

פרוייקט מסדי ענק

1. להתקין nodejs – על מנת להקים אובייקט/שרת , אפשר דרך האתר או בלינוקס ע"י פקודות

```
sudo apt-get update
```

```
sudo apt-get install nodejs
```

```
sudo apt-get install npm
```

אחר כך על מנת לבדוק שהתקנה הצליחה להקליד בcmd

```
node --version
```

```
nodejs --version
```

בעזרת Nodejs אפשר לבצע קריאה לכל מיני מודלים בעזרת השימוש בrequire , וכך להשתמש באוספים של פונקציות שנרשמו , או שנכתוב בעצמנו.

העיקרון מאחורי require ברגע שקראנו לו מוחזר לנו אובייקט שנקרא export

אחד מהמודלים המוכרים הוא http ("require" שהוא מאפשר לנו ליצור שרת ולדבר איתו בעזרת תקשורת http שמובסת על פרוטקול שרת לקוח

אנחנו נשתמש במטלה במודולו express על מנת להתקין אותו

```
npm install express --save
```

כך נתקין אותו ואת כל המודולים שהוא תלוי בהם ואם נרצה להשתמש בו

```
var express = require('express');
```

```
var app = express();
```

כעת כל קריאה על app היא תשתמש באובייקט של express ונוכל להאזין לפורט מסוים ולפעול בהתאם

Routing.2 – כעת נוכל ליצור ראוינטג , משמע נוכל להעביר בקשות מהלקוח ולהחזיר לו את המידע כרצוננו

```
app.get('/', function (req, res) {  
  
  res.send('Hello World!');  
  
});  
  
app.post('/', function (req, res) {  
  
  res.send('POST to Hello World!');  
  
});
```

כך נטפל בבקשות שלו בעזרת `get` ו `post`

מבנה בקשת Http

שלב זה מתבצע בשכבת האפליקציה כאשר הלקוח מבקש מידע מהשרת , הבקשה נשלחת ע"ג חיבור TCP שנפתח בין הלקוח לשרת ובפרט לפורט עליו מאזין השרת . (בדפדפן לרוב זה פורט 80)

כל בקשה HTTP היא למעשה טקסט רגיל המורכב מ3 חלקים :

- שורת פתיחה
- שורת כותרת
- גוף הבקשה

שורת הפתיחה – מורכב גם כן מ3 חלקים : 1. שיטת הבקשה (`get`, `post`) כתובת המשאב המבוקש 3 גרסת הפרוטוקל.

GET /index.html HTTP/1.1

מבנה תגובת HTTP

תגובת השרת ללקוח , מורכבת מ3 חלקים כמו הבקשה :

- שורת מצב – מכילה קוד מצב וגרסת פרוטוקל
- שורת כותרת – כמו מבנה הבקשה
- גוף התגובה (לא חובה)

כעת אחרי שהבנו איך מתנהלות בקשות ותגובות http

על מנת שנוכל להשתמש בראוטינג יש צורך להגדירו בתור קובץ נוסף ולטעון אותו כמו מודל , ועל מנת להפעילו צריך להשתמש ב use , דוגמא לקובץ router

```
var express = require('express');
```

```
var router = express.Router();
```

```
router.get('/about-me', function(req, res) {
```

```
    res.send('About me page');
```

```
});
```

```
router.get('/how-to-use', function (req, res) {
```

```
    res.send('How to use');
```

```
});
```

```
module.exports = router; אינסטנס של אקספרס שמחזירה אינסטנס של ראوتر
```

ומהקובץ הראשי נוכל פשוט לעשות (app.use(/model,model כאשר ./model) var model = require(. (

המטרה של ראוטינג זה לנתב , בגלל שיש לי באתר אין ספור נתבים , כך בעזרת model נוכל לנתב בקשות מסוימות ולפתור בהתאם

Middleware

כשמו כן הוא , מתווך בין אובייקטים , כך שלמעשה כל בקשה שתעבור ב app.js והיא בדרך ל/help תעבור ב Middleware

לפני שהיא תגיע ל/help (החלק התחתון של הקוד) , וזאת מכיוון שהשתמשנו ב use .

ניתן לשים ש middleware מקבלת 3 ארגומנטים הרעיון הוא להמשיך בשרשרת הפעולות עד שהבקשה עוברת ל send , היא ממשיכה לעבור הלאה בזכות ה next() , חשוב לציין זה רץ לפי סדר הכתיבה .

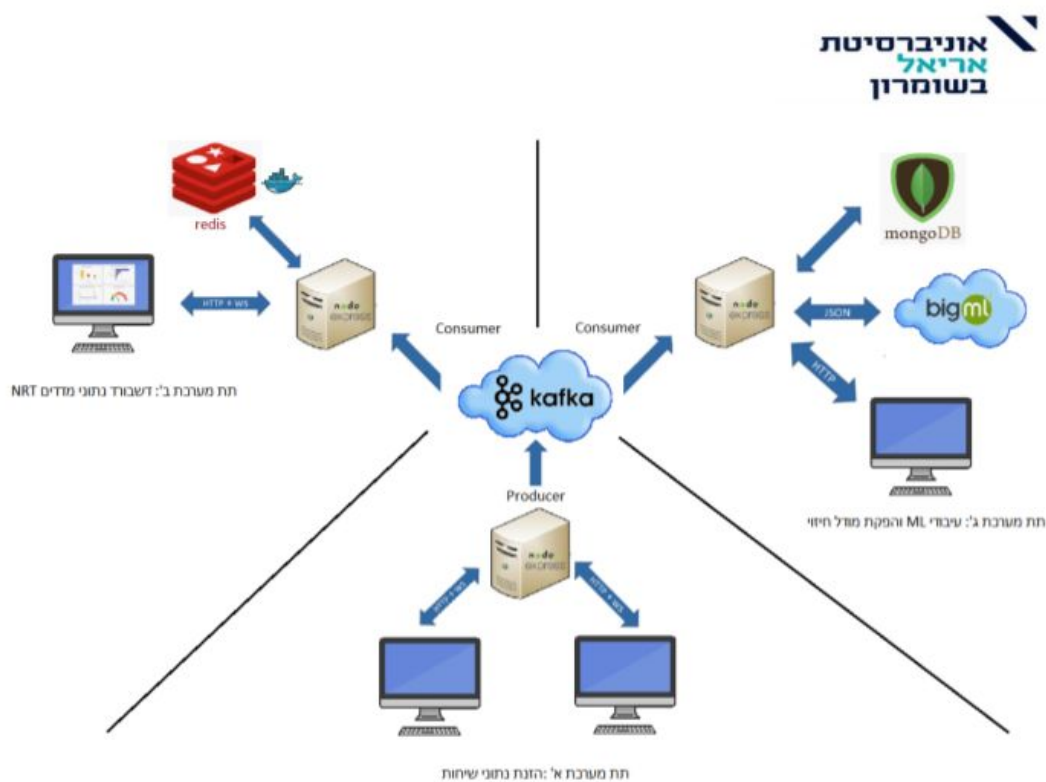
אופציות נוספות להשתמש ב middleware

Npm install bigml

npm i csvjson

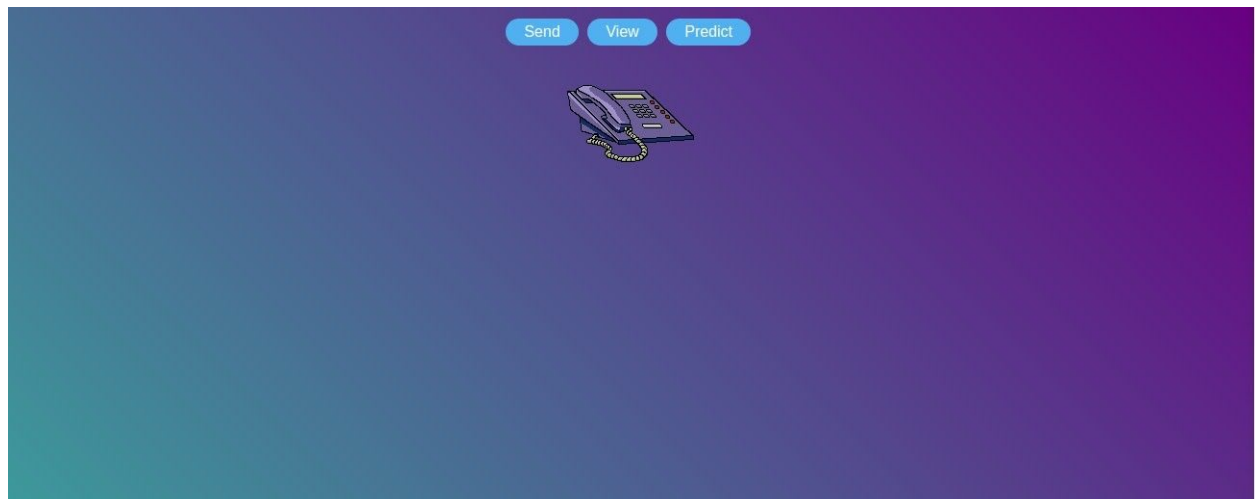
npm i fs

מבנה המטלה

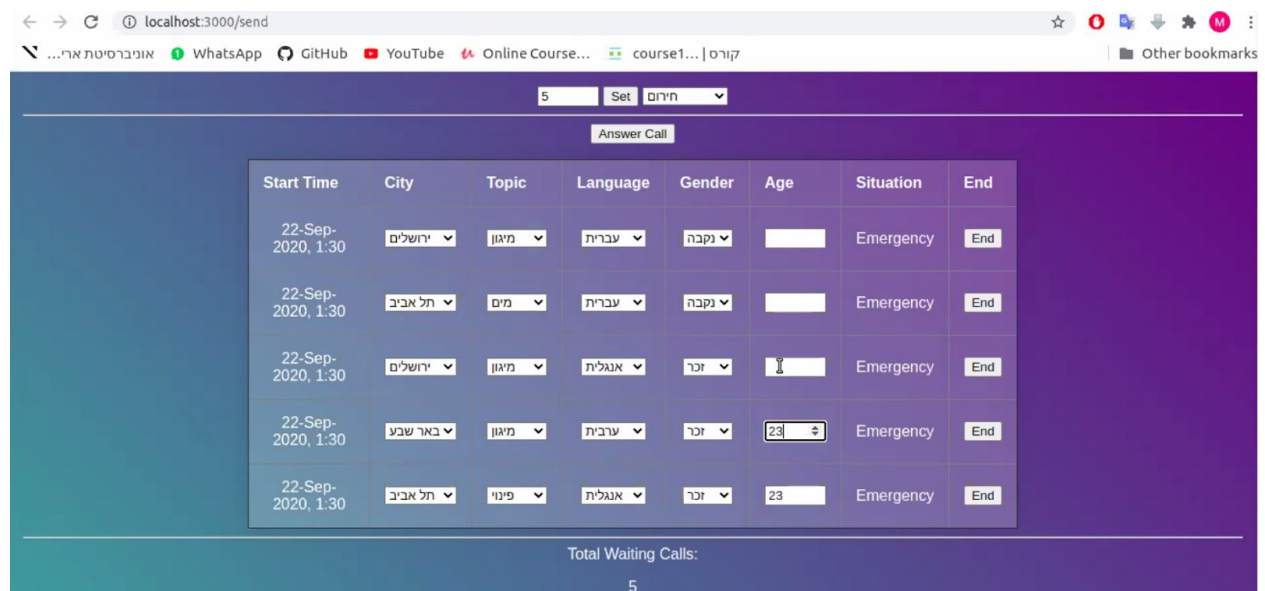


ScreenShoot:

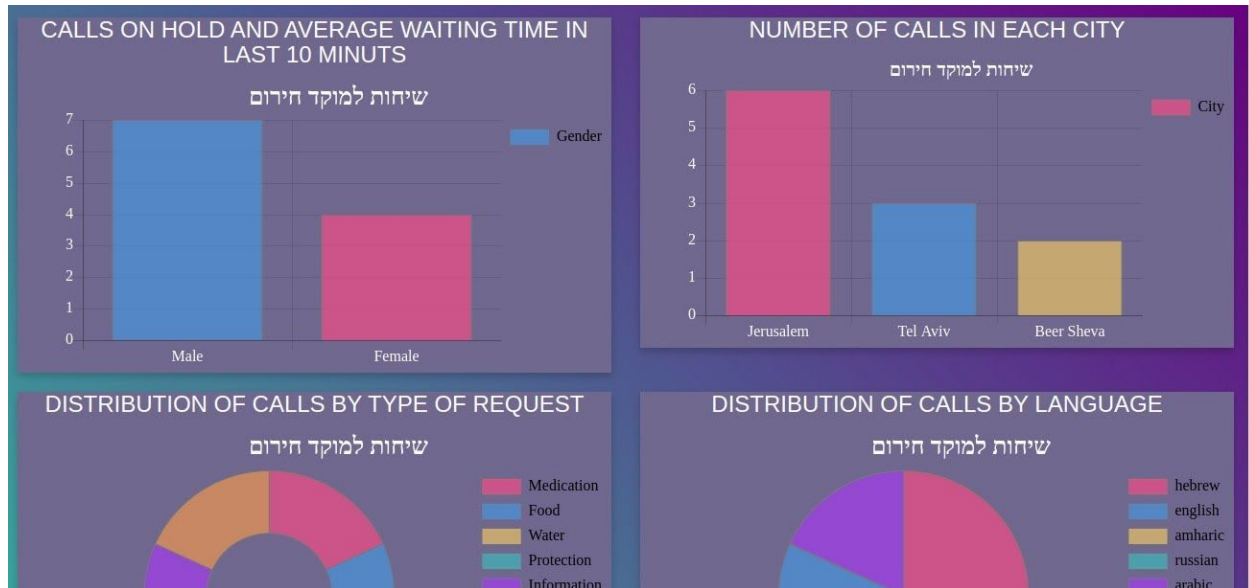
Index



Sender



Dashboard



predict

localhost:4500/predict

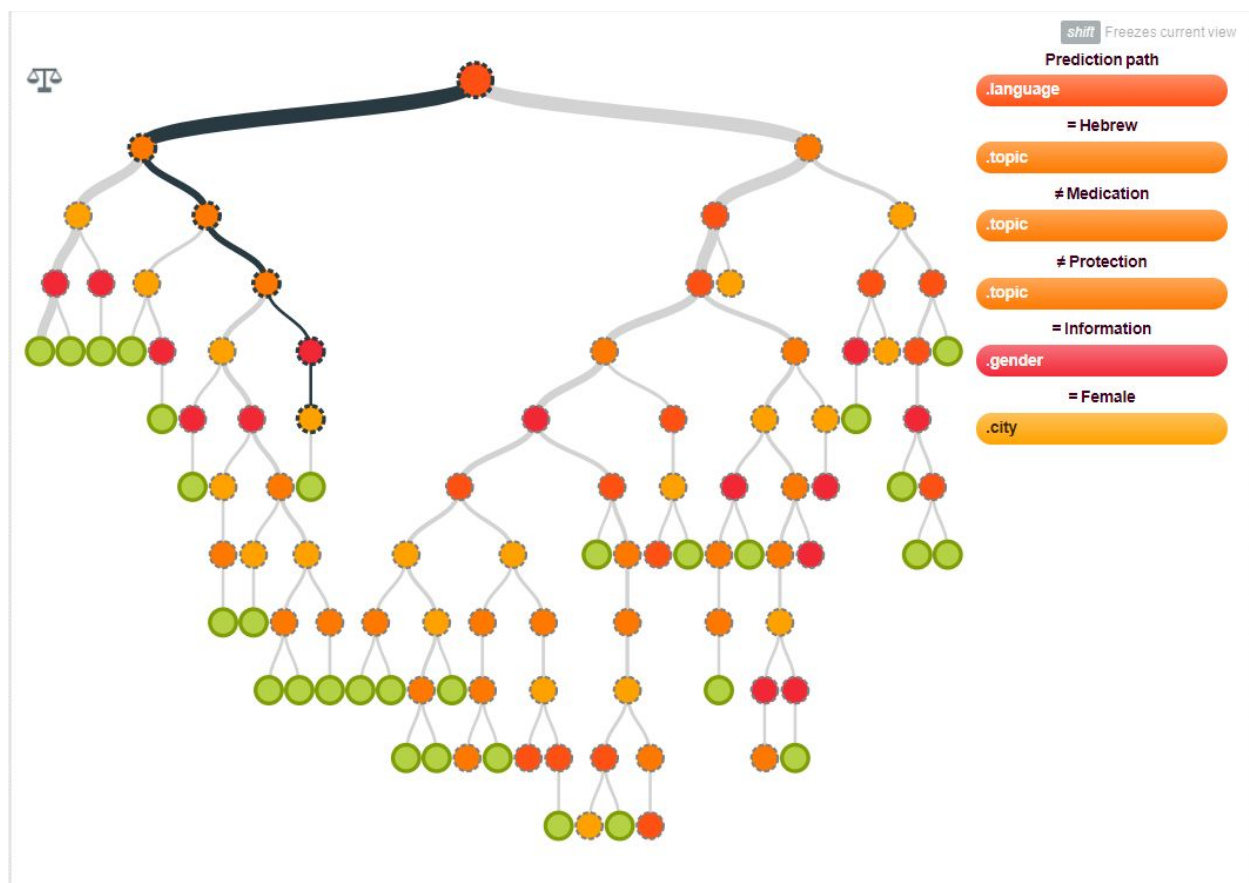
prediction is :39.06968

Start Time	City	Topic	Language	Gender	Age	Situation	End
22-Sep-2020, 1:45	באר שבע	מים	אנגלית	זכר	40	Emergency	get predict

Total Waiting Calls:

1

חיזוי לפי גיל



Data-set

.id	.city	.topic	.language	.gender	.age	.situation	.totalTime
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	0.744
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	55	Emergency	2.651
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	5.432
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	2.841
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	2.511
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	2.407
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	2.269
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	2.113
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	158.675
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	216.535
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	216.53
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	216.517
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	216.519
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	216.533
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	216.512
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	45	Emergency	13.438
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	55	Emergency	14.131
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	28.075
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	32.641
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	25.629
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	1.809
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	1.805
1.60E+12	Jerusalem	Medicatio	Hebrew	Male	18	Emergency	27.062
1.60E+12	Beer Sheva	Water	Russian	Female	65	Emergency	81.808
1.60E+12	Beer Sheva	Water	Amharic	Male	55	Emergency	84.987
1.60E+12	Tel Aviv	Informatic	Hebrew	Female	54	Emergency	86.262
1.60E+12	Jerusalem	Evacuatio	Hebrew	Female	45	Emergency	87.634
1.60E+12	Jerusalem	Protection	English	Male	34	Emergency	89.001

