Project in Android Development

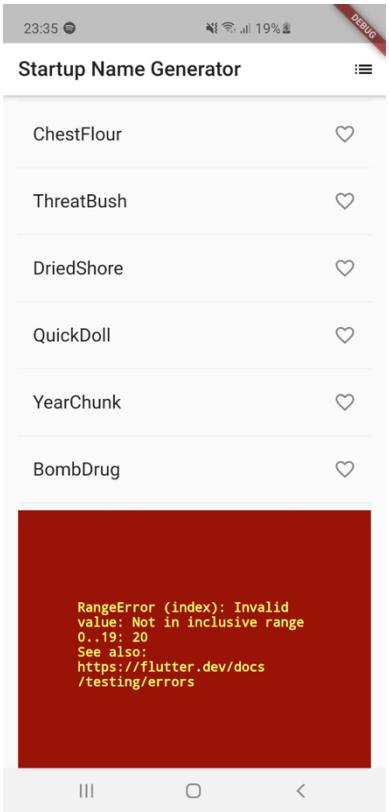
HW1

Ofir Yaniv, 314626623

חלק 1

1. שתי השורות שגורמות לכך שהרשימה היא אינסופית הן:

```
if (index >= _suggestions.length) {
  // ...then generate 10 more and add them to the
  // suggestions list.
  _suggestions.addAll(generateWordPairs().take(10));
}
```



הדרך שבה הן עושות את זה היא על ידי בדיקה האם הגענו לאינדקס שגדול מהגודל של הרשימה של ההצעות שיש לנו עד כה ואם כן אנו מוסיפים עוד איברים לרשימה.
במידה ונמחק את השורות הללו וננסה לרדת למטה ברשימה נקבל שגיאה כמתוארת פה:

2. הדרך ליצור את הרשימה עם המפרידים כפי שמתוארת בדוקיומנטציה היא שימוש בListView.separated במקום ב ListView.build שמאפשר הוספה של מפרידים על ידי קריאה לseperatorBuilder בכל פעם שנוצרת שורה חדשה וכך יוצר מפריד לאחר כל שורה. עושים זאת באופן הבא:

```
Widget _buildSuggestions() {
  return ListView.separated(
      padding: const EdgeInsets.all(16),
      itemCount: _suggestions.length,
      itemBuilder: (BuildContext _context, int i) {

        if (i >= _suggestions.length) {
          _suggestions.addAll(generateWordPairs().take(10));
        }
        return _buildRow(_suggestions[i]);
      },
      separatorBuilder: (_, __) => Divider(),
  ); // ListView.separated
}
```

אני חושב שמימוש זה הוא מוצלח בהרבה מהמימוש הקודם שכן הוא קריא יותר וברור יותר. אין התעסקות עם לוגיקה והתעסקות עם אינדקסים כפולים (כפי שתואר בהערות הקוד הקודם- כך שהאינדקסים לא היו באמת הערך של האובייקט).
המימוש הזה דורש הרבה פחות דוקומנטציה כדי להבין אותו באופן פשוט. הקוד הגיוני ומסביר את עצמו.

3. החלק המטפל onTap() משנה את הרשימה של המילים השמורות _saved כלומר הוא משנה את המצב (state) של המחלקה RandomWords (כלומר את המחלקה RandomWordsState). כפי שראינו, על מנת שהשינוי של הwidget יוצג על המסך מעודכן לאחר שינוי המצב נידרש לre-rendering ולכן קוראים לsetState(). בלי קריאה זו אז השינוי לא יגרום לrender מחדש והשינוי לא יראה במסך.

חלק 2

1. המתודה בה השתמשתי להחלפת העמוד היא המתודה push. המתודה הזו דוחפת למעשה route חדש אל המחסנית (ושולפים אותו כאשר רוצים לחזור לroute הקודם).
כלומר אנו יוצרים למעשה PageRoute חדש ודוחפים אותו לראש המחסנית במעבר למסך login.
מתודה נוספת שיכולה לממש את החלפת המסכים היא באמצעות המתודה pushNamed. על מנת להשתמש במתודה זו מגדירים widget לכל אחד מהמסכים,

מגדירים את שמות המסכים השונים ואז קוראים בעזרת pushNamed למסך שמעוניינים שיוצג.

2. המתודה שהשתמשתי בה על מנת להציג את הsnackBar היא showSnackBar. מתודה זו שייכת לscaffold (widget) ועל כן הוא הווידג'ט הנוסף הנדרש על מנת להציג את הsnackBar. כמובן שההצגה מתרחשת על ידי יצירת ווידג'ט מסוג SnackBar ושליחה שלו לתוך המתודה שתוארה.