הצעת פרויקט – י"ד הנדסת תוכנה

**שם הסטודנט:** גד סרוסי

**ת.ז הסטודנט:** 212913842

**שם רכז המגמה:** אלי גוריאל

**שם המנחה:** אלי גוריאל

**סמל מוסד:** 570077

**שם המוסד:** מכללת אורט סינגאלובסקי

**מקום ביצוע הפרויקט:** מכללה

IntelliFleetBus" "

AI-Powered Fleet Management



# תיאור הפרויקט

מאז שנת 1996, ה- obd2 נהווה לחלק סטנדרטי ברכבים המודרניים בעולם. חלק זה נועד בכדי לנתר ולאבחן את רכיבי הרכב בזמן אמת ולהתריע את המשתמש בתקלות שנמצאו במערכות השונות ברכב. הנתונים שסיפק חלק זה עזרו לאבחן את בעיות הרכב בקלות רבה יותר מאי פעם שהייתה בעבר, ה- obd2 עזר להתריע לנהג על תקלה קיימת ישר ברגע שנמצאה וגורמת לנהג לפנות למוסך ובכך למנוע תאונות. (במהלך התיאור נגדיר את ה obd2 בתור "הרכיב").

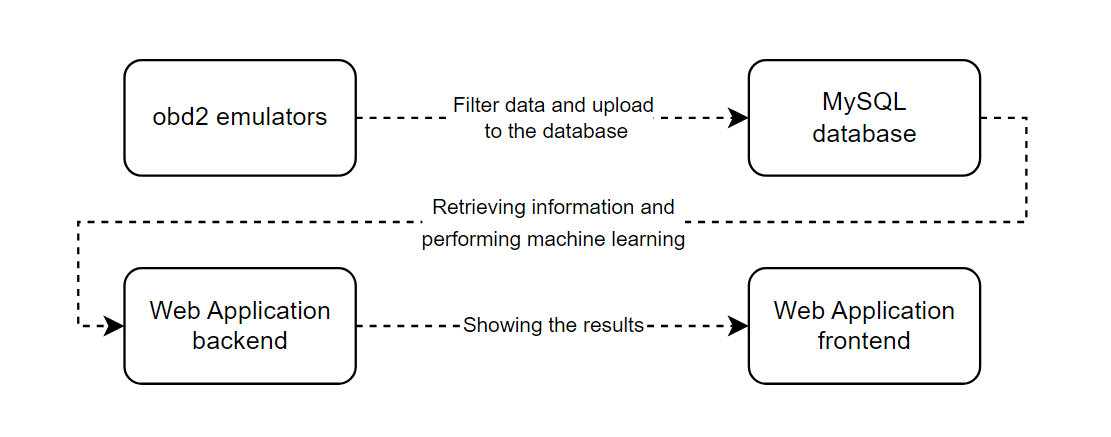
בחברות האוטובוסים כיום, קיימת עלות גבוהה בעקבות בלאי תדיר ברכיבי האוטובוס. רכיבים אלה יקרים ומהווים הוצאה גדולה לחברות. בפרויקט זה, אשתמש ברכיב הנ"ל על מנת ליצור מערכת המשמשת חברת אוטובוסים אשר תעקוב אחר שימוש הנהגים ברכיבי האוטובוס.

הפרויקט יעבוד בכך שישלוף מידע מהרכיב אשר יעבור סינון בהתאם לצורך ויישלח בזמן אמת באמצעות בקשות [HttpPost] למאגר מידע MySQL server. לאחר מכן, אשתמש ב WebApplication שמטרתה לקרוא את הנתונים מה- database ולבנות עליהם מודל למידת מכונה אשר תחזה את רמת הבלאי העתידית שהנהג יגרום לאוטובוס בעתיד ותדווח לאחראים בחברה דרך ה- WebApplication.

את ה- WebApplication ולמידת המכונה אכתוב בשפת Java.

# רקע תיאורטי

## תהליך העלאת הנתונים מה Obd2 עד להצגת תוצאות ב Frontend WebApplication



שלבים בתהליך:

1. להפעיל כמה סימולציות obd2 במקביל.
2. לסנן את המידע שמקבלים לפי מה שנרצה לעלות ל database.
3. לעלות את המידע הנחוץ למאגר המידע MySQL server.
4. להריץ קוד ב java ששולף את המידע מהמאגר מידע.
5. להשתמש במידע הזה ללמידת תוכנה ב deep learning ואימון רשת נוירונים.
6. שמירת המודל / לעדכונו שיהיה אופטימלי.
7. להציג בפני המשתמש את התוצאות.

## הבעיה האלגוריתמית:

הבעיה:

**ניבוי פוטנציאל לתקלות שונות ברכב, לפי שימוש נהג ברכיבי הרכב.**

תיאור:

מקבלים נתונים מתוך רכיב המוצמד לרכב הכוללים נתונים על שימוש ברכיבי הרכב.

דרכי הפעולה:

נבנה מודל אשר על סמך נתוני רכיבי השימוש של הנהג ברכיבי הרכב ותקלות שדווחו בעבר, יחשב פוטנציאל לתקלות ברכב.

חלק 0: נבנה רכיב תוכנה אשר יקרא נתונים מה obd2 אמולטור ויכתוב אותם לתוך טבלה שתגדיר מראש ב MySQL- חשוב!!!! בטבלה \*חייבת\* לכלול את כל עמודות שמוצא האמולוטור + עמודת קוד תקלה

חלק 1: נבנה מודל המסווג תקלות על פי נתוני ה שימוש ברכב.

חלק 2: לשפר את המודל ולברור רק נתונים שבאמת יחזו תקלות

חלק 3: לקחת נתוני test ולהפעיל עליהם את המודל

חלק 4: לשפר את המודל אחרי הפעלת ה test

# טכנולוגיות בפרויקט

* **MySQL database –** היא מערכת פופולרית לניהול מסדי נתונים בקוד פתוח. היא משמשת לאחסון וניהול נתונים בצורה מובנית, המאפשרת למשתמשים לבצע פעולות שונות על הנתונים כגון יצירה, קריאה, עדכון ומחיקת רשומות.
* **Spring –** היא מסגרת מבוססת Java לבניית יישומים. היא נועדה להקל על פיתוח יישומי Java ארגוניים, ומספקת מספר תכונות וכלים שימושיים שיכולים לעזור למפתחים ליצור יישומים במהירות וביעילות רבה יותר.
* **WebApplication –** תוכנה שהגישה אליה היא דרך האינטרנט באמצעות דפדפן אינטרנט. זהו סוג של יישום הפועל על שרת אינטרנט ומשתמשים ניגשים אליו דרך רשת.
* **TCP/IP –** (פרוטוקול בקרת שידור/פרוטוקול אינטרנט) היא קבוצה של פרוטוקולים המשמשים להעברת נתונים דרך האינטרנט ורשתות אחרות. זהו הבסיס של האינטרנט והתקן לתקשורת בין מכשירים באינטרנט.
* **OBD2 –** היאמערכת סטנדרטית לכלי רכב המאפשרת למכונאים וטכנאים לאבחן ולפתור בעיות במנוע הרכב ובמערכות נוספות.
* **IntelliJ IDEA–** היא סביבת פיתוח משולבת (IDE) עבור Java ושפות תכנות אחרות כגון Kotlin, Scala ו-Groovy. הוא פותח על ידי JetBrains, חברה הידועה ביצירת כלים עבור מפתחים.
* **Git** – מערכת לניהול גרסאות מבוזרת (distributed VCS) הפופולרית בעולם. תשתמש לנהל את הגרסאות השונות במהלך הפיתוח של הפרויקט.
* **GitHub** – שירות ניהול ואחסון אינטרנטי עבור Git repositories. יחד עם Git היא תשמש לניהול הקוד של הפרויקט לאורך הפיתוח.
* **ספריות עבור ה-backend לשפה Java:**
  + **Spring Framework**
  + **MySQL**
  + **Tomcat**
  + **Deep learning**
* **ספריות עבור ה-frontend לשפות JavaScript/Typescript:**
  + **Angular** – ספרייה ליצירת אפליקציות web שפותחה על ידי גוגל. הספרייה תשמש ליצירת האתר.

**או שאשתמש ב-**

**React** – ספרייה ליצירת אפליקציות web שפותחה על ידי פייסבוק. הספרייה תשמש ליצירת האתר.

* + **Axios** – ספריית JavaScript/Typescriptמבוססת Promises המפשיטה יצירת בקשות HTTP. תשמש לתקשורת בין ממשק המשתמש ל-API של המערכת.

## תרשים מסכים (User Flow)

**לוח זמנים**

|  |  |  |
| --- | --- | --- |
| משימה | זמן עבודה מוערך | לו"ז סיום |
| בניית ה WebApplication | 1.5 חודש | 30.1.22 |
| בניית ה Database | 0.5 חודש | 15.2.22 |
| בניית ה מודל למידת מכונה | 1.0 חודש | 15.3.22 |
| שיפור ה frontend צד לקוח | 0.75 חודש | 31.3.23 |
| שיפורים ותיקונים בכללי | 1.0 חודש | 15.4.23 |

# **אישורים**

****

**חתימת הסטודנט:**

**חתימת רכז המגמה:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**אישור ממשרד החינוך:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_