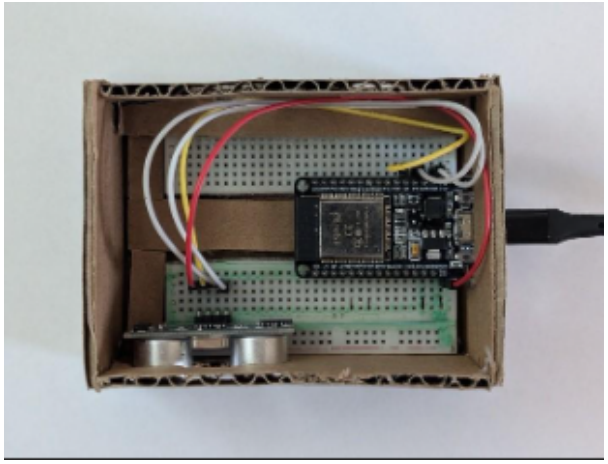
דו"ח פרויקט IOT

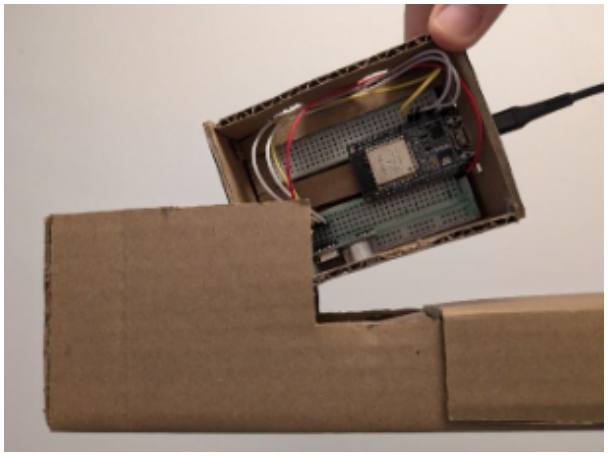
מגישים: בועז מרון, זוהר אברמוביץ, ליבי קוגן

חלק 1 - חומרה:

מטרת החומרה היא לתת אינדיקציה האם כיסא תפוס או פנוי, ולאפשר חיבור נוח למערכת.
היא עושה זאת על ידי חיישן מרחק אולטרסוני הנמצא מעל הכיסא ומחובר לבקר.
הבקר וכל החלק האלקטרוני מכוסה כחלק מהמעמד הכולל כך שרק חיישן המרחק פונה כלפי חוץ.



חיבור חיישן מרחק לבקר **Figure 1:**



הכנסת הבקר למעמד **Figure 2:**

מעמד כולל של הבקר **Figure 3:**

תחילה, הבקר מבצע התחברות לרשת.
במידה ואינו מוצא רשת שהוא מכיר, הוא פותח רשת בעצמו דרכה אפשר:

1) לחבר אותו לרשת חיצונית.
2) להחליט לאיזה קרון הוא ישלח מידע.

במידה והרשת מתנתקת, החיישן יחזור למצב בו הוא פותח רשת משל עצמו ויחכה שיחברו אותו לרשת חיצונית.
לאחר מכן הבקר דוגם מרחק ומבצע לוגיקה הקובעת האם הכיסא תפוס או לא. את הערך המתקבל הוא שולח ל-Firebase במקום המתאים לקרון הייעודי. בנוסף, החיישן שולח timestamp, כך שניתן יהיה לדעת כאשר הוא אינו מחובר לזמן מה.
ערך המספר הסידורי של הקרון אליו הבקר שולח את המידע נמצא על הזכרון ארוך הטווח של הבקר. במקרה של נפילת רשת, החיישן יידע באופן זה לאיזה קרון לחזור.
ניתן לשנות את הקרון הייעודי גם דרך ה- Firebase, כך שהשינוי יתבצע מיידית וישנה גם את הערך בזכרון ארוך הטווח של הבקר.

חלק 2 - תוכנה

בחלק זה נתאר אתר וענן שמאפשרים גישה בזמן אמת למידע על רכבות, כולל תצוגת זמני יציאה, יעדים וזמינות מקומות ישיבה.

תיאור מערכת הענן

תשתית הענן של הפרויקט מבוסס על Firebase והיא משמשת כגשר בין חיישני החומרה לבין ממשק המשתמש המערכת אחראית על אחסון הנתונים, סנכרונם, ועדכון בזמן אמת בין כל רכיבי המערכת.

תפקידי המערכת המרכזיים:

- קבלת נתונים מהחיישנים המותקנים בקרונות הרכבת ואחסונם בענן.
- העברת הנתונים בזמן אמת לממשק המשתמש.
- שמירה על מבנה נתונים מאורגן של רכבות, קרונות, מושבים ולוחות זמנים.
- מתן אפשרות למנהל המערכת לעדכן לוחות זמנים, להוסיף רכבות או לשנות את מבנה הקרונות

זרימת הנתונים

- כל חיישן שולח כל שתי שניות מידע על מצב המושבים בקרון – האם הם פנויים או תפוסים.
- יחידת הבקרה מזהה לאיזה קרון שייך כל חיישן ושולחת את הנתונים בהתאם.
- Firebase מקבלת את הנתונים ומעדכנת את רשומות הקרון והמושבים במסד הנתונים.
- ממשק המשתמש מציג את הנתונים בזמן אמת למשתמשים.

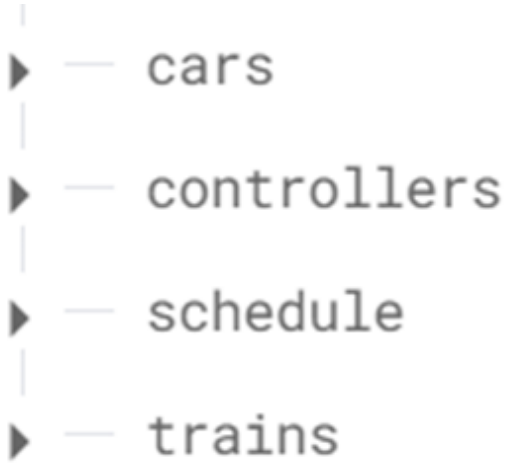
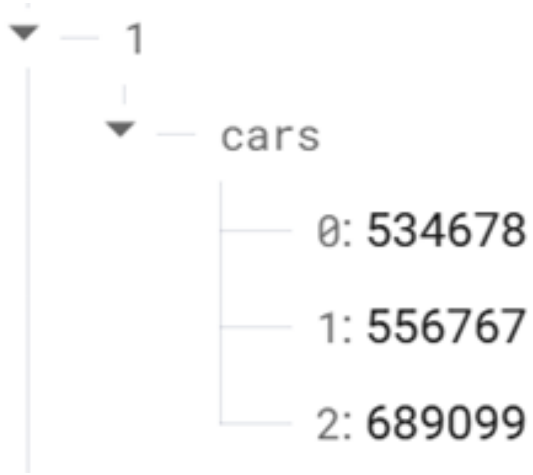


Figure 4: מבנה מסד הנתונים

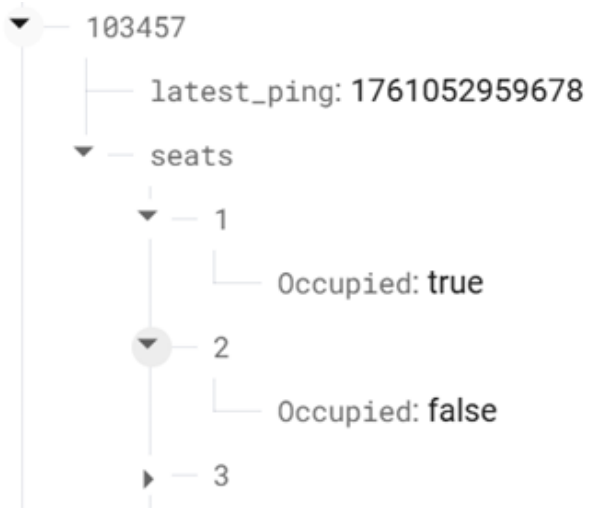
1. **אוסף רכבות**
   - כולל רשומה לכל רכבת לפי מזהה ייחודי .
   - כל רכבת מכילה מערך של קרונות, כך שניתן לשנות את ההרכב בקלות (להעביר קרונות בין רכבות).



אוסף רכבות **Figure 5:**

2. אוסף קרונות

- לכל קרון יש מזהה ייחודי ושדות הכוללים:
- זמן העדכון האחרון (המועד שבו התקבלו נתונים אחרונים מהחיישן).
- רשימת מושבים, כאשר לכל מושב יש שדה המציין אם הוא פנוי או תפוס .



אוסף קרונות Figure 6:

3. אוסף לוחות זמנים

- מאורגן לפי תאריך
- לכל תאריך רשימה של זמני יציאה של רכבות
- לכל זמן יציאה מאוחסנים - מספר הרכבת, יעד הנסיעה, מספר הרציף

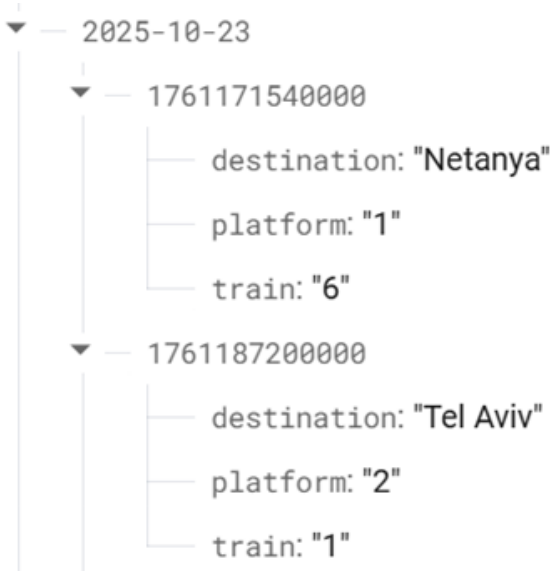


Figure 7: אוסף לוחות זמנים

4. נתוני יחידת הבקרה

במסד הנתונים נשמר גם המידע על איזה קרון שולחת יחידת הבקרה את הנתונים כעת, כדי לוודא שיוך מדויק של המידע.

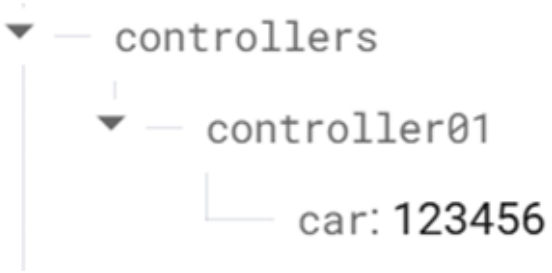


Figure 8: נתוני יחידת הבקרה

יתרונות המבנה

- גמישות: מאפשר עדכון קל של לוחות זמנים, הוספת רכבות והעברת קרונות בין רכבות.
- יעילות: ממשק המשתמש מאזין רק לשינויים הרלוונטיים (למשל רק לקרונות המוצגים כרגע), ובכך מפחית עומס ומייעל ביצועים.
- הרחבה עתידית: מבנה הנתונים המודולרי מאפשר הוספת חיישנים, רכבות או שדות מידע חדשים בעתיד.

תיאור ממשק המשתמש

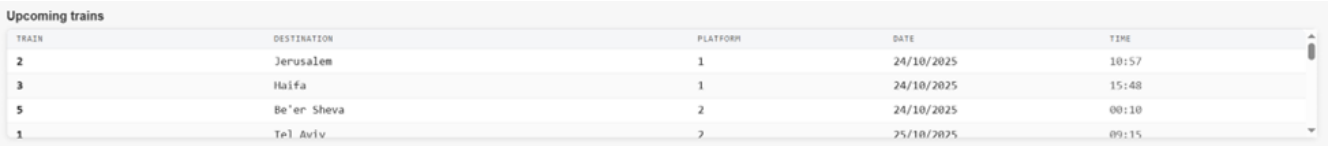
ממשק המשתמש של הפרויקט פותח כאתר אינטרנט המתארח ב־Firebase Hosting . האתר מספק פלטפורמה נגישה וידידותית למשתמש, המאפשרת לנוסעים ולמנהלים לצפות בלוחות הזמנים ובמצב המושבים בזמן אמת.

מאפיינים עיקריים:

- מציג לוח זמנים של נסיעות רכבת קרובות בצורה טבלאית.
- מאפשר צפייה במצב התפוסה של מושבים לפי רכבת וקרון, בשני מצבי תצוגה.
- מתעדכן בזמן אמת בהתאם לשינויים בנתונים המתקבלים מהענן.
- מותאם לשימוש גם ממחשבים וגם ממכשירים ניידים.

תצוגת לוח הזמנים

- העמוד הראשי מציג טבלה של 20 הרכבות הקרובות.
  - במסד הנתונים מאוחסנות 6 רכבות לכל יום, אך המערכת משלבת נתונים ממספר ימים קדימה.
- כל שורה בטבלה מציגה:
  - מזהה רכבת
  - יעד הנסיעה
  - תאריך ושעת היציאה
  - מספר הרציף
- כל שורה ניתנת ללחיצה, ובלחיצה מוצגת תצוגה מפורטת של הרכבת.



Upcoming trains

TRAIN	DESTINATION	PLATFORM	DATE	TIME
2	Jerusalem	1	24/10/2025	10:57
3	Haifa	1	24/10/2025	15:48
5	Be'er Sheva	2	24/10/2025	00:10
1	Tel Aviv	2	25/10/2025	09:15

Figure 9: תצוגת לוח הזמנים

תצוגת רכבת מפורטת

לאחר בחירת רכבת, ניתן לצפות בה בשני מצבים:

1. תצוגת קרונות
   - מציגה את כל קרונות הרכבת בצבעים לפי כמות המושבים הפנויים:
     - ירוק – הרבה מושבים פנויים
     - כתום – מספר מוגבל של מקומות
     - אדום – כמעט מלא
     - אפור – לא התקבלו נתונים לאחרונה (ייתכן תקלה או הפסקת שידור).
   - מאפשרת סקירה מהירה של מצב הקרונות.

Figure 10: תצוגת רכבת מפורטת בתצוגת קרנות

2. **תצוגת מושבים (Seat View)**
   - מציגה את סידור המושבים בכל קרון.
   - כל מושב צבוע לפי מצבו:
     - ירוק – פנוי
     - אדום – תפוס
     - אפור – לא התקבלו נתונים לאחרונה (ייתכן תקלה או הפסקת שידור).
   - בשתי התצוגות מוצגת מקרא צבעים המסביר את משמעות הצבעים.

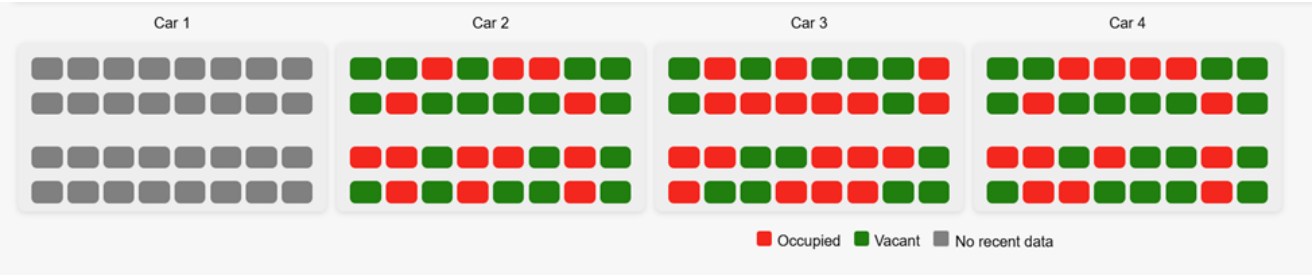


Figure 11: תצוגת רכבת מפורטת בתצוגת מושבים

אפשרויות שליטה

- כפתורי תצוגה: מאפשרים מעבר בין תצוגת קרונות לתצוגת מושבים.
- שדה חיפוש: מאפשר להזין מספר רכבת ידנית במקום לבחור מהטבלה.
- כפתור "בדוק": בודק את המספר שהוזן ומציג את נתוני הרכבת המתאימה.
- כפתור "הרכבת הקרובה ביותר": מציג באופן אוטומטי את הרכבת הבאה לפי לוח הזמנים.

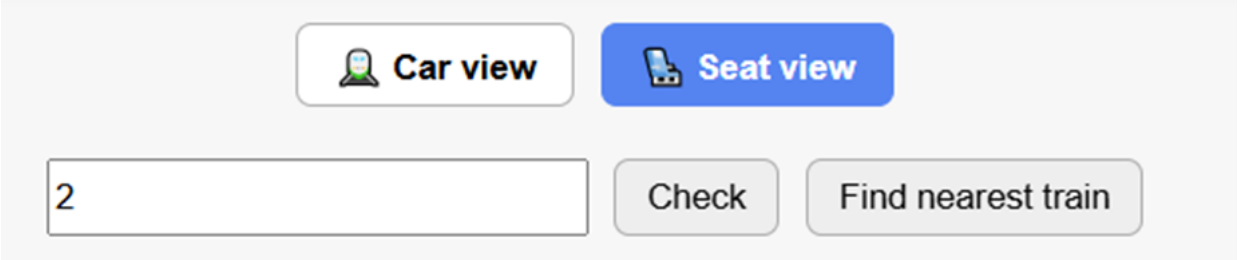


Figure 12: כפתורי תצוגה ושדה חיפוש

מאפיינים נוספים

- בעת בחירת רכבת, הממשק מציג:
  - מידע כללי על הרכבת (יעד, זמן יציאה, רציף).
  - המלצה על קרון לעלייה – המערכת בוחרת באקראי קרון שבו יש לפחות מושב אחד פנוי, כדי לעודד פיזור נוסעים שווה.

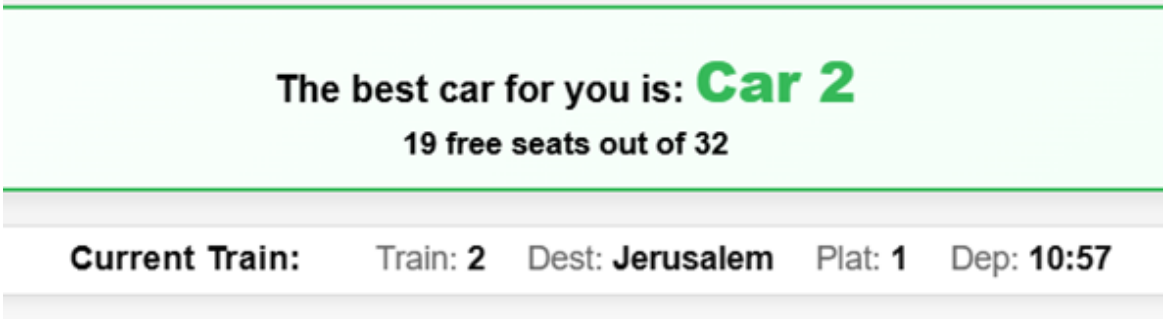


Figure 13: המלצה על קרון לעלייה

- **כאשר רכבת עוזבת את התחנה:**
  - הטבלה מתעדכנת אוטומטית.
  - אם המשתמש צפה באותה רכבת, תופיע הודעה שהרכבת יצאה.
  - התצוגה וההמלצה מתאפסות מהמסך.

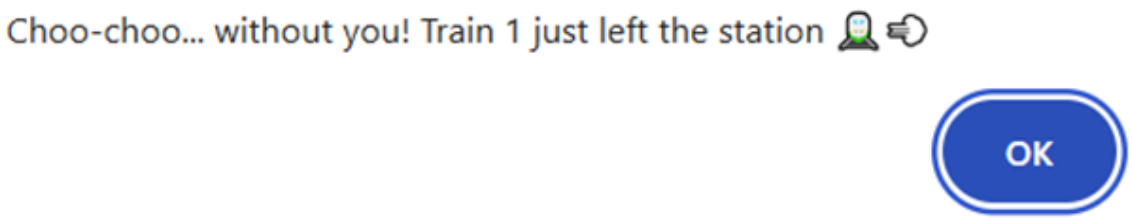


Figure 14: ודעה שהרכבת יצאה

עדכונים בזמן אמת

- האתר מאזין לשינויים גם בנתוני הרכבת הנוכחית וגם בלוח הזמנים.
- אם חיישן מפסיק לשדר במשך יותר מ־6 שניות, הקרון יוצג בצבע אפור.
  - הבדיקה מתבצעת כל 6 שניות, כך שבתוך לכל היותר 12 שניות המשתמש יראה את השינוי.

התאמה לנייד

- האתר ריספונסיבי ומותאם למכשירים ניידים.
- עיצוב הרכיבים הותאם למסכים קטנים, כך שניתן לבדוק את נתוני הרכבות והמושבים גם מהטלפון בקלות.

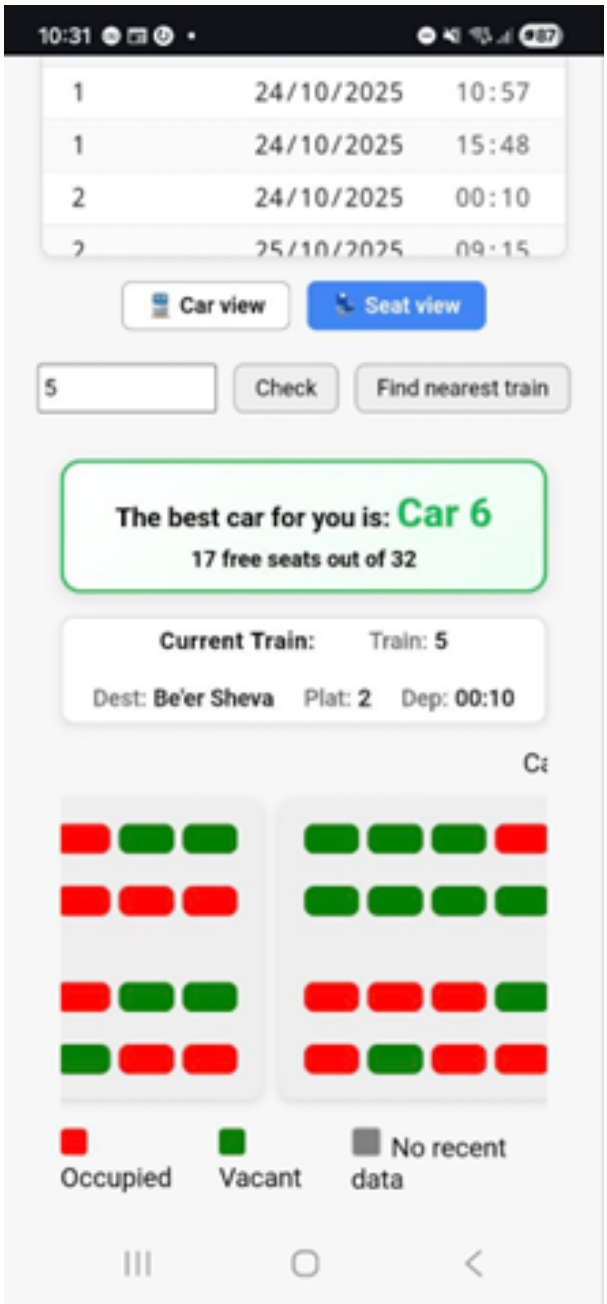


Figure 15: אתר מטלפון

חלק 3 - מבנה הסימולציה

אופן פעולת הסימולציה:

מטרת הסימולציה היא למדל את התפלגות התור של הנוסעים המחכים לרכבת ברציף עבור תחנת רכבת נתונה.

נושא זה כבר קיים בשם $Dynamic\ crowding$ והינו נושא מורכב שדורש ידע מתקדם ע"מ לצלול לעובי הקורה.
אם זאת, לאחר חקר ברשת, מצאנו מודל שמתאר באופן ראליסטי (עד כמה שניתן) את ההתפלגות.
לא נעמיק רבות על הפרטים הטכניים (שיופיעו במסמך שנצרף), אולם ננסה להסביר את האינטואציה מאחורי המודל שבחרנו.

ראשית, כל נוסע n הוא אינדיבידול בעל שני תכונות - ההעדפה להליכה וההעדפה להמתנה. לדוגמא:

נוסע **1** (אדם סבלני "ששונא" ללכת) עשוי לקבל:

- $\beta_{walk} = -10$
- $\beta_{wait} = -1$

נוסע 2 (אדם חסר סבלנות שמעדיף ללכת) עשוי לקבל:

- $\beta_{walk} = -2$
- $\beta_{wait} = -8$

לאחר מכן כל נוסע צריך להחליט איזה תור הכי כדאי. הוא עושה זאת על ידי חישוב "ציון כדאיות" (שנקרא תועלת שיטתית או V_{in}) עבור כל תור שהוא רואה, כאשר i מייצג כניסה מסויימת לקרון ברציף.
הנוסחה הבסיסית לציון הזה היא:
(ההעדפה האישית שלי להליכה $\times$ מרחק ההליכה) + (ההעדפה האישית שלי להמתנה $\times$ אורך התור)
$V_{in} =$

כעת, בהינתן נוסע n והתורים הזמינים לנוסע, נסמן בC_n אנו מעוניינים להגדיר את ההסתברות שהנוסע יבחר בתור $i \in C_n$ בהתחשב כמובן ב"ציון הכדאיות" V_{in}, כלומר $P(i|C_n)$.
המודל שנבחר נקרא $Mixed\ Logit\ (MIXL)\ Model$ והוא מגדיר את ההסתברות באופן הבא:

$$P(i|C_n) = \int \frac{\exp(\mu V_{in}\beta_n)}{\sum_{j\in C_n}\exp(\mu V_{jn}\beta_n)} f(\beta_n|\theta)d\beta_n$$

כאשר θ מכיל שני דברים חשובים עבור כל מאפיין (כמו 'עדיפות להליכה' או 'עדיפות לתור'):

1. העדפה ממוצעת $\overline{\beta_k}$ (β_k מייצג מאפיין כלשהו): מהי ההעדפה הממוצעת של כלל הנוסעים למאפיין הזה? (למשל, בממוצע, כמה אנשים שונאים לחכות בתור).

2. סטיית תקן σ_k: כמה גיוון יש בהעדפה הזו באוכלוסייה? האם כולם כמעט אותו דבר, או שיש שונות גדולה? (למשל, האם כולם שונאים לחכות באותה מידה, או שיש אנשים סבלניים ואנשים חסרי סבלנות?).

θ הוא בעצם "תעודת הזהות" הסטטיסטית של העדפות הנוסעים עבור התכונות השונות.

הפונקציה $f(\beta_n|\theta)$ היא "מתכון" ליצירת העדפה אישית β_n עבור נוסע בודד n, בהתבסס על הפרופיל הסטטיסטי של האוכלוסייה θ.

f היא פונקציית צפיפות ההסתברות (probability density function). היא מגדירה את הצורה של התפלגות ההעדפות. בסימולציה בחרנו להשתמש בהתפלגות נורמלית.

בפשטות, $f(\beta_n|\theta)$ עונה על השאלה: "עבור נוסע n, מה הסיכוי שיהיה לו בתכונה β_n ערך מסוים x מתוך סה"כ האפשרויות של תכונה β_n? " לדוגמא, מה הסיכוי שעבור נוסע מסויים n יהיה לו $\beta_{walk} = -1$ מתוך כלל הערכים של β_{walk} בפיזור הנוסעים?'

הפרמטר האחרון שנרחיב עליו (בכך נסיים את החלק הטכני של הסבר הנוסחה) הוא μ.

μ מייצג את רמת הרציונליות של כל נוסע. כפי שראינו לכל נוסע יש אישיות שנקבעת ע"פ שני מאפיינים שקבועות את ה"אג'נדה" לבחירת התור. ככל שהנוסע יהיה רציונלי יותר הוא ידבוק באג'נדה שלו לבחירת התור - כלומר, עבור הנוסע הרציונלי בהינתן תנאים שווים הוא **תמיד** יבחר את אותו התור, באופן מעשי המודל הופך להיות דטרמיניסטי.

באופן סימטרי, עבור נוסע שכלל אינו רציוני תנאי ההתחלה אינם משפיעים ולכן הבחירה שלו תמיד תיהיה באופן מעשי בהתפלגות שווה.
עבור $\mu = 0$ הנוסע כלל אינו רציונלי ועבור $\mu \to \infty$ הנוסע רציונלי לחלוטין.

לאחר ההסבר כללי על אופן פעולת הסימולציה נוכל לשלוט על פרמטרים כדי להשפיע על התפלגות האנשים בתורים בתחנה. המטרה שלנו כמובן היא תחילה, להזין פרמטרים שמייצגים התפלגות של נוסעים ללא השפעות חיצוניות, ושנית להזין פרמטרים שמציגים השפעות חיצוניות על הנוסעים על מנת לדמות השפעה חיצונית כגון אפלקציה ששולחת לכל נוסע את מיקום הקרון המומלץ עבורו ברציף. לאחר מכן נרצה לבחון את ההבדלים בינהם.

מבנה הסימולציה:

האתגר של יצירת הסימולציה הוא בין היתר ייצוג וויזאולי של מבנה התחנה, שמכילה את הכניסות לתחנה ואת איזורי ההמתנה לקרונות (אשר שם נוצר התור). מידע זה יכול להיות מיוצג בטבלה אך האינטואציה למבנה התחנה ואיך התורים מתפלגים נעלמת חיש מהר. ולכן בחרנו לייצג אותה באופן וויזאולי בסגנון משחק.

כפי שניתן לראות בתמונה, לסימולציה קיימים שני מצבים - בניית התחנה והרצאת הסימולציה על התחנה.

Figure 16: חלון הפתיחה

במצב של בניית התחנה, ניתן לעצב ולבנות אותה, הגשה העיקרי הוא מיקום המדרגות שמסמנות את מיקום כניסת התחנה אשר ממנה מתחילים אנשים ללכת ומיקום ההמתנה לקרונות הרכבת שמסמן היכן עומד להיווצר תור.

ניתן לשמור באופן מקומי ולפי שם נבחר מספר תחנות שונות ולטעון אותן לאחר מכן. הרעיון הוא לתת יד חופשית לבניית תחנת רכבת.

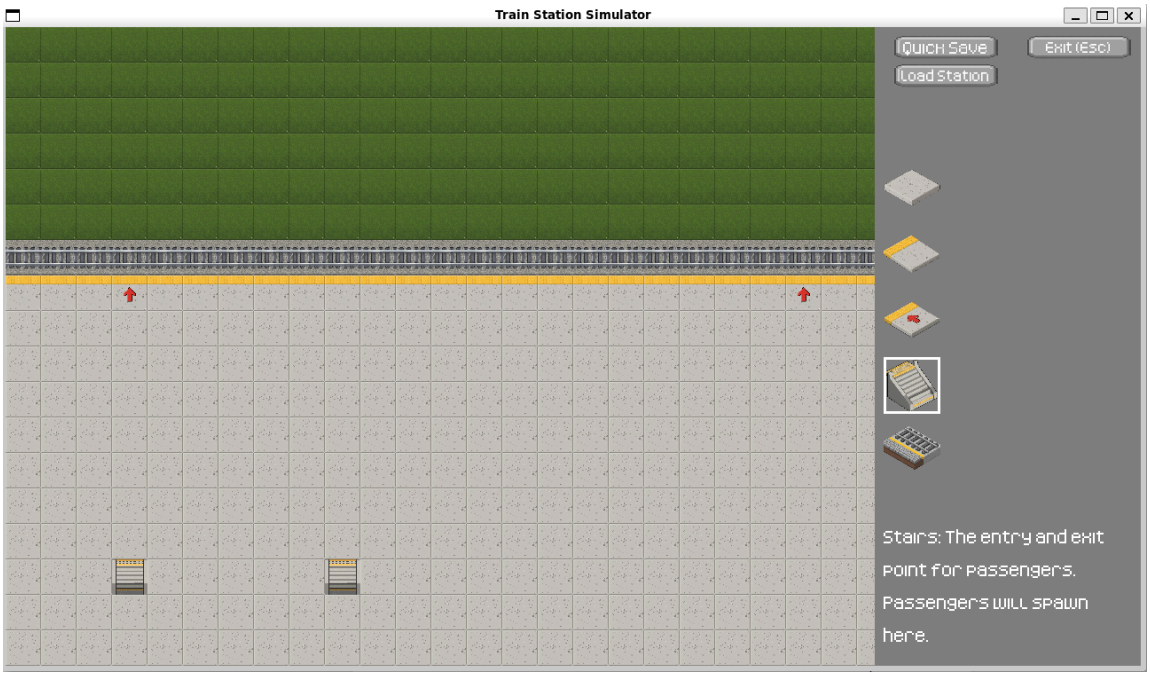


Figure 17: עיצוב ובנייה של התחנה

במצב סימולצייה מכינים את ראשית את התחנה לפני ריצת הסימולציה ע"י כיול שלושה פרמטרים:
הפרמטר הראשון והכי פשוט דורש לבחור גרם מדרגות (באיור בחרנו את השמאלי) שינוי כמות הנוסעים שיצאו ממנו (ערכים בין 0 ל100) לאורך הריצה.

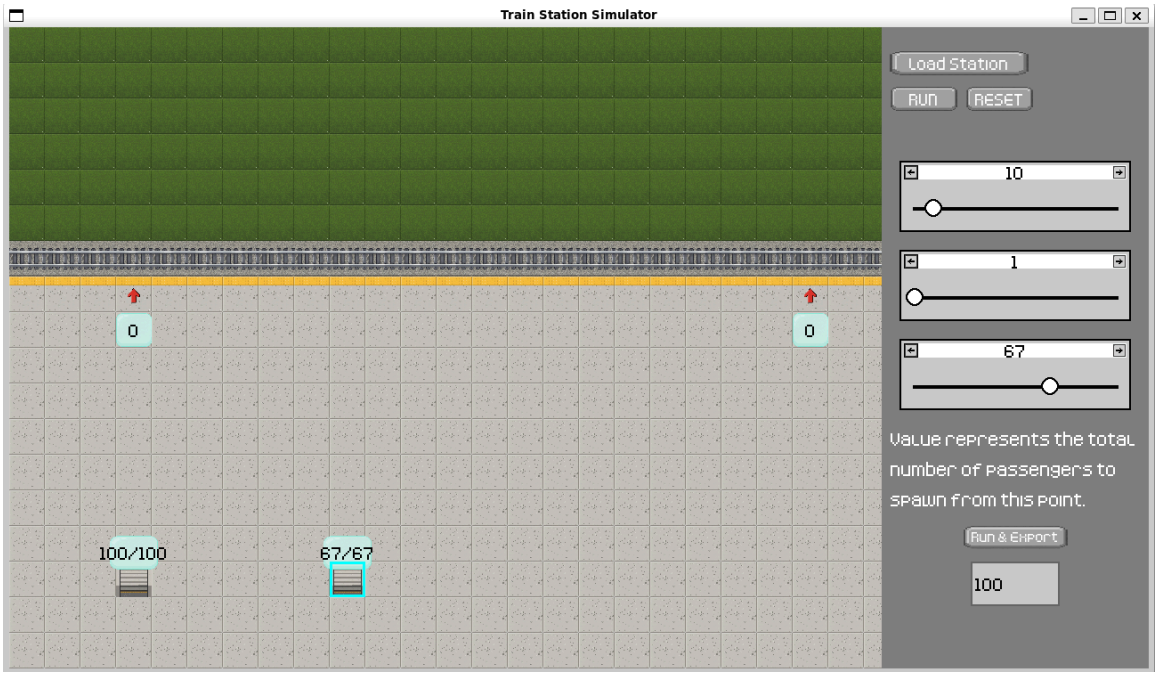
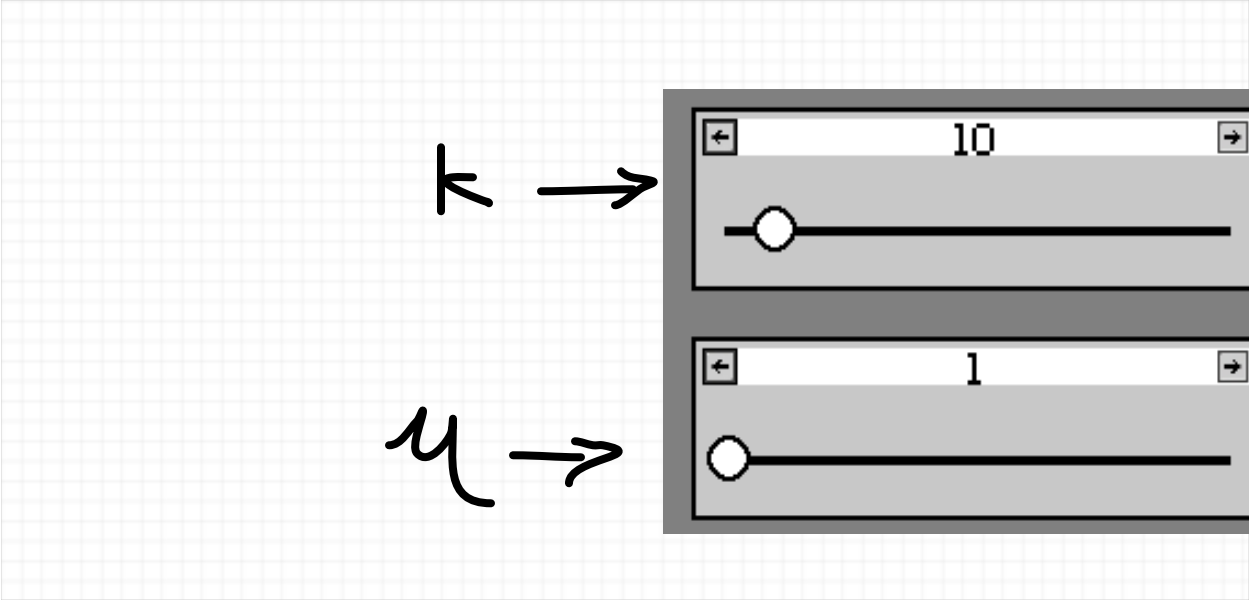


Figure 18: הכנה לריצת הסימולציה

שני הפרמטרים האחרים מתחברים להסבר על המודל של הסימולציה.
הראשון הינו μ שכפי שכבר הוזכר משפיע על מידת הרציונליות של הנוסעים ($\mu = 0$ מייצג חוסר רציונליות מוחלט).

עבור הפרמטר השני, בסימולציה תמיד מתקיים $\beta_{walk} + \beta_{wait} = -100$ (שהוזכרו בהתחלה) כאשר עבור הפרמטר שנסמנו בk מתקיים $\beta_{wait} = -k$ ולכן $\beta_{walk} = k - 100$.
כלומר כמה הנוסעים מעדיפים ללכת מול לחכות בתור **בממוצע**. ($k = 0$ מייצג חוסר סבלנות לתורים ארוכים וההעדפה ללכת רחוק יותר עבור תור קצר).



לאחר בחירת הפרמטרים ותנאי ההתחלה, המשתמש יכול לבחור את כמות הפעמים שהסימולציה תתבצע.
התוצאות ישמרו בקובץ csv .

לסיכום, עברנו על המודל התיאורתי של הסימולציה וכן גם על מבנה הסימולציה - בכך תם חלק זה.

חלק 4 - דו"ח מסכם:

עתה, לאחר שהתעמקנו בפרטי הפרויקט, נבחן את שיפור זמן ההמתנה שחל מרגע הכנסתו לתחנת הרכבת.

מבנה הרכבת יהיה באופן הבא:

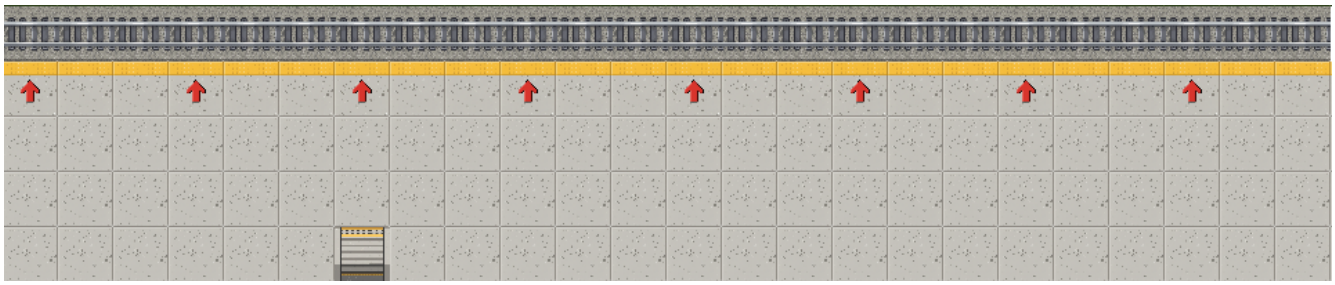


Figure 19: Station layout

בדומה לתחנת רכבת באוקסופרד שמופיעה במאמר *Understanding Users' Behaviours in" "Relation to Concentrated Boarding: Implications for Rail Infrastructure and Technology*

במאמר מופיעה פריסה של התחנה עם התפלגות הנוסעים ברציף לפי כל קרון:

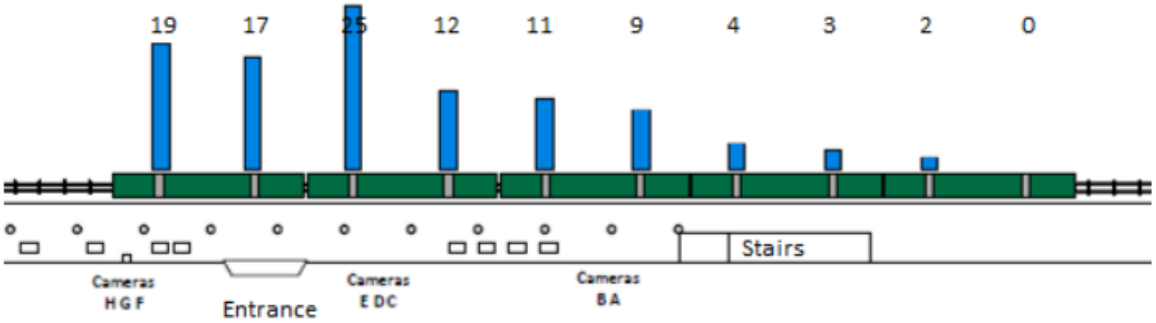


Figure 20: Passengers distribution

כיילנו את הפרמטרים $\mu = 3$ ו$k = 10$ כך שבסימולציה תתבצע פריסה דומה בממוצע:

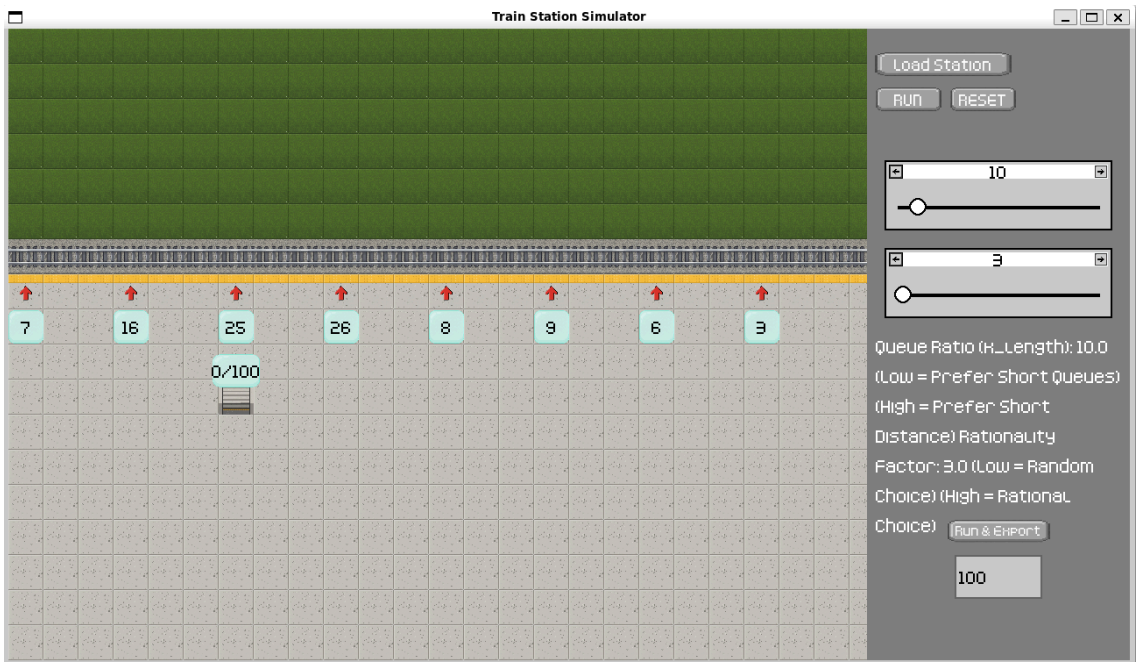


Figure 21: One run exmaple

לאחר 100 איטרציות נקבל התפלגות ממוצעת של מספר הנוסעים בכל תור בתחנה:

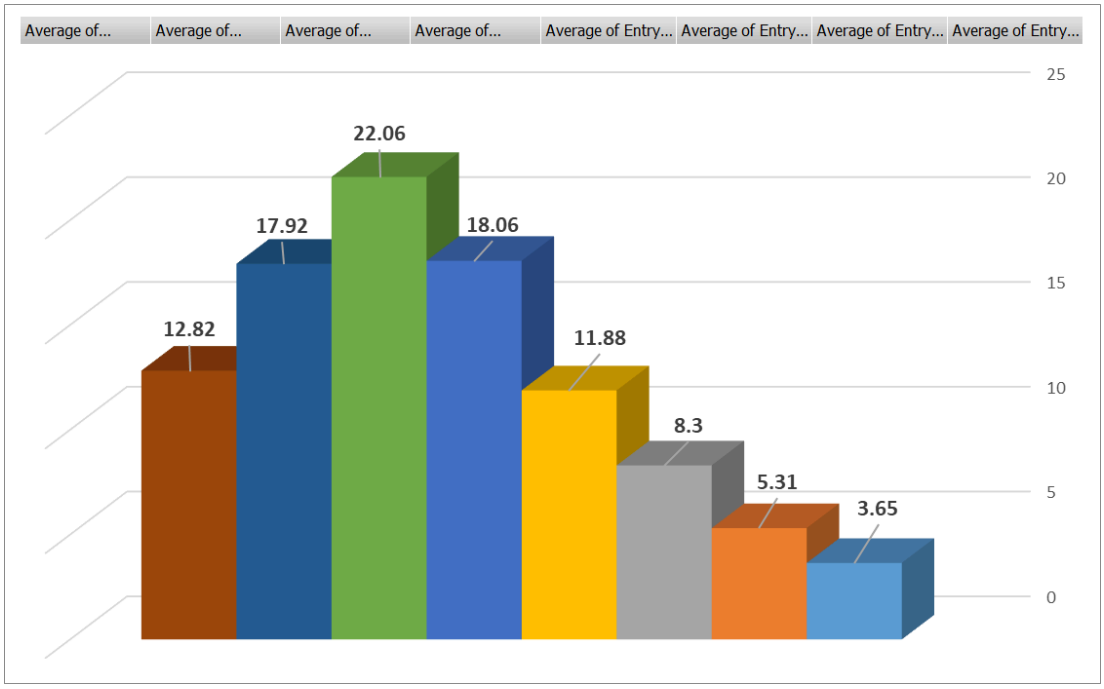


Figure 22: Average distribution

נבחין כי זמן העליה הכולל לרכבת נקבע לפי התור הכי ארוך ולכן אם נניח שלכל נוסע לוקח בממוצע 4 שניות לעלות לרכבת נקבל כי הזמן לעלייה לרכבת בתחנה הנוכחית היא כ$88 = 4 \times 22$. קיבלנו כי זמן ההמתנה של הרכבת בתחנה הוא **כדקה וחצי**.

לאחר שהראנו כי לקבוצת הבקרה לוקח כדקה וחצי לעלות לרכבת בממוצע, נראה כיצד הפרוייקט שלנו מקצר את זמן ההמתנה. כפי שכבר הסברנו בחלק השני, האפלקציה ממליצה מושב לנוסע בהתפלגות אחידה ממספר המושבים הפנויים (כדי למנוע מצב שהאפלקציה תשבץ את כולם לכל קרון).
אם נניח שהנוסעים קשובים לחלוטין לאפלקציה ומתעלמים לחלוטין מההעדפות האישיות שלהם ("עדיפות להמתנה" ו"עדיפות להליכה") נקבל כי באופן מעשי ניתן לסמלץ את אופן הבחירה שלהן ע"י איפוס רמת הרציונליות שלהם, קרי $\mu = 0$, זאת מכיוון שרמת הרציונליות קובעת כמה הם עקביים אחרי העדפות שלהם, אך היא לחלוטין אינה רלוונטית מכיוון שהם בוחרים את התור לפי המלצת האפליקציה. אולם, בחרנו לקבוע $\mu = 1$ עקב העובדה שאנשים אף פעם אינם מתעלמים מהעדפות האישיות שלהם לחלוטין.

כעת נראה את תוצאות הריצה עבור $u = 1$ ו $k = 10$:

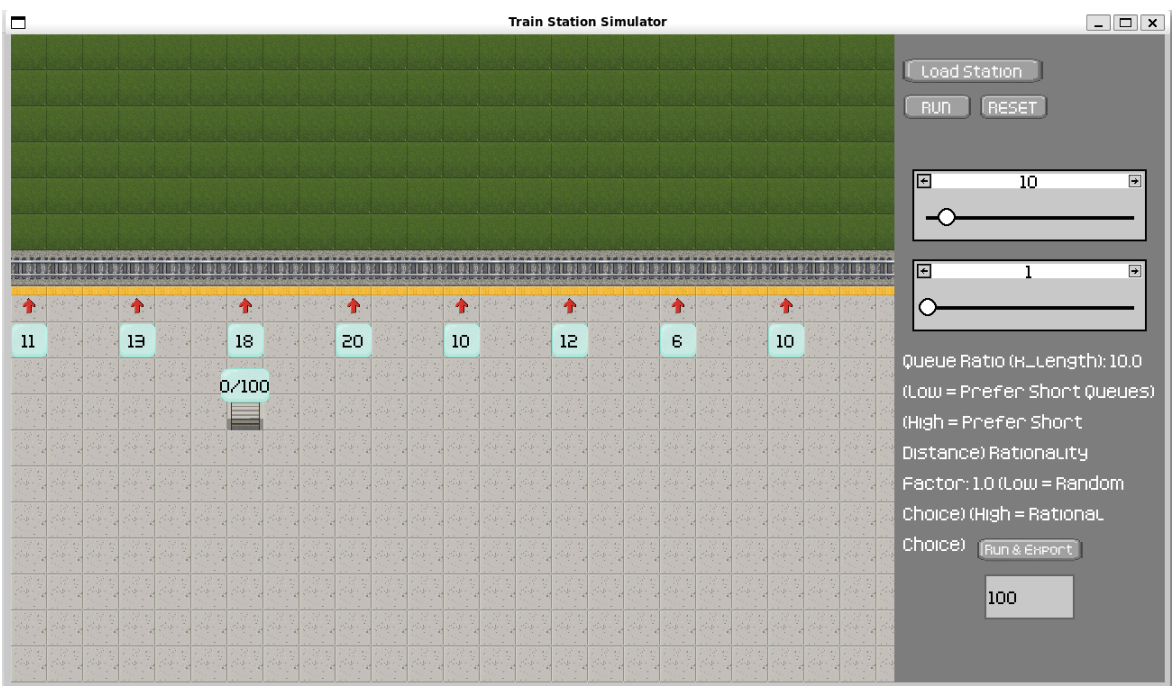


Figure 23: One run exmaple

לאחר 100 איטרציות נקבל התפלגות ממוצעת של מספר הנוסעים בכל תור בתחנה:

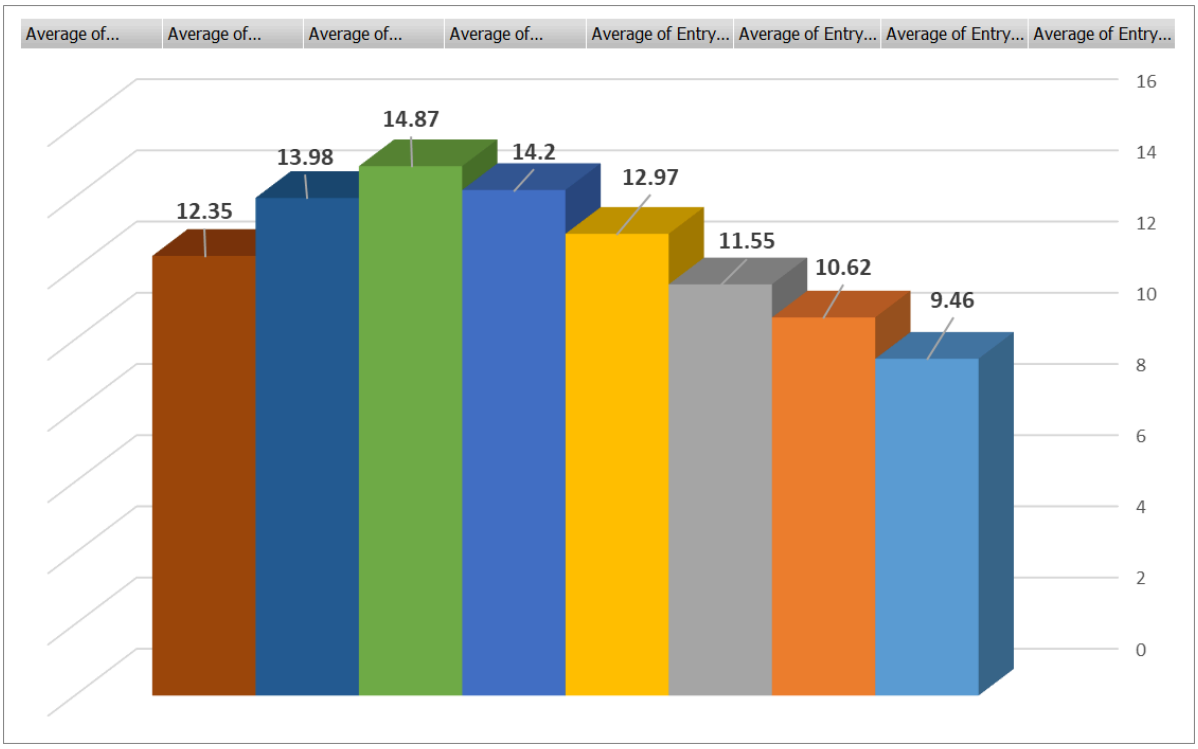


Figure 24: Average distribution

לפי אותו חישוב שעשינו לקבוצת הבקרה נקבל כי הזמן הממוצע של הרכבת בתחנה הוא כ$60 = 15 \times 4$.

המסקנה היא שעבור תחנת הרכבת באוקספורד (תחת ההנחות המתאימות) נקבל זמן ההמתנה של הרכבת מתקצר בחצי דקה או לפי אחוזים ב33% מזמן ההמתנה המקורי.