click here to log in

כפתור המיועד
ללקוחות שנרשמו
בעבר(ואולי בטעות
נכנסו למסך הזה).

מסך שכחתי סיסמה:



מסך המיועד ללקוחות שנרשמו
בעבר אך שכחו את הסיסמה
שלהם. ע"י הזנת האימייל איתו
הם רשומים יהיה ניתן לשנות
את הסיסמה.

במסך זה כפתור אחד-

-Send password to email

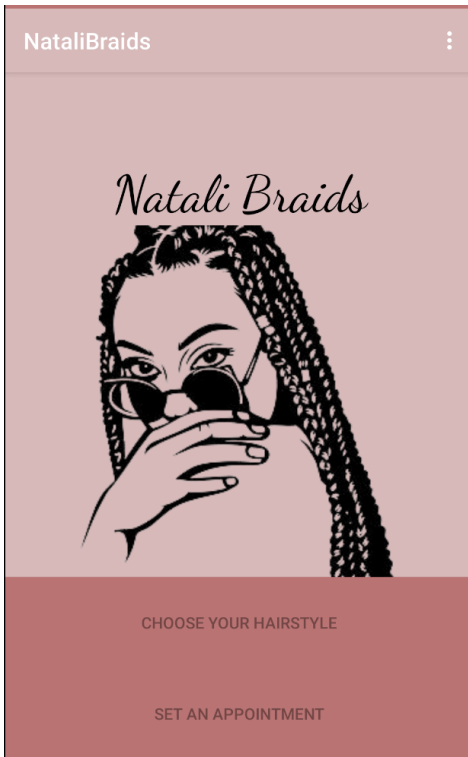
כפתור שלחיה עליו
תשלח לאימייל של
המשתמש אפשרות
לשנות סיסמה.

מסך תפריט:

במסך התפריט שני כפתורים-

-Choose your hairstyle
כפתור בחירת התסרוקת
היוביל למסך בו יהיה
ניתן לסמן קטגוריות
ותסרוקות אהובות בלב.

-Set an appointment
כפתור המוביל למסך של
קביעת תור.

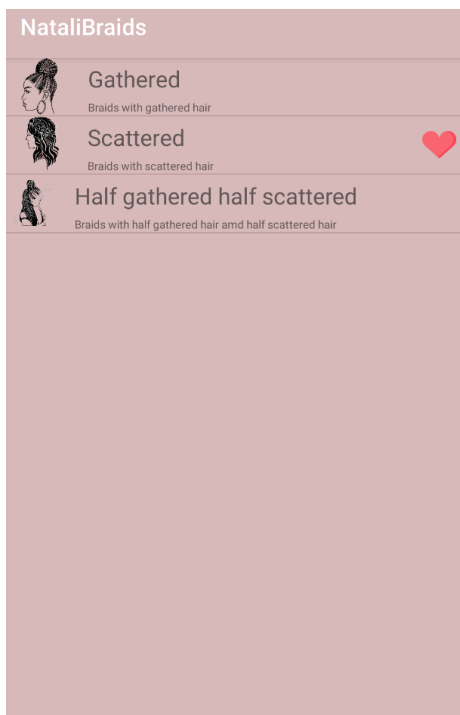


מסך בחירת סוג תסרוקת:

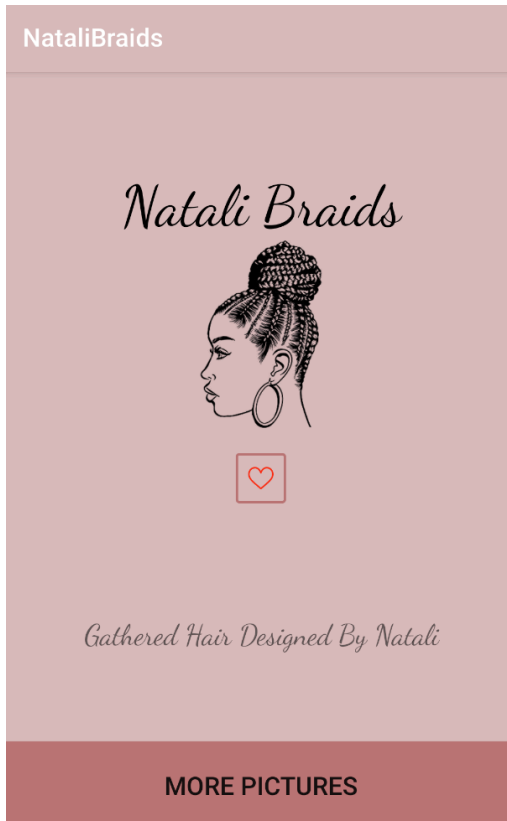
מסך זה מחלק את
התסרוקות לשלושה
סוגים:

- צמות עם שיער אסוף.
- צמות עם שיער פזור.
- צמות עם חצי שיער
אסוף וחצי שיער פזור.

על סוגי התסרוקות ניתן
ללחוץ כדי לראות תמונות
ופרטים נוספים על כל
אחד.



מסך צמות עם שיער אסוף:

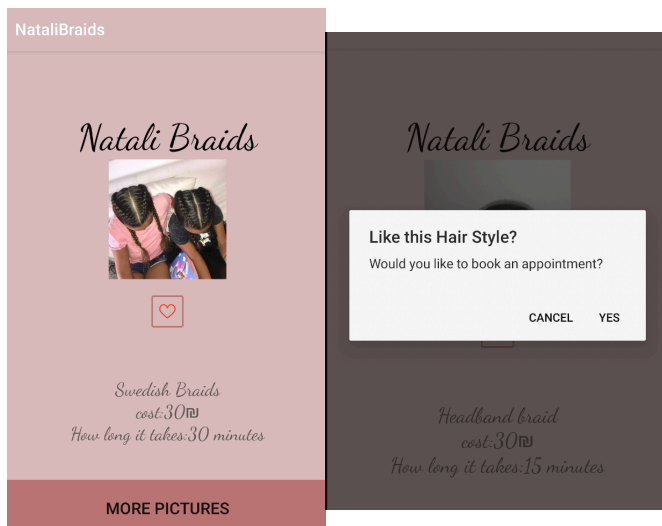


במסך זה ניתן לראות את כל התסרוקות עם השיער האסוף הקיימות. לחיצה על הלב במסך זה תסמן את הקטגוריה "צמות עם שיער אסוף" בלב (במסך הקודם).

במסך זה כפתור אחד-

More pictures - לחיצה על

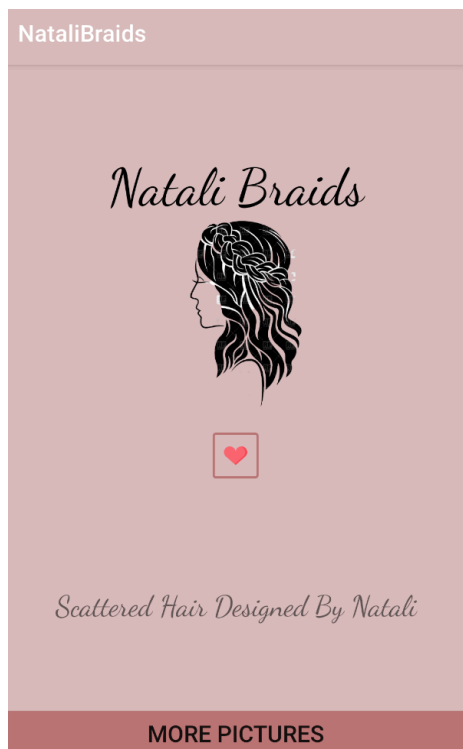
כפתור זה תציג בפני הלקוח תמונה של תסרוקת רלוונטית עם פרטים עליה.



לדוגמה בתסרוקת זאת ניתן לראות את התמונה שלה, את השם שלה, כמה היא עולה וכמה זמן היא לוקחת.

לחיצה על הלב תפתח חלון למשתמש שישאל אותו האם הוא מעוניין לקבוע תור. אם ילחץ "כן" הוא יעבור למסך קביעת התור.

מסך צמות עם שיער פזור:

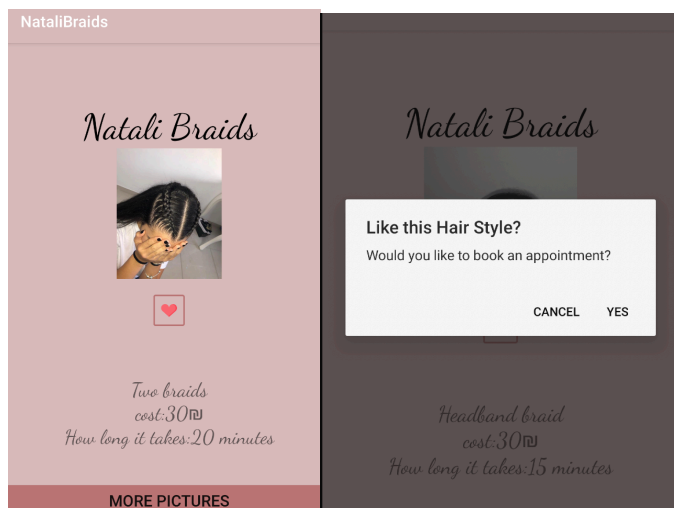


במסך זה ניתן לראות את כל התסרוקות עם השיער הפזור הקיימות. לחיצה על הלב במסך זה תסמן את הקטגוריה "צמות עם שיער פזור" בלב (במסך הקודם).

במסך זה כפתור אחד-

More pictures - לחיצה על

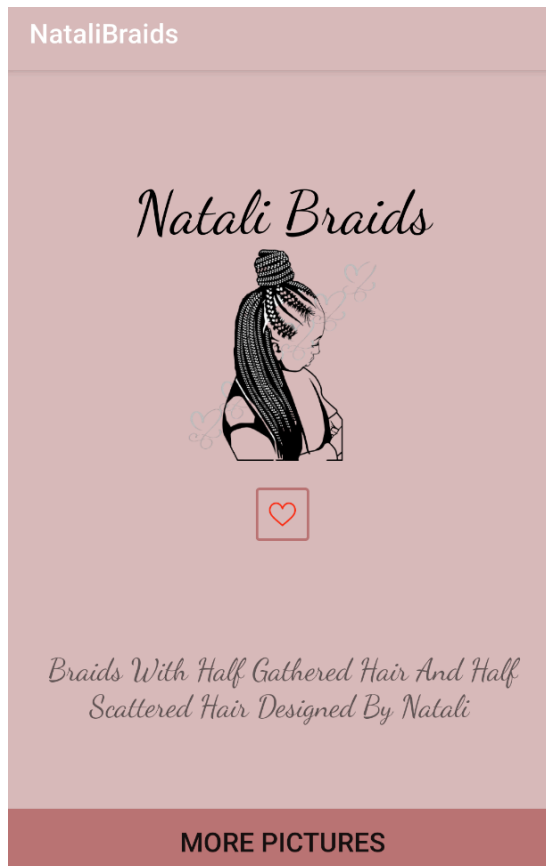
כפתור זה תציג בפני הלקוח תמונה של תסרוקת רלוונטית עם פרטים עליה.



לדוגמה בתסרוקת זאת ניתן לראות את התמונה שלה, את השם שלה, כמה היא עולה וכמה זמן היא לוקחת.

לחיצה על הלב תפתח חלון למשתמש שישאל אותו האם הוא מעוניין לקבוע תור. אם ילחץ "כן" הוא יעבור למסך קביעת התור.

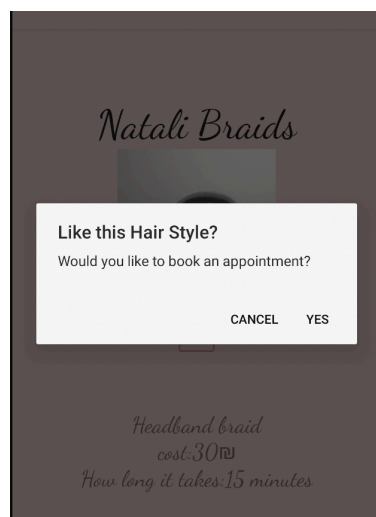
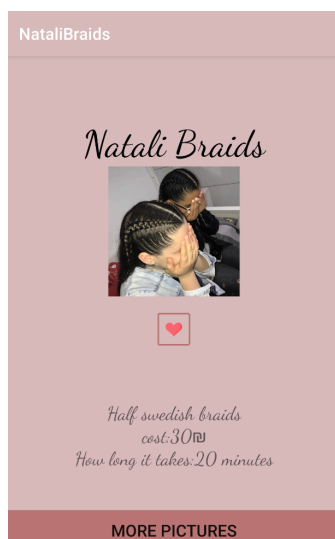
מסך צמות עם חצי שיער אסוף וזה שיער פזור:



במסך זה ניתן לראות את כל התסרוקות עם החצי שיער אסוף וחצי שיער פזור הקיימות. לחיצה על הלב במסך זה תסמן את הקטגוריה "צמות עם חצי שיער אסוף וחצי שיער פזור" בלב (במסך הקודם).

במסך זה כפתור אחד-

More pictures - לחיצה על כפתור זה תציג בפני הלקוח תמונה של תסרוקת רלוונטית עם פרטים עליה.




לדוגמה בתסרוקת זאת ניתן לראות את התמונה שלה, את השם שלה, כמה היא עולה וכמה זמן היא לוקחת.

לחיצה על הלב תפתח חלון למשתמש שישאל אותו האם הוא מעוניין לקבוע תור. אם ילחץ "כן" הוא יעבור למסך קביעת התור.

מסך צמות קביעת תור:

NataliBraids

Natali Braids



Day Hour

Sunday 15:00

BOOK NOW

מסך בו ניתן לבחור את היום
בשבוע והשעה המתאימים
למשתמש לצורך קביעת התור.

במסך זה כפתור אחד-

Book now - לחיצה על
כפתור זה תקבע
למשתמש תור לצמות
ביום ובשעה שבחר.

הרשאות

`android.permission.VIBRATE`

הרשאה לברודקאסט של הרטט שמודיע למשתמש לא לשכוח לקבוע תור.

גרסת אנדרואיד מינימלית

Android 4.1.3 API 30

מכשירים עליהם נבדקה המערכת

Pixel 2 API 30

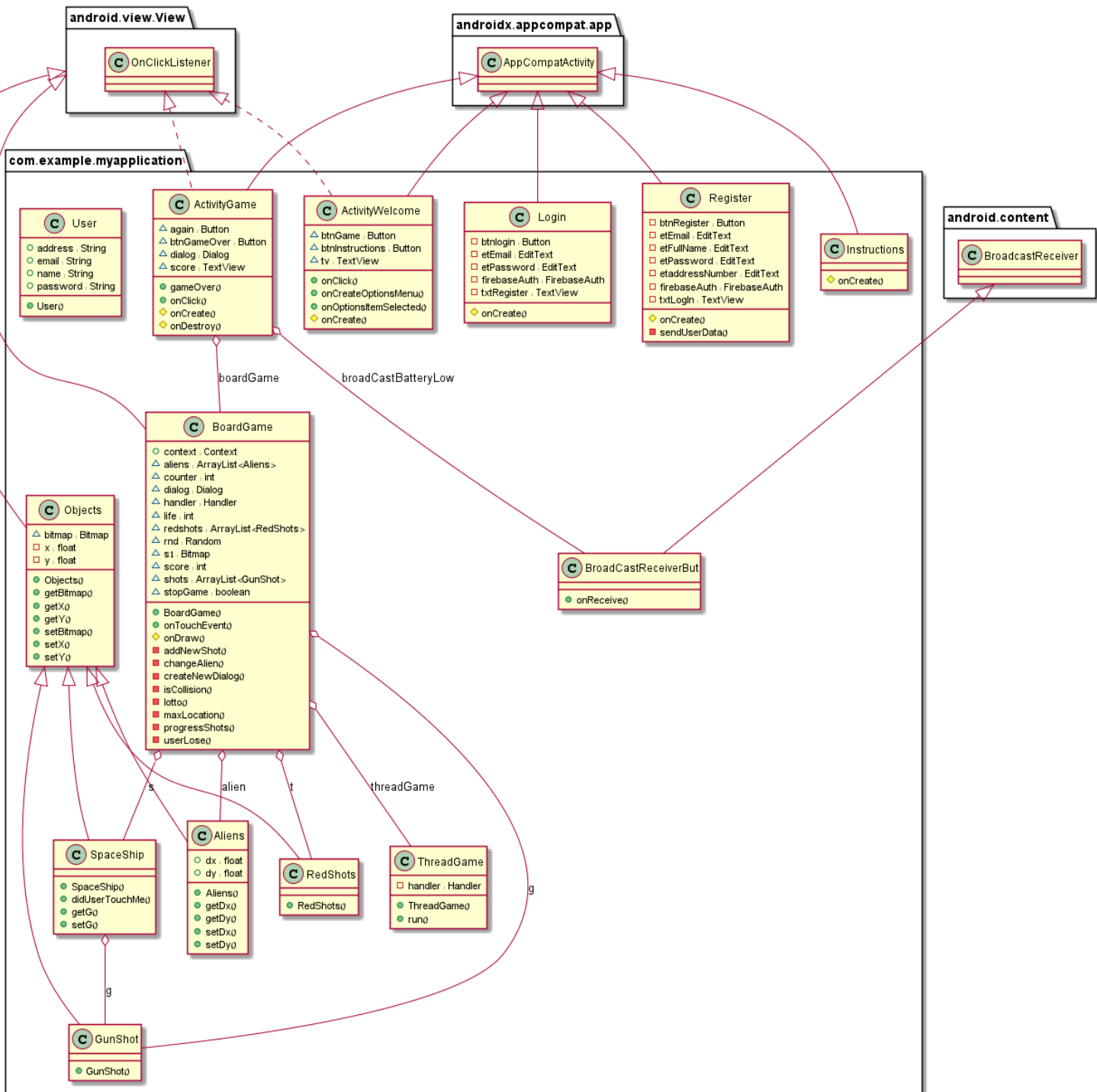
מדריך מתכנת

אתגרים מרכזיים

1. בפרוייקט בחרתי להשתמש במסד הנתונים Firebase וכדי להשתמש בו הייתי צריכה ללמוד עליו באופן עצמאי וללמוד כיצד להשתמש בממשק התכנות שלו על אנדרואיד.

תרשים מחלקות

MYAPPLICATION's Class Diagram

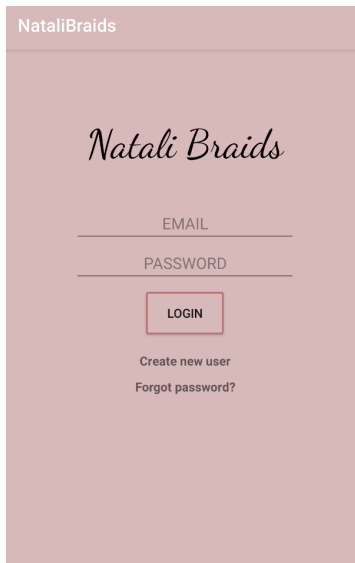


PlantUML diagram generated by SketchUML (<https://bitbucket.org/pmesmeurs/sketchuml>)
For more information about this tool, please contact philippe.mesmeur@gmail.com

Activities

Login Activity

תפקידו- התחברות ללקוחות שנרשמו בעבר.



תיאור הפעולה	כותרת הפעולה מתוך JAVA
פעולת הonClick שקוראת עקב לחיצה על הטקסט ויו tvRegister תגרום למעבר לאקטיבי RegisterActivity	tvRegister.setOnClickListener(new View.OnClickListener(){ Override@ public void onClick(View v)
פעולת הonClick שקוראת עקב לחיצה על הטקסט ויו tvForgotPassword תגרום למעבר לאקטיבי ForgotPasswordActivity	tvForgotPassword.setOnClickListener(new View.OnClickListener(){ Override@ public void onClick(View v)
פעולת הonClick שקוראת עקב לחיצה על הכפתור btnlogin בודקת דרך הפיירסבייס ע"י הקלדת אימייל וסיסמה אם המשתמש רשום.	btnlogin.setOnClickListener(new View.OnClickListener(){ public void onClick(View view)
אם המשתמש אכן רשום התוכנית תעבור לאקטיביטי הבא, MenuActivity אם התחברותו נכשלה, יוצג ה-Toast "Log In Failed" או "Illegal Arguments"	public void onComplete(@NonNull Task<AuthResult> task)

Natali Braids

FULL NAME

EMAIL

PASSWORD

PHONE NUMBER

REGISTER

Already Registered? Click To Log In

Register Activity

תיאור הפעולה	כותרת הפעולה מתוך JAVA
פעולה זו שקוראת עקב לחיצה על הטקסט ויו tvLogin תביא לסיום האקטיביטי הזה ותחזיר את התוכנית לאקטיביטי הקודם (Login Activity)	tvLogin .setOnClickListener(new View.OnClickListener(){ Override@ public void onClick(View view)
פעולה זו שקוראת עקב לחיצה על הכפתור btnRegister תיצור משתמש חדש בפירבייס.	btnRegister .setOnClickListener(new View.OnClickListener(){ Override@ public void onClick(View view)
אם ההרשמה תצליח תקרה הפעולה sendUserData ויהיה מעבר לאקטיביטי הבא. אם ההרשמה תכשל יוצג ה-Toast "Registration Failed"	public void onComplete(@NonNull Task<AuthResult> task)
פעולה השולחת ל-Database את הפרטים של המשתמש שהתחבר דרך customer . היא שולחת את השם המלא שלו, האימייל, הסיסמה ואת מספר הטלפון שלו. בנוסף, היא שומרת את הנתונים של המשתמש שהתחבר ב- class Customer	sendUserData (fullName , email , password , phone)

Natali Braids

Enter Email

EMAIL

SEND PASSWORD TO EMAIL

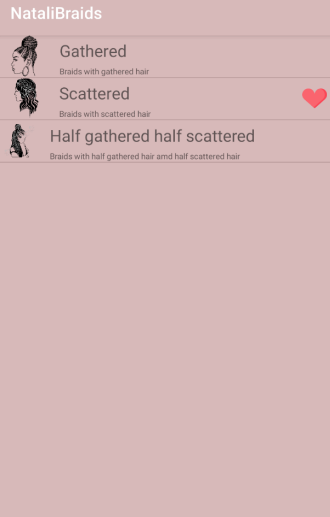
Forgot password Activity

תיאור הפעולה	כותרת הפעולה מתוך JAVA
פעולה זו שקוראת עקב לחיצה על הכפתור btnReset תביא לשליחת דואל למשתמש לאיפוס סיסמה.	btnReset .setOnClickListener(new View.OnClickListener(){ Override@ public void onClick(View view)
פעולת ולידציה לאימייל הבודקת אם הוא ריק או לא ומחזירה טיפוס מסוג בוליאן . אם האימייל תקין תחזיר true ואם לא תחזיר false ותציג את ה-Toast "Email cant be empty"	public boolean validate()
פעולה הבודקת האם שליחת הסיסמה לאימייל בוצעה בהצלחה. אם היא בוצעה יוצג ה-Toast "Password reset email was sent" אם היא נכשלה יוצג ה-Toast "Password reset failed"	public void onComplete(@NonNull Task<Void> task)



Menu Activity

תיאור הפעולה	כותרת הפעולה מתוך JAVA
פעולת שיתוף האפליקציה	public boolean onOptionsItemSelected(@NonNull MenuItem item)
פעולה היוצרת את התפריט	public boolean onCreateOptionsMenu(Menu menu)
פעולה המזהה את הלחיצות על הכפתורים. - לחיצה על הכפתור btHairStyles תעביר ל- MainActivity - לחיצה על הכפתור btAppointment תעביר ל- AppointmentActivity	public void onClick(View view)



Main Activity

כותרת הפעולה מתוך JAVA	תיאור הפעולה
<code>public void setlist()</code>	פעולה שמגדירה את העצמים ומכניסה נתונים אליהם, מוסיפה את הנתונים לadapter, array list ומקשרת לlist view
<code>public void onItemClick(AdapterView<?> parent, View view, int position, long id)</code>	
<code>protected void onResume()</code>	פעולה שמפעילה מחדש את הפעולה set list בכל פעם שאני חוזרת למסך main activity בצורה כזו או אחרת.
<code>()startAlert</code>	פעולה שלאחר 10 שניות מפעילה את הבדורקאסט שמזכיר למשתמש לקבוע תור.

Gathered hair Activity

תיאור הפעולה	כותרת הפעולה מתוך JAVA
בפעולה זו מכניסים לתוך מערך שיצרנו מסוג Details את השם של כל תסרוקת עם שיער אסוף, את המחיר, id, וכמה התסרוקת עולה.	<code>private void initObjects()</code>
הפעולה מקשרת בין הכפתורים שיצרנו בxml לכפתורים ב-java ול- OnClickListener ומגדירה את ה-SharedPreferences אם הערך gatheredLike יהיה true התמונה תתחלף לעם לב ועם הערך יהיה false התמונה תתחלף לבלי לב	<code>private void initViews()</code>
פעולה המזהה לחיצה על כפתורים. -לחיצה על הכפתור btMoreP תעביר את הlayout לתמונה הבאה עם הפרטים עליה(מהמערך). -לחיצה על הכפתור btnLike תחליף את התמונה לעם לב, תשמור true ב-SharedPreferences ותציג למשתמש את הdialog box	<code>public void onClick(View view)</code>
הפעולה תזהה לחיצה ארוכה על כפתור הלב, תחליף את התמונה מעם לב לבלי לב ותשמור false ב-SharedPreferences	<code>public boolean onLongClick(View view)</code>
פעולה שמעבירה את התוכנית למסך קביעת התור אם המשתמש לחץ על yes	<code>alert.setPositiveButton("yes", new DialogInterface.OnClickListener(){ Override@ public void onClick(DialogInterface dialogInterface, int which)</code>
פעולה שסוגרת את הdialog אם המשתמש לחץ על cancel	<code>alert.setNegativeButton("cancel", new DialogInterface.OnClickListener(){ Override@ public void onClick(DialogInterface dialogInterface, int i)</code>

Scattered hair Activity

תיאור הפעולה	כותרת הפעולה מתוך JAVA
בפעולה זו מכניסים לתוך מערך שיצרנו מסוג Details את השם של כל תסרוקת עם שיער אסוף, את המחיר, id, וכמה התסרוקת עולה.	<code>private void initObjects()</code>
הפעולה מקשרת בין הכפתורים שיצרנו בxml לכפתורים ב-java ול- OnClickListener ומגדירה את הSharedPreferences אם הערך scatteredLike יהיה true התמונה תתחלף לעם לב ועם הערך יהיה false התמונה תתחלף לבלי לב	<code>private void initViews()</code>
פעולה המזהה לחיצה על כפתורים. -לחיצה על הכפתור btMoreP תעביר את הlayout לתמונה הבאה עם הפרטים עליה(מהמערך). -לחיצה על הכפתור btnLike תחליף את התמונה לעם לב, תשמור true ב SharedPreferences ותציג למשתמש את ה dialog box	<code>public void onClick(View view)</code>
הפעולה תזהה לחיצה ארוכה על כפתור הלב, תחליף את התמונה מעם לב לבלי לב ותשמור false ב- SharedPreferences	<code>public boolean onLongClick(View view)</code>
פעולה שמעבירה את התוכנית למסך קביעת התור אם המשתמש לחץ על yes	<code>alert.setPositiveButton("yes", new DialogInterface.OnClickListener(){ Override@ public void onClick(DialogInterface dialogInterface, int which)</code>
פעולה שסוגרת את ה dialog אם המשתמש לחץ על cancel	<code>alert.setNegativeButton("cancel", new DialogInterface.OnClickListener() { Override@ public void onClick(DialogInterface dialogInterface, int i)</code>

HalfGS Activity

תיאור הפעולה	כותרת הפעולה מתוך JAVA
בפעולה זו מכניסים לתוך מערך שיצרנו מסוג Details את השם של כל תסרוקת עם שיער אסוף, את המחיר, id, וכמה התסרוקת עולה.	<code>private void initObjects()</code>
הפעולה מקשרת בין הכפתורים שיצרנו בxml לכפתורים ב-java ול- OnClickListener ומגדירה את הSharedPreferences אם הערך halfLike יהיה true התמונה תתחלף לעם לב ועם הערך יהיה false התמונה תתחלף לבלי לב	<code>private void initViews()</code>
פעולה המזהה לחיצה על כפתורים. -לחיצה על הכפתור btMoreP תעביר את הlayout לתמונה הבאה עם הפרטים עליה(מהמערך). -לחיצה על הכפתור btnLike תחליף את התמונה לעם לב, תשמור true ב SharedPreferences ותציג למשתמש את ה dialog box	<code>public void onClick(View view)</code>
הפעולה תזהה לחיצה ארוכה על כפתור הלב, תחליף את התמונה מעם לב לבלי לב ותשמור false ב- SharedPreferences	<code>public boolean onLongClick(View view)</code>
פעולה שמעבירה את התוכנית למסך קביעת התור אם המשתמש לחץ על yes	<code>alert.setPositiveButton("yes", new DialogInterface.OnClickListener(){ Override@ public void onClick(DialogInterface dialogInterface, int which)</code>
פעולה שסוגרת את ה dialog אם המשתמש לחץ על cancel	<code>alert.setNegativeButton("cancel", new DialogInterface.OnClickListener() { Override@ public void onClick(DialogInterface dialogInterface, int i)</code>

Appointment Activity

תיאור הפעולה	כותרת הפעולה מתוך JAVA
פעולה זו שקוראת עקב לחיצה על הכפתור btnBook מתחברת ל auth ול realtime database שב firebase	btnBook.setOnClickListener(new View.OnClickListener(){ Override@ public void onClick(View view)
פעולה המכניסה לתוך משתנים את שם המשתמש והאימייל שלי, היא מוציאה את הפרטים האלה מן הפירובייס.	public void onDataChange(DataSnapshot dataSnapshot)
אם מאיזו סיבה המידע על המשתמש לא נמצא יופיע Toast שירשום "Error"	public void onCancelled(@NonNull DatabaseError error)
פעולה ששולחת לווטסאפ שלי את היום והשעה שהמשתמש קבע תור ביחד עם שמו והאימייל שלו.	private void openWhatsApp(String name,String email)
פעולה שיוצרת את הספינר של השעות בשבוע	private void loadSpinner()
פעולה שיוצרת את הספינר של הימים בשבוע	private void loadSpinner1()
פעולה ששומרת במשתנים day ו- hour את השעה שנבחרה בספינר הראשון והיום שנבחר בספינר השני.	public void onItemSelected(AdapterView<?> parent, View view, int position, long id)

Resources

שם הLayout	Activity אליו שייך	הסבר
activity_splash_screen.xml	SplashScreenActivity	אקטיביטי למען האנימציה של המספרים
activity_login.xml	LoginActivity	אקטיביטי לצורך התחברות
activity_forgot_password.xml	ForgotPasswordActivity	אקטיביטי למשתמשים ששכחו סיסמה
activity_register.xml	RegisterActivity	אקטיביטי להרשמה
activity_main.xml	MainActivity	אקטיביטי לבחירת סוג תסרוקת רצויה
activity_gathered_hair.xml	GatheredHairActivity	אקטיביטי עם תמונות של צמות עם שיער אסוף, פרטים על כל תמונה ואפשרות לסמן בלב וללכת לאקטיביטי קביעת התור.
activity_scattered_hair.xml	ScatteredHairActivity	אקטיביטי עם תמונות של צמות עם שיער פזור, פרטים על כל תמונה ואפשרות לסמן בלב וללכת לאקטיביטי קביעת התור.
activity_half_g_s.xml	HalfGSActivity	אקטיביטי עם תמונות של צמות עם שיער חצי אסוף וחצי פזור, פרטים על כל תמונה ואפשרות לסמן בלב וללכת לאקטיביטי קביעת התור.
activity_appointment.xml	AppointmentActivity	אקטיביטי קביעת התור

Menu

שם הLayout	Activity אליו שייך	הסבר
activity_menu.xml	MenuActivity	תפריט שדרכו ניתן לגשת או לmain activity שבו בוחרים את סוג התסרוקת שאתה אוהב או לגשת לappointment activity שבו ניתן לקבוע תור לצמות

מבנה נתונים- פיירבייס

למה בחרתי דווקא בפיירבייס?

רוב האפליקציות צריכות לדעת את זהות המשתמש. ידיעת זהות המשתמש מאפשרת לאפליקציה לשמור באופן מאובטח את נתוני המשתמש בענן ולספק את אותה חוויה מותאמת אישית בכל מכשירי המשתמש.

אימות Firebase מספק שירותי SDK קלים לשימוש וספריות ממשק משתמש מוכנות לאימות משתמשים באפליקציה שלך. הוא תומך באימות באמצעות סיסמאות, מספרי טלפון, ספקי זהות מאוחדים פופולריים כמו גוגל, פייסבוק וטוויטר ועוד.

Business Model

אלגוריתם מרכזי

האלגוריתם המרכזי שלי הוא האלגוריתם של קביעת התור. המטרה והתפקיד העיקרי של האפליקציה שלי הוא לגרום למשתמש לקבוע תור.

תכונות

```
private Spinner spinner;
private Spinner spinner1;
private Button btnBook;
private String day = "";
private String hour = "";
private String name = "";
private String email = "";

ArrayAdapter<String> dataAdapter;
private List<String> listOfItems;
private FirebaseAuth firebaseAuth;
```

פעולות

```
private void openWhatsApp(String name,String email) {
    PackageManager packageManager = this.getPackageManager();
    Intent i = new Intent(Intent.ACTION_VIEW);
    String phone = "+972549796297";
    String message = "\nשלום, אני אשמח לקבוע פגישה אצלכם!" +
        "\n:בטאריך:" + day + "\n" +
        "\n:בשעה:" + hour + "\n" +
        "\n:שמי:" + name + "\n" +
        "\n:האימייל שלי הוא:" + email + "\n" +
        "\n:תודה רבה!";
    try {
        String url = "https://api.whatsapp.com/send?phone=" + phone + "&text=" + URLEncoder.encode(message, "UTF-8");
        i.setPackage("com.whatsapp");
        i.setData(Uri.parse(url));
        if (i.resolveActivity(packageManager) != null) {
            this.startActivity(i);
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

המתודה openWhatsApp שולחת למספר הטלפון שלי את היום והשעה שהמשתמש בחר לקבוע בו תור (באמצעות הספינר) ובנוסף שולחת את שם המשתמש והאימייל שלו- היא לוקחת את הפרטים האלה מהrealtime database .

```

private void loadSpinner() { //the hours in the week
    listOfItems = (Arrays.asList(getResources().getStringArray(R.array.hours_of_visits)));

    dataAdapter = new ArrayAdapter<String>( context: this, android.R.layout.simple_spinner_item, listOfItems);

    dataAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);

    spinner.setAdapter(dataAdapter);
    spinner.setOnItemSelectedListener(this);
}

private void loadSpinner1() { //the days in the week
    listOfItems = (Arrays.asList(getResources().getStringArray(R.array.time_of_visits)));

    dataAdapter = new ArrayAdapter<String>( context: this, android.R.layout.simple_spinner_item, listOfItems);

    dataAdapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);

    spinner1.setAdapter(dataAdapter);
    spinner1.setOnItemSelectedListener(this);
}

```

פעולות אלה יוצרות את הספינר של השעות ואת הספינר של הימים בשבוע.

```

@Override
public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
    String selected= parent.getItemAtPosition(position).toString();
    //String checkView= view.toString();
    if (view == spinner)
        day = selected;
    if (view==spinner1)
        hour = selected;
}

```

פעולה זו מכניסה את היום שנבחר למשתנה day ואת השעה שנבחרה למשתנה hour.

```

btnBook.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        FirebaseDatabase firebaseDatabase=FirebaseDatabase.getInstance();
        FirebaseAuth= FirebaseAuth.getInstance();
        DatabaseReference ref = firebaseDatabase.getReference( path: "Users/"+ Objects.requireNonNull(firebaseAuth.getUid())+"/UserData");
        ref.addValueEventListener(new ValueEventListener(){
            @Override
            public void onDataChange(DataSnapshot dataSnapshot) {
                for(DataSnapshot datas: dataSnapshot.getChildren()){
                    if(datas.getKey().equals("name"))
                        name = datas.getValue().toString();
                    if(datas.getKey().equals("email"))
                        email = datas.getValue().toString();
                }
            }
            @Override
            public void onCancelled(@NonNull DatabaseError error) {
                Toast.makeText(context: AppointmentActivity.this, text: "Error", Toast.LENGTH_SHORT).show();
            }
        });
        openWhatsApp(name,email);
        finish();
    }
});

```

לחיצה על הכפתור btnBook תביא להוצאת שם המשתמש והאימייל שלו מה-real time database. לאחר מכן תקרה הפעולה openWhatsApp.

Service

הסרויס שבחרתי משמיע מוזיקה באפלקציה שלי.

```
public class BackgroundSoundService extends Service {
    MediaPlayer player;
    public IBinder onBind(Intent arg0) { return null; }
    @Override
    public void onCreate() {
        super.onCreate();
        player = MediaPlayer.create( context: this, R.raw.music);
        player.setLooping(true); // Set looping
        player.setVolume( leftVolume: 100, rightVolume: 100);
    }
    public int onStartCommand(Intent intent, int flags, int startId) {
        player.start();
        return START_STICKY;
    }

    public IBinder onUnbind(Intent arg0) {
        // TO DO Auto-generated method
        return null;
    }

    @Override
    public void onDestroy() {
        player.stop();
        player.release();
    }

    @Override
    public void onLowMemory() {

    }
}
```

*שמרתי את המוזיקה בraw

Broadcast

מטרת הברודקאסט היא לעשות רעש של רטט 10 שניות לאחר שהוא לוחץ על תא ב custom list view ולהזכיר לו לא לשכוח לקבוע תור.

על מנת ליצור Broadcast Receiver צריך ליצור class ולרשת מ - BroadcastReceiver וליישם את הפונקציה onReceive.

```
public class MyBroadcastReceiver extends BroadcastReceiver {  
    @Override  
    public void onReceive(Context context, Intent intent) {  
        Toast.makeText(context, text: "Don't forget to book an appointment!",  
            Toast.LENGTH_LONG).show();  
        Vibrator vibrator = (Vibrator) context.getSystemService(Context.VIBRATOR_SERVICE);  
        assert vibrator != null;  
        vibrator.vibrate( milliseconds: 1000);  
    }  
}
```

```
public void startAlert() {  
    int sec = 10;  
    Intent intent = new Intent( packageContext: this, MyBroadcastReceiver.class);  
    PendingIntent pendingIntent = PendingIntent.getBroadcast(  
        this.getApplicationContext(), requestCode: 123, intent, PendingIntent.FLAG_UPDATE_CURRENT);  
    AlarmManager alarmManager = (AlarmManager) getSystemService(ALARM_SERVICE);  
    assert alarmManager != null;  
    alarmManager.set(AlarmManager.RTC_WAKEUP, triggerAtMillis: System.currentTimeMillis()  
        + (sec * 1000), pendingIntent);  
}
```

Drawable

בגלל שהאפליקציה שלי מכילה את האפשרות לראות תמונות של תסרוקות שינן תמונות רבות באפליקציה.

logo1- תמונת הלוגו של הlayout של צמות עם שיער אסוף.

תמונות של צמות עם שיער אסוף:

swedenbraids

sixbraids

fourbraids

logo3- תמונת הלוגו של הlayout של צמות עם שיער פזור.

תמונות של צמות עם שיער פזור:

noabraids

headbandbraid

topbraids

sidebraid

sidepartheadband

smallsidebraid

braidswithbuns

halfpartheadbandbraid

logo10- תמונת הלוגו של הlayout של צמות עם חצי שיער אסוף וחצי שיער פזור.

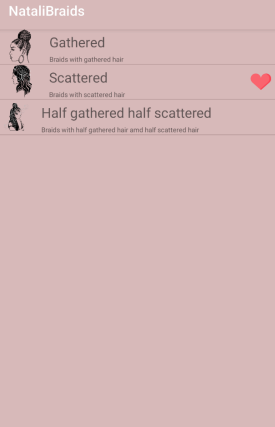
תמונות של צמות עם שיער חצי אסוף וחצי פזור:

halfswedenbraids

twoconnectedbraids

round_button- תמונה של המלבן מסביב לכפתור של הלב

menuphotoo- התמונה שמופיעה בתפריט הראשי



Custom list view

איך יצירתי?

1. עיצבתי layout בשם item.xml שיהווה תבנית לכל שורה ב-custom list view. בעצם יצירת תא בודד



2. בניתי class בשם BraidsKinds שמתאר את מה שכל שורה מכילה.

```
public BraidsKinds(boolean like, String title, String subTitle,
    Bitmap bitmap)
{
    this.like=like;
    this.title = title;
    this.subTitle = subTitle;
    this.bitmap = bitmap;
}
```

3. בניתי class בשם BraidsKindsAdapter שיורש ArrayAdapter. הקלאס ירוץ כגודל הרשימה וישים את הפרטים בו על הליסט ויו שלנו.

4. יצירתי אקטיביטי בשם Main Activity שבו אני מגדירה את העצמים ומכניסה נתונים אליהם, מוסיפה את הנתונים לarray list, יוצרת adapter ומקשרת לlist view.

Dialog box

בלחיצה על כפתור הלב (שמופיע בתמונות של התסרוקות) יופיע למשתמש הדיאלוג שישאל אותו אם הוא מעוניין לקבוע תור, אם ילחץ כן יועבר לAppointment Activity.

איך יצירתי?

```
AlertDialog.Builder alert = new AlertDialog.Builder(context: this);
alert.setTitle("Like this Hair Style?");
alert.setMessage("Would you like to book an appointment?");
alert.setCancelable(false);

alert.setPositiveButton(text: "yes", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialogInterface, int which) {
        dialogInterface.dismiss();
        Intent intent = new Intent(packageContext: GatheredHairActivity.this, AppointmentActivity.class);
        startActivity(intent);
    }
});

alert.setNegativeButton(text: "cancel", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialogInterface, int i) {
        dialogInterface.dismiss();
    }
});

alert.create();
alert.show();
```

1. פתיחת דיאלוג חדש
2. הגדרת הכותרת וההודעה
3. הכפתור ה"חיובי" יהיה "yes" והכפתור ה"שלילי" יהיה "cancel"
4. המשתמש לא יוכל לבטל את הדיאלוג מבחוץ
5. לחיצה על הכפתור "yes" תביא לסגירת הדיאלוג ולמעבר לAppointment Activity
6. לחיצה על הכפתור "cancel" תביא לסגירת הדיאלוג

Shared Preferences

אני שומרת בשרד פרפרנס את כל הלבבות שסומנו על תמונה או קטגוריה כלשהי.

איך עשיתי?

```
btnLike.setOnLongClickListener(new View.OnLongClickListener() {  
    @Override  
    public boolean onLongClick(View view) {  
        SharedPreferences prefs = GatheredHairActivity.this.getSharedPreferences( name: "likes", mode: 0);  
        SharedPreferences.Editor editor = prefs.edit();  
        editor.putBoolean("Gathered"+photoNumG, false);  
        editor.apply();  
        btnLike.setImageResource(R.drawable.ic_heart);  
        return false;  
    }  
});
```

כשכפתור הלב באקטיביטי "צמות עם שיער אסוף" למשל מזהה לחיצה ארוכה עליו, כלומר הורדה מלב הוא ילך לשרד פרפרנס ויכניס לתוך gathered של התמונה הספציפית עליה לחץ false . בנוסף, תיווצר החלפה של התמונה לתמונה עם לב ריק.

```
if (view == btnLike){  
    //save like to database  
    SharedPreferences prefs = GatheredHairActivity.this.getSharedPreferences( name: "likes", mode: 0);  
    SharedPreferences.Editor editor = prefs.edit();  
    editor.putBoolean("Gathered"+photoNumG, true);  
    editor.apply();  
    btnLike.setImageResource(R.drawable.ic_heart_full);  
}
```

כשכפתור הלב באותו האקטיביטי מזהה לחיצה רגילה עליו, כלומר השמה של לב ע"י המשתמש הוא ילך לשרד פרפרנס ויכניס לתוך gathered של התמונה הספציפית עליה לחץ true . בנוסף, תיווצר החלפה של התמונה לתמונה עם לב מלא.

רעיון לפיתוח עתידי

האפליקציה שלי מתמקדת רק ב"סניף" אחד, הסניף שלי. אם היה לי עוד זמן הייתי יוצרת כמה סניפים ונותנת ללקוח את האפשרות להקליד את הכתובת שלו והיה מופיע לו הסניף שקרוב ביותר לביתו.

בנוסף, אם היה ברשותי עוד זמן הייתי מטפלת באפשרות של כפילויות בתורים ולא הייתי יוצאת מנקודת הנחה שהיום והשעה שרשומים בספינר שלי בהכרח פנויים.

רפלקציה

למרות כל הקשיים שהיו לי בפיתוח הפרוייקט, נהנתי מאוד לעשות אותו, העשרתי את הידע שלי בתכנות בצורה יוצאת דופן, למדתי לעבוד בצורה עצמאית והכי חשוב- לא משנה כמה קשה לא להתייאש אף פעם! אני מאוד מרוצה מהתוצר הסופי מבחינת העיצוב ומבחינת הקוד: