

סיור מפת פריז

איור. 1-6



בפרק זה, תבנה אפליקציית מדרכי טיולים לטיול בפריז. יצירת אפליקציית מפות שפועלת במלואה עשויה להיראות ממש מסובכת, אבל App Inventor מספק שני רכיבים ברמה גבוהה כדי לעזור: ActivityStarter המאפשר לך להפעיל אפליקציה נוספת מהאפליקציה שלך, כולל מפות Google, ו-WebView שמציג כל דף אינטרנט שאתה רוצה בתוך לוח משנה של האפליקציה שלך. תוכלו לחקור את שני המרכיבים הללו ולבנות שתי גרסאות שונות של מדריך טיולים.

מה תלמד

פרק זה מציג את הרכיבים והמושגים הבאים של ממציא האפליקציות:

- הרכיב Activity Starter להפעלת אפליקציות אנדרואיד אחרות מהמכשיר שלך

אפליקציה:

- רכיב ה- WebView להצגת דפי אינטרנט בתוך האפליקציה שלך.

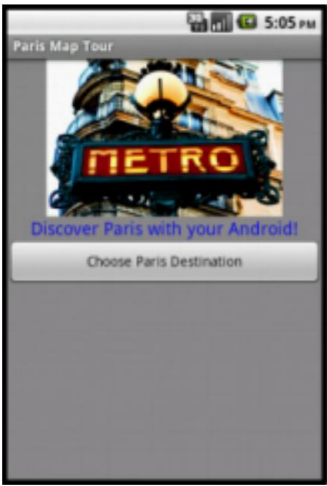
- כיצד להשתמש במשתני רשימה כדי לאחסן מידע עבור האפליקציה שלך.

•רכיב ListPicker לתת למשתמש את היכולת לבחור מתוך רשימה של מיקומים.

•כיצד לבנות כתובת URL באופן דינמי כדי להציג מפות שונות.

עיצוב הרכיבים

צור פרויקט חדש App Inventor-בוקרא לו "ParisMapTour".ממשק המשתמש עבור לאפליקציה יש רכיב Imageעם תמונה של פריז, רכיב Labelעם קצת טקסט, רכיב ListPickerשמגיע עם כפתור משווק, ובזה גרסה ראשונה, רכיב ActivityStarter(לא גלוי). אתה יכול לעצב את רכיבים באמצעות תמונת המצב באיור. 6-1



איור. 6-2.אפליקציית Paris Map Tourפועלת באמולטור

תזדקק לרכיבים המפורטים בטבלה 6-1כדי לבנות אפליקציה זו. גרור כל אחד רכיב מהלוח לתוך Viewer-הושם אותו כפי שצוין.

טבלה. 6-1.רכיבים לסיוור המפה בפריז

קטגוריית רכיבים	איך תקרא לזה	מטרה		
תמונה	תמונה	הצגת תמונה של פריז על המסך.		
טקסט	טקסט	הצגת טקסט של "תגלה את פריז עם האנדרואיד שלך!"		
רכיב	רכיב	לחיצה על רכיב של אפליקציה יעיד תופיע		
רכיב	רכיב	ActivityStarter1הפעל את אפליקציית המפות כאשר נבחר יעד.		

הגדרת המאפיינים של ActivityStarter

ActivityStarter הוא רכיב שבאמצעותו אתה יכול להפעיל כל אפליקציית אנדרואיד, כולל מפות Google או אפליקציה אחרת משלך. תחילה תבנה את ParisMapTour כך שאפליקציית המפות תופעל כדי להציג מפות מסוימות בהתבסס על בחירת המשתמש. לאחר מכן המשתמש יכול להקיש על כפתור החזרה כדי לחזור לאפליקציה שלך ולבחור יעד אחר.

ActivityStarter הוא רכיב ברמה נמוכה יחסית בכך שתצטרך להגדיר חלק נכסים עם מידע שיהיה מוכר למתכנת, Java Android SDK, אך זר לחלוטין לשאר 99.99% מהעולם. עבור אפליקציה זו, הזן את המאפיינים כפי שצוינו בטבלה 2-6 והיזהר - הם תלויי רישיות, כלומר חשוב אם אות היא אות גדולה או קטנה.

טבלה 2-6. מאפייני ActivityStarter להפעלת מפות Google

תכונה	ערך
פעולה	<code>android.intent.action.VIEW</code>
ActivityPackage	<code>com.google.android.apps.maps</code>
ActivityClass	<code>com.google.android.maps.MapActivity</code>

בעורך הבלוקים, תגדיר מאפיין אחד נוסף, `DataUri` המאפשר לך לספק כתובת URL להפעלת מפה ספציפית במפות Google. יש להגדיר מאפיין זה בעורך הבלוקים במקום Component Designer - במכיוון שהוא צריך להיות דינמי: הוא ישתנה בהתאם לאם המשתמש יבחר לבקר במגדל אייפל, בלובר או בקתדרלת נוטרדאם.

נגיע לעורך בלוקים בעוד רגע, אבל יש עוד כמה פרטים לטפל לפני שתוכל לעבור לתכנות ההתנהגות עבור הרכיבים שלך:

1. הורד את הקובץ `metro.jpg` כדי לטעון לפרויקט שלך. לאחר מכן, הגדר אותו כמאפיין של `Image1`.
2. רכיב `ListPicker` מגיע עם כפתור; כאשר המשתמש לוחץ עליו, האפשרויות מופיעות. הגדר את הטקסט של כפתור זה על ידי שינוי מאפיין הטקסט של `ListPicker1` לבחר יעד בפרויקט.

הוספת התנהגויות לרכיבים

בעורך הבלוקים, תצטרך להגדיר רשימה של יעדים ושתי התנהגויות:

•כאשר האפליקציה מתחילה, האפליקציה טוענת את היעדים לתוך ListPicker רכיב כך שהמשתמש יוכל לבחור אחד.

•כאשר המשתמש בוחר יעד מתוך ListPicker, האפליקציה המפות מופעל ומציג מפה של יעד זה. בגרסה הראשונה הזו של האפליקציה, פשוט תפתח את מפות ותנחה אותו להריץ חיפוש אחר היעד הנבחר.

יצירת רשימת יעדים

פתח את עורך הבלוקים וצור משתנה עם רשימת יעדי פריז לפי באמצעות הבלוקים המפורטים בטבלה 3-6.

טבלה 3-6.בלוקים ליצירת משתנה יעדים

משתנה/מקרה		
אתחול משתנים גלובליים ("יעדים") צור רשימה של היעדים.		
עשה את רשימת היעדים.		
מקלט הראשון.	טקסט ("סיוור אייפל")	
מקלט השני.	טקסט ("מוזיאון הלובר")	
מקלט השלישי.	טקסט ("קתדרלת נוטרדאם")	

כאשר אתה גורר את בלוק יצירת רשימה לתוך האפליקציה שלך, יהיו לו רק שניים זמינים שקעים. אתה יכול להוסיף עוד אחד על ידי לחיצה על הסמל הכחול כהה אותו והוספת שלישי פריט.

לאחר שעשית את זה, פשוט צור את בלוקי הטקסט עבור כל יעד ומקום אותם בשלושת השקעים של צור רשימה, כפי שמוצג באיור 2-6.



איור 2-6.רשימה של שלושה פריטים

מתן אפשרות למשתמש לבחור יעד

הרשימה שזה עתה הגדרת אינה מופיעה בממשק המשתמש -אין משתנים. אתה תעשה השתמש ברכיב ListPicker כדי להציג את רשימת הפריטים לבחירת המשתמש. אתה טוען מראש את האפשרויות לתוך ListPicker על ידי הגדרת המאפיינים Elements לרשימה. עבור אפליקציה זו, אתה רוצה להגדיר את המאפיין Elements עבור ListPicker -לרשימת היעדים שיצרת זה עתה. כי זה צריך להיות מוגדר רק פעם אחת, אתה תעשה זאת

הוספת התנהגויות לרכיבים 103

הגדר התנהגות זו באירוע . `Screen1.Initialize` אתה תצטרך את הבלוקים שהם
המפורטים בטבלה 6-4.

טבלה 4-6. חסימות להפעלת ListPicker כאשר האפליקציה מתחילה

מטרה	מגרה	סוג בלוק
אירוע זה מופעל כאשר האפליקציה מתחילה.	מסך 1. אתחול	
הגדר מאפיין זה לרשימה שברצונך שתופיע.	הגדר את ListPicker1-ListPicker1.Elements	
רשימת היעדים.	גרור החוצה מבלוק אתחול משתנה	להשיג יעדים גלובליים

איך הבלוקים עובדים

מסך 1. אתחול מופעל כאשר האפליקציה מתחילה. איור 3-6 ממחיש שהמטפל באירועים מגדיר את המאפיין של ListPicker של Elements ששלושת היעדים יופיעו.



איור 4-6. אתחול את ListPicker עם שלוש האפשרויות כאשר האפליקציה מופעלת



בדוק את האפליקציות שלך לחץ על התחבר והגדר בדיקה חיה עם המכשיר או האמולטור שלך. לאחר מכן, לחץ על הכפתור שכותרתו "בחר יעד בפריז". בחר הרשימה אמור להופיע עם שלושת הפריטים. בשלב זה, שום דבר לא אמור לקרות כשאתה בוחר פריט.

פתיחת מפות עם כתובת אתר לחיפוש

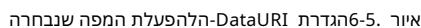
לאחר מכן, תתכנתו את האפליקציה כך שכאשר המשתמש בוחר באחד מהיעדים, `ActivityStarter` מפעיל את מפות גוגל ומחפש את המיקום הנבחר. ראשית, שקול את כתובת האתר `http://maps.google.com?q=Paris` כאשר אתה מקליד כתובת אתר זו בשורת הכתובת של דפדפן, היא מציגה מפה של פריז. ה "משתתף לכתובות URL רבות; זה מסמל שמגיע פרמטר. פרמטר הוא המידע שהאתר צריך כדי לעבד את הבקשה. במקרה זה, שם הפרמטר הוא "q" קיצור של "שאלתה", והערך שלו הוא "Paris". הוא מורה למפות גוגל איזו מפה להציג.

באפליקציה זו, תבנה כתובת URL באופן דינמי, ותוסיף את ערך הפרמטר על סמך המיקום שהמשתמש בוחר. כך תוכלו להציג מפות שונות בהתאם לבחירותיו של המשתמש.

טבלה 5-6. חסימות להפעלת מפות Google עם Activity Starter-ה

כאשר המשתמש בוחר מתוך `ListPicker` הפריט הנבחר מאוחסן ב `ListPicker.Selection` ואירוע `AfterPicking` מופעל. כפי שנראה ב

איור 4-6 המאפיין `DataUri` מוגדר לאובייקט טקסט המשלב "http://maps.google.com/?q=" עם הפריט הנבחר. לכן, אם המשתמש בחר בפריט הראשון, "סיום Eifel", ה-`DataUri` יוגדר "http://maps.google.com/?q= Tour Eifel".



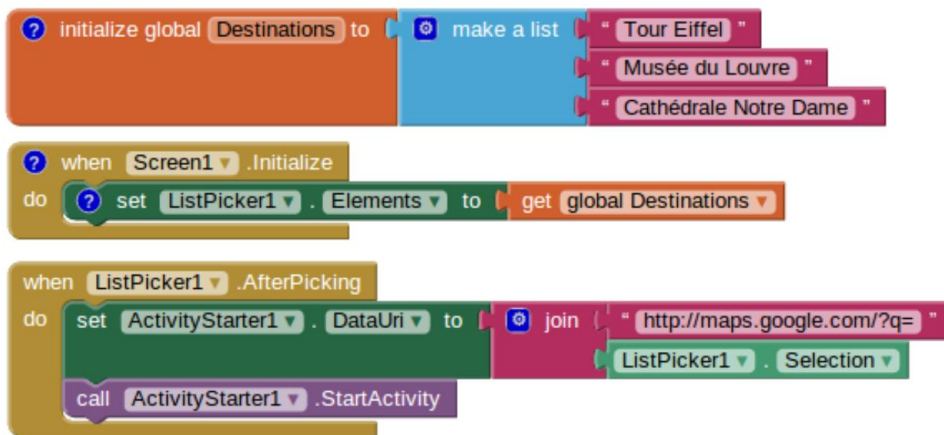
הוספת התנהגויות לרכיבים



בדוק את האפליקציה שלך הפעל מחדש את האפליקציה ולחץ שוב על הלחצן "בחר יעד בפריז". כאשר אתה בוחר באחד מהיעדים, האם מופיעה מפה של אותו יעד? האם אתה יכול לחזור לאפליקציה שלך עם כפתור החזרה של המכשיר?

האפליקציה השלמה: סיור במפה עם מתחיל פעילות

איור 5-6 מציג את תצורת הבלוק הסופי עבור גרסה 1 של סיור המפה של פריז.



איור 6-6. אפליקציית סיור המפה המלאה (גרסה 1)

סיור וירטואלי עם הצופה באינטרנט

`ActivityStarter` הוא רכיב חשוב מכיוון שהוא מספק גישה לכל אפליקציה אחרת במכשיר. אבל, יש דרך אחרת לבנות מדריך טיולים שמשתמש ברכיב אחר, במקום זאת; ה- `WebView` הוא חלונית שאתה מציב ישירות בתוך האפליקציה שלך שמתנהג כמו דפדפן. אתה יכול לפתוח כל דף אינטרנט, כולל מפת גוגל, במציג, ותוכל לשנות באופן תכנותי את הדף שיופיע. בניגוד ל-`ActivityStarter`, להשתמש שלך אף פעם לא עוזב את האפליקציה שלך, כך שאתה לא צריך לסמוך על כך שהוא ילחץ על כפתור החזרה כדי לחזור.

בגרסה השנייה הזו של האפליקציה, תשתמשו ב- `WebView` וגם תבלטו את האפליקציה כך שהיא תפתח כמה תצוגות מוגדלות ותצוגות רחוב של המונומנטים של פריז. אתה תגדיר רשימה שנייה ותשתמש בסכימה מסובכת יותר כדי להחליט איזו מפה להציג. כדי להתחיל, תחילה תחקור את מפות Google כדי לקבל את כתובות האתרים של כמה מפות ספציפיות. אתה עדיין תשתמש באותם ציוני דרך פריזאיים עבור היעדים, אך כאשר המשתמש בוחר אחד, תשתמש באינדקס (המיקום ברשימה) לבחירתו כדי לבחור ולפתוח מפה ספציפית עם זום או תצוגת רחוב.

לפני שתמשיך, אולי תרצה לשמור את הפרויקט שלך (באמצעות שמירה בשם) כדי שיהיה לך א
עודק של סיור המפה של ActivityStarter שיצרת עד כה. בדרך זו, אם אתה עושה משהו
שגורם לבעיות באפליקציה שלך, אתה תמיד יכול לחזור לגרסה העובדת הזו ולנסות שוב.

הוסף את ה-WebView

במעצב, מחק את הרכיב . ActivityStarter לאחר מכן, מהמגירה של ממשק המשתמש, גרור
פנימה רכיב WebView והצב אותו מתחת לרכיבים האחרים.
בטל את הסימון של המאפיין Screen1.Scrollable כדי שה-WebView יציג דפים בצורה נכונה.

מציאת כתובת האתר למפות ספציפיות

השלב הבא הוא לפתוח את מפות Google במחשב שלך כדי למצוא את המפות הספציפיות
שברצונך להפעיל עבור כל יעד:

1. במחשב שלך, דפדף אל <http://maps.google.com>.

2. חפש נקודת ציון (למשל, מגדל אייפל).

3. התקרב לרמה הרצויה.

4. בחר את סוג התצוגה הרצויה (למשל, Street View).

5. תפוס את כתובת האתר. בגרסה הקלאסית של מפות, אתה לוחץ על כפתור הקישור ליד
בפינה השמאלית העליונה של חלון המפות והעתק את כתובת האתר של המפה.
בגרסה החדשה יותר של מפות גוגל אתה יכול פשוט לתפוס את כתובת האתר מסרגל הכתובות.

השתמש בתוכנית זו כדי ליצור כמה מפות מגניבות של המונומנטים של פריז ולחלץ את
כתובות אתרים. טבלה 6-6 מספקת כמה דוגמאות אם אתה מעדיף להשתמש בהן (כתובות
האתר קוצרו עם השירות bit.ly).

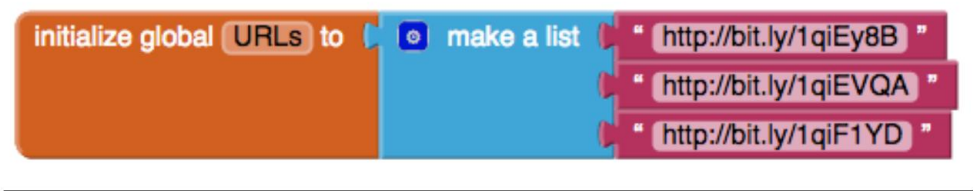
טבלה 6-6. כתובות אתרים של סיורים וירטואליים עבור מפות Google

כתובת האתר של מפות	ציון דרך
bit.ly/1qiEVQA	סיור באייפל
http://1qiEy8B	מוזיאון הלובר
http://bit.ly/1qiF1YD	מוזיאון הלובר (תצוגת רחוב)

כדי להציג כל אחת מהמפות הללו בדפדפן, הדבק את כתובות האתרים מטבלה 6-6
בשורת הכתובות.

הגדרת רשימת כתובות האתרים

תזדקק לרשימה בשם כתובות אתרים, המכילה כתובת אתר עבור כל אחד מהיעדים. ליצור רשימה זו כפי שמוצג באיור 6-6 ככך שהפריטים מתאימים לפריטים ב- רשימת היעדים (כלומר, כתובת האתר הראשונה צריכה להתאים ליעד הראשון, האיילפל מגדל).



איור 6-7. ההערכת והדבק את כתובות האתרים לתוך בלוקי הטקסט של רשימת כתובות האתרים

שינוי בורר הרשימה. התנהגות לאחר בחירה

בגרסה הראשונה של אפליקציה זו, התנהגות ListPicker.AfterPicking מגדירה את UriData לשילוב של "http://maps.google.com/?q=" והיעד של המשתמש בחר מתוך הרשימה (למשל, "טור איילפל"). בגרסה השנייה הזו, ה- AfterPicking ההתנהגות חייבת להיות מתוכננת יותר, מכיוון שהמשתמש בוחר מתוך רשימה אחת (יעדים), אבל האפליקציה בוחרת מרשימת כתובות האתרים עבור כתובת האתר. ספציפית, כאשר המשתמש בוחר פריט מתוך ListPicker, אתה צריך לדעת את האינדקס של הבחירה כדי שתוכל להשתמש בה כדי לבחור את כתובת האתר הנכונה מהרשימה. נסביר יותר על מה זה אינדקס בעוד רגע, אבל זה עוזר להגדיר את הבלוקים תחילה לטובים יותר להמחיש את הרעיון. יש לא מעט בלוקים הנדרשים לפונקציונליות הזו, כולם המפורטים בטבלה 6-7.

טבלה 6-7. חסימות לבחירת פריט רשימה על סמך בחירת המשתמש

מטרה	מקור	מקור
אירוע זה מופעל כאשר המשתמש בוחר פריט.	ListPicker1	ListPicker1.AfterPicking
האינדקס (מיקום) של הפריט הנבחר.	ListPicker1	ListPicker1.SelectionIndex
בחר פריט מרשימת כתובות האתרים.	בחר פריט רשימה	
רשימת כתובות האתרים.	גרור אותו מהמשתנה אתחול	לקבל כתובות URL גלובליות
טען את כתובת האתר במציג כדי להציג את המפה.	WebView	WebView.GoToURL

איך הבלוקים עובדים

כאשר המשתמש בוחר פריט מתוך `ListPicker` אירוע `AfterPicking` מופעל, כפי שמוצג באיור 6-7. הפריט הנבחר - למשל, "טור אייפל" - נמצא ב- `ListPicker.Selection` השתמשת בנכס זה בגרסה הראשונה של אפליקציה זו. עם זאת, `ListPicker` ליש גם מאפיין `SelectionIndex` המתאים למיקום היעד הנבחר ברשימה. לכן, אם נבחר "סיור אייפל", ה- `SelectionIndex` יהיה 1; אם "מוזיאון הלובר" ייבחר, זה יהיה 2; ו-3 "קתדרלת נוטרדאם דה פריז" תיבחר, היא תהיה 3.



איור 6-8. פתח את כתובת האתר שנבחרה ב-`WebView`

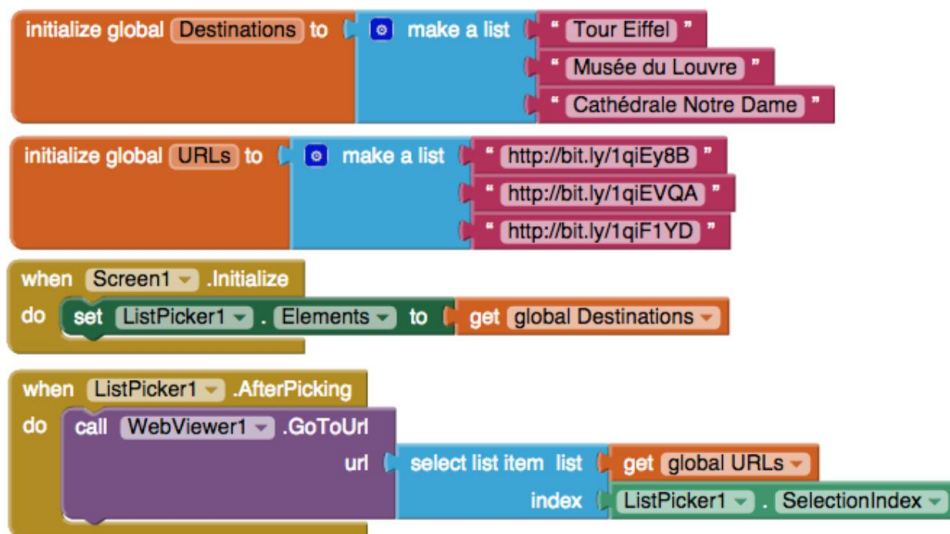
אתה משתמש ב- `ListPicker.SelectionIndex` כדי לבחור פריט מרשימת כתובות האתרים. זה עובד מכיוון שהפריטים בשתי הרשימות, יעדים וכתובות URL מסונכרנים: היעד הראשון מתאים לכתובת ה-LRU הראשונה, השני לשני והשלישי לשלישי. לכן, למרות שהמשתמש בוחר פריט מרשימה אחת, אתה יכול להשתמש בבחירה שלו (טוב, האינדקס לבחירתו) כדי לבחור את כתובת האתר הנכונה להצגה.



בדוק את האפליקציה שלך במכשיר, לחץ על הכפתור שכותרתו "בחר יעד בפריז". הרשימה אמורה להופיע עם שלושת הפריטים. בחר אחד מהפריטים וראה איזו מפה מופיעה.

האפליקציה השלמה: סיור במפה (מציג אינטרנט)

איור 6-8 מציג את תצורת הבלוק הסופי עבור גרסה שנייה זו של סיור המפה של פריז.



איור 9-6. אפליקציית סיור המפה המלאה (גרסת WebView)

וריאציות

הנה כמה וריאציות מוצעות לנסות:

- צור סיור וירטואלי במקום העבודה או בית הספר שלך, או לחופשה הבאה שלך יעד.
- חקור את ActivityStarter והשתמש בו כדי לשלוח אימייל או להפעיל אפליקציה כגון YouTube (ראה <http://bit.ly/1qiFx8Z> לעזרה).
- קשה: צור אפליקציית סיור וירטואלי הניתנת להתאמה אישית המאפשרת למשתמש ליצור מדריך למיקום לפי בחירתה על ידי הזנת השם של כל יעד יחד עם כתובת האתר של המפה המתאימה. תצטרך לאחסן את הנתונים במסד נתונים של TinyWebDB וליצור אפליקציית סיור וירטואלי שעובדת עם הנתונים שהזנו. לדוגמה כיצד ליצור מסד נתונים של TinyWebDB, ראה MakeQuiz/TakeQuiz.

אפליקציה.

סיכום

הנה כמה מהרעיונות שכיסינו בפרק זה:

- אתה יכול להשתמש במשתני רשימה כדי להחזיק נתונים כגון יעדי מפה וכתובות URL.
- רכיב ListPicker מאפשר למשתמש לבחור מתוך רשימה של פריטים. ה מאפיין האלמנטים של ListPicker מחזיק את הרשימה, המאפיין Selection מחזיק את

הפריט שנבחר, `SelectionIndex` מחזיק את המיקום של הפריט שנבחר, ואירוע `AfterPicking` מופעל כאשר המשתמש בוחר פריט מהרשימה.

• רכיב `ActivityStarter` מאפשר לאפליקציה שלך להפעיל אפליקציות אחרות. פרק זה הדגים את השימוש שלו עם אפליקציית `Google Maps`, אבל אתה יכול להפעיל גם דפדפן או כל אפליקציית אנדרואיד אחרת, אפילו עוד אחת שיצרת בעצמך.

• אתה יכול להשתמש ב- `ListPicker.SelectionIndex` כדי לקבל את המיקום של פריט שא המשתמש בוחר מתוך רשימה. לאחר מכן תוכל להשתמש באינדקס הזה כדי לבחור מידע מרשימה אחרת (שהפריטים שלה מסונכרנים עם הרשימה הראשונה). למידע נוסף על משתני רשימה ורכיב `ListPicker`, ראה פרק 19.

