

## התקנת חבילות:

החבילות שהותקנו הן:

1. Body- parser
2. Csvtojson
3. Express
4. Mysql
5. pug

## עמודי HTML:

signUp.pug - העמוד הראשון באפליקציה, במידה והמשתמש לא רשום לערכת הוא מזין שם משתמש וסיסמא, לוחץ על Join ונכנס לעמוד הבית, במידה והמשתמש רשום הוא עובר למסך Sign In.

signIn.pug - משתמש שכבר רשום למערכת מגיע לעמוד זה ומזין את שם המשתמש והסיסמא שלו ולוחץ על Login, מתבצעת בדיקה שאכן המשתמש קיים במערכת והוא עובר לעמוד הבית. במידה והמשתמש שכח את סיסמתו הוא יכול ללחוץ על Update password ולעבור לעמוד שבו יוכל לעדכן אותה.

updatePassword.pug - המשתמש מזין את שם המשתמש שלו וסיסמא חדשה ולוחץ על Update, קופץ לו חלון שהסיסמא עודכנה והסיסמא מתעדכנת בDB. המשתמש עובר לעמוד Sign In שם מזין את השם משתמש שלו עם הסיסמא החדשה שבחר.

homepage.pug - בעמוד הבית על המשתמש לבחור את סוג הרכב שלו ולאפשר גישה למיקום שלו. בכל חניון יש מספר סוגים של חניות וכאשר המשתמש ירצה לבחור חניון תתבצע בדיקה האם קיים מקום בחניון בהתאם לסוג הרכב שבחר בעמוד זה. כאשר המשתמש לוחץ על P הוא עובר לעמוד Map.

Map.pug - עמוד בו מוצג על המפה מיקום המשתמש וכל חניוני אחוזת החוף בעיר, המשתמש יוכל לראות מה החניון הקרוב ביותר אליו וללחוץ עליו, במידה ובחניון אין מקום תקפוץ למשתמש הודעה שאין לו מקום בחניון והוא יוכל לבחור בחניון אחר. בנוסף, אם המשתמש רוצה לחפש חניון על פי השם שלו הוא יכול ללחוץ על Find ולעבור לעמוד Find Parkinglot.

FindParkingLot.pug - בעמוד זה המשתמש יכול לחפש חניון על פי השם, ולקבל את הכתובת של החניון.

Leave.pug - לאחר שהמשתמש בחר חניה הוא צריך לעדכן גם כשעוזב אותה. כאשר המשתמש עוזב את החניון עליו ללחוץ על הכפתור L וכך הDB מתעדכן שיש בחניון חניה פנויה נוספת.

contactUs.pug - עמוד בו המשתמש יכול לשלוח הודעה למערכת. באמצעות מייל שיכולו לחזור אליו במקרה הצורך.

ThankYou.pug - כשאר המשתמש מסיים להשתמש באפליקציה הוא עובר לעמוד זה. כלומר, לאחר שמעדכן כי עזב את החניה וגם לאחר ששלח הודעה למערכת.

Layout.pug - עמוד המסגרת של כל העמודים באפליקציה המכיל את header והfooter.

Footer.pug - התחתית של העמוד, על ידי מי נעשה האתר.

Header.pug - navbar מכיל קישור לעמוד הראשון של האפליקציה, עמוד signUp כך השמשתמש יכול תמיד לחזור להתחלה ולהתחבר מחדש ולעמוד contact us כך תמיד יכול לגשת ולשלוח הודעה למערכת. בנוסף, navbar מכיל המבורגר כאשר משתמשים באפליקציה בmobile.

## עמודי JavaScript:

Index.js - הקובץ שאחראי על ההפעלה של הפרויקט, הניתוב והמעבר בין העמודים.

## Static:

Maps.js - קובץ יצירת המפה ודקירת מיקום המשתמש והחניונים על גביה. החניונים נמצאים בעיר תל אביב כך שבמידה ומיקום המשתמש הוא לא בתל אביב יש להזיז את המפה ולראות שם את החניונים.

## Scripts:

Main.js - כשאר המשתמש לוחץ על כפתור במסכים השונים הבקשה נשלחת לקובץ זה ומעבירה את המשתמש לעמוד הבא תוך ביצוע ולידציות במקרה הצורך.

Leave.js - קובץ עם פונקציה המעדכנת את ה DB כאשר המשתמש לוחץ על כפתור העזיבה של החניון, מוסיפה חניה פנויה לחניון על פי סוג הרכב של המשתמש.

FindParking.js - קובץ בו נמצאת קריאה לפונקציה שמוצאת את כתובת החניון לפי השם שלו כאשר המשתמש לוחץ על כפתור החיפוש.

Auth.js - קובץ בו מתבצעים האישורים בהקשרי המשתמש, כלומר, כאשר המשתמש לוחץ על כפתור בקובץ זה מתבצעת העברה לפונקציה הרלוונטית, תוך ביצוע ולידציה לקלט במקרה הצורך והעברת המשתמש לעמוד הבא.

## Routes:

apiRoutes.js - ניתוב לפי endpoint השונים תוך שימוש בפונקציות.

## DB

creatDB.js - קובץ בו נמצאות פונקציות היצירה של ה DB, ההכנסה של הערכים לטבלאות, זריקת הטבלאות במקרה הצורך והצגתן. בכדי להשתמש בפונקציות אלה יש לקרוא להן ב URL כלומר, לדוגמא בשביל ליצור את טבלת המשתמשים יש לכתוב: <http://localhost:3000/CreateUsersTable>.

db.config.js - שם ה DB של הפרויקט הוא parkmetlv.

## Controllers:

authController.js - קובץ עם פונקציות `signin`, `signup`, `update password`, כלומר אשר עובדות על DB של המשתמשים, יצירת משתמש חדש, עדכון סיסמא ובדיקה שהמשתמש אכן קיים במערכת.

parkingController.js - קובץ עם פונקציות של החניונים, כלומר מעדכנות את ה DB ושולפות ממנו נתונים, כלומר, מציאת חניון על פי שם, הצגת כל החניונים המשמש לעמוד המפה בו מוצגים כל החניונים ועדכון מספר החניות הפנויות בחניון.