



סמל מוסד - 270538

פרויקט גמר

הנדסאי מערכות תקשוב

התמחות: פרויקט IOT HTML

הנושא: הקמה, תכנון ופיתוח לחברת smart secure

מנחה הפרויקט: שאלתיאל יעיש

המגיש:

שם: שי פרידמן ת.ז. 213908825

תאריך ההגשה: 18.4.23

תוכן עניינים

3	מבוא
4	על החברה smart secure
5	IDE/S שהיו בשימוש בפרויקט
5	VS CODE
5	Arduino
6	הטכנולוגיות בהן היה שימוש בפרויקט
6	HTML
7	CSS
8	Bootstrap
9	Java Script
10	ESP32
11	Firebase
12	ספריות ו-API
13	העמודים באתר
13	עמוד הבית
15	עמוד הרשמה
16	עמוד כניסה
17	עמוד גלריה
18	עמוד על יוצר האתר
19	עמוד על האתר
20	עמוד יצירת קשר
21	עמוד IOT
22	תרשימי זרימה של הפרויקט
22	תרשים זרימה של האתר לפני התחברות למשתמש
22	תרשים זרימה של האתר לאחר התחברות למשתמש
23	תרשים זרימה בין כל הרכיבים בפרויקט
23	כיצד נראה עץ הfire base בפרויקט
24	החיישנים בפרויקט
25	תרשים זרימה של מערכת חיישן הלחץ
26	תרשים זרימה כיצד המצלמה פועלת
27	חיישן המנוע
28	קוד החיישנים בפרויקט
28	חיישן touch sensor
30	חיישן מצלמה
34	רפלקציה
35	ביבליוגרפיה

מבוא

שמי שי פרידמן ואני סיימתי את לימודי בכיתה יד' במקצוע התקשוב במכללת כנרת. אני גר בנס ציונה ובתיכון הרחבתי את המגמות מדעי המחשב, פיזיקה וסייבר. הלימודים בתיכון היו מאוד מעניינים בשבילי ורציתי להמשיך להרחיב את הידע שלי בתחום ולכן הלכתי ללימודי הנדסאי תקשוב במכללת כנרת. החלטתי ללכת ללמוד את תואר הנדסאי תקשוב לפני הצבא על מנת לקבל תפקיד איכותי ומשמעותי לשירות שלי בצבא ההגנה לישראל. בזכות השקעתי הרבה בלימודים אני התקבלתי להיות פרויקטנט.

במהלך שנת הלימודים הראשונה (שנת יג') למדתי מספר קורסים אשר עזרו לי בהכנת הפרויקט ובניהם: פיתוח WEB, אלקטרוניקה, בסיסי נתונים ועוד. כמובן שבזמן העבודה על הפרויקט הרחבתי בעצמי את החומר הנלמד ולמדתי בבית חומר נוסף. זאת ועוד בשנת הלימודים למדתי מספר קורסים נוספים אשר לא התקשרו לפרויקט הסופי אך גם בהם עשיתי פרויקט למשל לינוקס, שפת פייתון, שפת סי ועוד...

כעת אני בשנת יד' של לימודיי והשנה אני ממשיך ולומד מספר קורסים במכללה כמו שנה שעברה וישנם מספר קורסים נוספים חדשים כגון AWS ו firewall.

כל שנות לימודי מתחילת לימודי התיכון ועד היום (לימודי הנדסאי במכללת כנרת) עזרו לי לעשות את הפרויקט כמו שהוא ולבצע אותו ברמה הגבוהה ביותר. בעוד פרק זמן קצר אני אתגייס ליחידה 108 בחיל האוויר בתור מפתח מערכות תקשורת ובזכות הלימודים במכללה, ההשקעה של המרצים בנו וכמובן כל הזמן בו ישבתי ולמדתי בבית אני יודע שאהיה הגרסה הטובה ביותר של עצמי בתפקיד ואתרום בצורה הטובה ביותר.

על החברה smart secure

החברה הוקמה בשנת 2022 על ידי שי פרידמן. החברה מנוהלת כיום על ידי שי פרידמן ומכילה כ-120 עובדים בכל המחלקות השונות.

לחברה שלושה סניפים המפוזרים בארץ בטבריה, תל אביב ונס ציונה. החברה מייצרת מצלמות אבטחה אשר מזהות פנים ומשווה אותם לבסיס נתונים בזמן אמת. אם הפנים שזוהו מתאימות לפנים שבבסיס הנתונים הדלת תיפתח.

בנוסף יש אופציה להוסיף חיישן לחץ לאיזור סטרילי שנבחר בבית וכאשר החיישן מרגיש לחץ (אדם המשוטט שם) החיישן יתריע למשתמש והמשתמש יבחר מה ברצונו לעשות (להתעלם או לדווח למשטרה).

בחברה ישנן מחלקות רבות בניהן מחלקות פיתוח, מתן שירות, מחקר לרעיונות חדשים ועוד... מטרת הפרויקט היא לתת אבטחה חכמה לבית אשר תעדכן בזמן אמת את המשתמש שקנה את המצלמה וחיישן הלחץ. עם קניית המצלמה או חיישן הלחץ נפתח חשבון באתר למי שקנה ובאתר ניתן לראות את מה שהמצלמה רואה בזמן אמת.

IDE'S שהיו בשימוש בפרויקט

VS CODE

הוא עורך קוד יעיל עם תמיכה בפעולות פיתוח כמו איתור באגים, הפעלת משימות ובקרת גרסאות. מטרתו היא לספק בדיוק את הכלים הדרושים למפתח ובנייה מהירה של הקוד. התקנתו קלה וכל מה שצריך לעשות הוא להיכנס לאתר מייקרוסופט ולהוריד משם את התוכנה. לאחר התקנה יש להוריד לו את ההרחבה הנקראת live server אשר נותנת להריץ את דפי HTML ללא צורך בריענון כל פעם מחדש של הדף.

Arduino

היא פלטפורמת אלקטרוניקה בקוד פתוח המבוססת על חומרה ותוכנה קלות לשימוש. לוחות ארדואינו מסוגלים לקרוא כניסות - אור על חיישן, אצבע על כפתור, הפעלת מנוע, הפעלת LED ופרסום משהו באינטרנט. אתה יכול להגיד ללוח שלך מה לעשות על ידי שליחת סט הוראות למיקרו-בקר שעל הלוח. לשם כך אני משתמש בשפת התכנות Arduino.

שפה זו מכילה מספר פונקציות אשר חייבות להיות בכל תוכנית. פעולות אלה הן setup, loop שתי פעולות אלה חייבות להופיע בכל תוכנית ארדואינו. פונקציית setup היא פונקציה אשר בה מגדירים את הדברים הראשוניים בהתחברות לESP. פונקציית loop היא פונקציה שכל הקוד שלה חוזר בלופים בתוך הESP ושם נשים את מה שאנחנו רוצים שהESP יבצע ללא הפסקה.

התקנת הארדואינו פשוטה אך הגדרתו מעט מסובכת. ראשית מתקינים את הIDE דרך האינטרנט. לאחר התקנה מלאה יש לחבר את הESP בכבל למחשב ולבחור באופציה COM את האחד שאליו הESP מחובר. לאחר מכן יש להוריד ספריה הנקראת ESP32 ספריה זו מביאה לנו את הספריות הנחוצות לכתיבת הקוד לESP. לאחר התקנת הספריה ובחירת הCOM הנכון אנו יכולים להעלות את הקוד לESP ולהריץ עליו את הקוד.

הטכנולוגיות בהן היה שימוש בפרויקט

HTML

HTML היא שפה שאומרת לדפדפן מה אנחנו רוצים להציג למשתמש. השפה בנויה מתגיות (tags), כאשר כל תגית מוצגת בצורה הבאה:

```
<tag opt="Something"...> Content </tag>
```

היא מותרת לשימוש על ידי כל מפתח אתרים, ללא צורך ברכישת זכויות יוצרים מחברה כלשהי, והיא ניתנת לקריאה בכל סוגי המערכות.

HTML גם מאפשר הטמעה של תסריטים בדף, למשל JavaScript, ובכך להשפיע על התנהגות הדף במצבים שונים, שילוב שפת תכנות זו מאפשרת יצירת אינטראקטיביות בין האתר למשתמש בו.

לHTML מספר גרסאות

אך בכל שנותיה של השפה בסיסה לא השתנה ואנו מגדירים כל קובץ HTML כך

```
<HTML>
```

```
<head>
```

```
</head>
```

```
<body>
```

```
</body>
```

```
</HTML>
```

תחילה פותחים תגית HTML ובתוכה נכתוב את כל הקוד לעמוד.

לאחר מכן נשים תגית HEAD ובה נשים רפרנסים ליבוא של קבצים ודברים נוספים. בנוסף בתגית HEAD אנו נותנים כמה הגדרות לעמוד למשל כותרת אשר נראה בדפדפן. נסגור את תגית הHEAD בעזרת / (כך סוגרים כל תגית בשפה זו).

לאחר תגית הHEAD פותחים את תגית הBODY בתוך תגית זו אנו רושמים את כל התוכן שיוצג באתר וכשמסיימים נסגור את התגיות של הHTML והBODY וכך מסיימים את הקובץ

CSS

CSS הם ראשי תיבות של Cascading Style Sheets ובתרגום לעברית: גליונות סגנון מדורגים.

משתמשים ב CSS כדי לתת עיצוב לדפי HTML. אמנם את רוב העיצוב של הדף אפשר לעשות ללא שימוש ב CSS בכלל, אך זה לא מומלץ.

קיימות 3 דרכים עיקריות לשימוש בקוד CSS בעת כתיבת קוד ועיצוב של עמוד. האפשרות הראשונה היא שימוש של קוד באמצעות טעינה של קובץ CSS חיצוני נפרד בתגית הראש כך:

```
<link rel="stylesheet" href="filename.css">
```

האפשרות השנייה היא על ידי טעינה של קוד CSS בתוך אזור ה-hader של האתר עם הקוד העיצובי שאתם רוצים בתוך העמוד

דרך נוספת לטעינת קוד CSS בתוך עמוד היא על ידי הוספת קוד לאלמנט ספציפי בעמוד. למשל לקחת פסקה באתר שמכילה שורת טקסט נוכל להוסיף טקסט ולשנות את הצבע של הטקסט.

```
<p style="color: red">טקסט עם צבע אדום</p>
```

ל CSS יש מספר סלקטורים

1. class

את הקלאס מגדירים כך:

```
.intro {
background-color: yellow;
}
```

שמים נקודה תחילה ולאחר מכן את שם הקלאס (אצלנו intro) ובתוך תגית HTML אנו מזמנים את הקלאס כך class="intro"

2. ID

סלקטור לפי ID מגדירים כך:

```
#firstname {
background-color: yellow;
}
```

לאחר התו # נשים את ה ID שנבחר וכל תגית HTML אשר יש לה את ה ID הספציפי הזה תושפע.

3. element

אלמנט מגדירים כך:

```
p {
background-color: yellow;
}
```

האלמנט היא תגית HTML אשר תושפע מהגדרות אלו (כל התגיות מסוג זה יושפעו).

Bootstrap

Bootstrap היא סביבת עבודה בקוד פתוח לצד לקוח, שמכילה אוסף של כלים ליצירת אפליקציות רשת ואתרים. Bootstrap שוחררה בשנת 2011 על ידי טוויטר, אחרי שהתחילה כספרייה סגורה שנוצרה לשימוש פנימי בטוויטר.

ל-Bootstrap יש אוסף מחלקות CSS, שמתאימות לסיטואציות נפוצות באתרי אינטרנט.

בנוסף לרכיבים הרגילים של HTML ל-Bootstrap יש רכיבים נוספים שכתובים ב-CSS וב-JavaScript שנועדו להקל על המפתח לייצר ממשק משתמש איכותי.

הרכיבים כוללים: כפתורים, תמונות, מד התקדמות והודעות.

Bootstrap נותן את היכולת ליצור בקלות עיצובים רספונסיביים אשר מוכנים מבראשית רק על ידי שיוך לקלאס. בנוסף עם bootstrap כל העיצובים מותאמים גם למחשב וגם לטלפון נייד. העיצובים נמתחים ותכווצים בהתאם לגודל המסך בצורה הטובה ביותר וזאת ללא שימוש בקוד נוסף. דבר זה מקל מאוד על המפתח ומונע עשרות שורות קוד נוספות.

על מנת להתחיל לעבוד עם bootstrap יש לייבא את הספרייה שרוצים. יש מספר רב של גרסאות ולכל אחת עיצוב מעט שונה לרכיבים.

כך מייבאים את הספרייה:

```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
```

ועל מנת לשייך תגית לקלאס רושמים כך:

```
<p class="bigparagraph"> this is big font paragraph</p>
```


Java Script

ג'אווה סקריפט היא שפת תכנות שמשולבת בתוך קוד (בד"כ קוד HTML) ומאפשרת להוסיף תכונות רבות ש HTML לא מאפשר – למשל פתיחת חלון הודעה מתוך דף HTML.

ג'אווה סקריפט היא שפת תכנות, כלומר יש לה תחביר (או חוקים) ואם לא נעמוד בתחביר של השפה נקבל הודעת שגיאה.

השפה היא שפה שתומכת בתכנות מונחה עצמים שעוברת תהליך אינטרפרטציה (פירוש) ולא קומפילציה (הידור). משמעות הדבר שקוד המקור לא צריך לעבור תהליך של קימפול (הידור) לשפת ביניים ואז להריץ אותו, אלא המפרש (interpreter) קורא שורה שורה או מקבץ של שורות ומריץ אותן ישירות. תהליך זה מקל לעיתים להבין היכן התבצעה שגיאה ומאפשר דיבוג קל יותר של השפה.

היא מאפשרת למפתחים לכתוב קוד מעבד בצד הלקוח (client-side) של אפליקציות אינטרנט, ולהביע אינטראקציה עם המשתמשים כגון לחיצות עכבר, מעברי עכבר וקליקים, וכן להצגת תוכן דינמי בדפי האינטרנט.

כיום, JavaScript היא אחת השפות הנפוצות ביותר לפיתוח יישומי אינטרנט והיא נפוצה במיוחד בגלל כך שהיא מאפשרת לפתח יישומים דינמיים ומתאימה לשימוש בצד הלקוח של הדפדפן. ישנם גם מספר מסגרות פיתוח (frameworks) מבוססות על JavaScript, כמו React ו-Angular, המאפשרות לפתח בצורה יעילה יותר.

בנוסף לשימוש בצד הלקוח של הדפדפן, JavaScript נעשה שימוש רחב גם בצד השרת (server-side) עם כלי פיתוח כמו Node.js. השימוש הזה מאפשר למפתחים לכתוב קוד בצד השרת שיכול לשמש את הלקוחות שלהם בצורה יעילה יותר, לדוגמה, לבצע טעינת נתונים אסינכרונית או לשמור מידע בבסיס נתונים.

ESP32

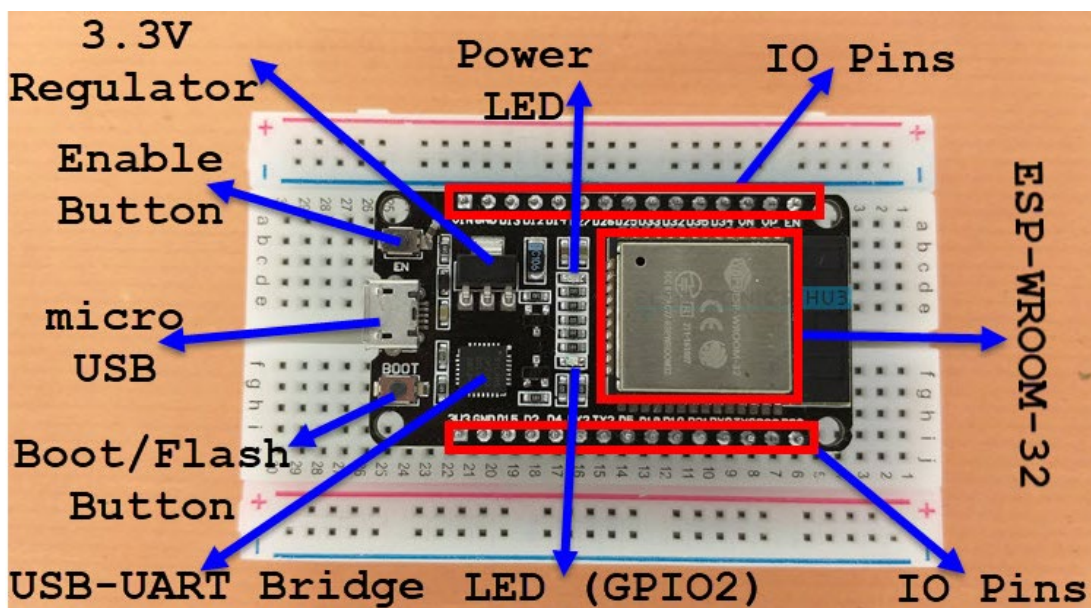
ESP32 הוא מיקרו-בקר מתקדם המבוסס על טכנולוגיות Wi-Fi ובלוטות', המיועד לפיתוח מגוון רחב של מוצרי חכמים ו-IoT. הוא נמצא בשימוש נרחב בפיתוח פתרונות חכמים כמו שלטים אלקטרוניים, מערכות הגברה, מכונות תפירה, מערכות בקרה ואבטחה, מכונות חכמות ועוד.

ESP32 מבוסס על שבב חזק ומתקדם הכולל מעבדים דיגיטליים דו-ליבים המבוססים על Xtensa, מודולי זיכרון ומגוון ציוד היקפי. הוא תומך בפרוטוקולי Wi-Fi 802.11 b/g/n, BLE, Bluetooth, כמו גם חיישנים וממשקים מרובים כגון SPI, I2C, UART ועוד. ESP32 כולל גם מגוון תכונות האצת חומרה כגון הצפנה, CRC וגיבוב, המאפשרים לו לספק ביצועים ויעילות גבוהים.

אחד היתרונות העיקריים של ESP32 הוא צריכת החשמל הנמוכה שלו, מה שהופך אותו לאידיאלי עבור יישומים המופעלים על ידי סוללה. יש לו גם מצב שינה מובנה שמפחית את צריכת החשמל כאשר המכשיר אינו בשימוש. בנוסף, ESP32 תומך במצב שינה עמוק, המאפשר למכשיר לצרוך רק כמה מיקרואמפר של חשמל, מה שמאריך את חיי הסוללה.

מאפיין משמעותי נוסף של ESP32 הוא קלות השימוש שלו ומערכת הפיתוח הנרחבת המקיפה אותו. ה-ESP-IDF (IoT Development Framework) היא סביבת פיתוח מקיפה עבור ESP32 הכוללת תמיכה בשפות וכלים תכנות שונים, כמו גם סט עשיר של ספריות ודוגמאות. זה מקל על מפתחים ליצור יישומים ומוצרים המבוססים על ESP32.

לסיכום, ESP32 הוא מיקרו-בקר רב עוצמה ורב-תכליתי המספק סט עשיר של תכונות וממשקים, מה שהופך אותו לאידיאלי עבור מגוון רחב של יישומי חכמים ו-IoT. עם צריכת החשמל הנמוכה, האצת החומרה ומערכת הפיתוח העשירה שלה, היא בחירה פופולרית עבור מפתחים המעוניינים ליצור מוצרים חכמים בעלי ביצועים גבוהים ויעילים. הסביבה המוכרת ביותר והנוחה ביותר לרוב היא ארדואינו ולכן אשתמש בה לפיתוח הפרויקט שלי.



Firestore

פירבייס הוקמה בשנת 2011 כחברה עצמאית בשם Envolv על ידי גיימס טמפלן.

תחילה, סיפקה החברה ממשק תכנות יישומים באמצעותו ניתן לשלב שירות צ'אט מקוון באתרי אינטרנט ובהמשך גילו שזה לא השימוש היחידי שנעשה בממשק ולכן הפרידו בין מערכת הצ'אט לבין הארכיטקטורה בזמן אמת שהפעילה אותה.

Firestore הוא מסד נתונים מסוג NoSQL.

היתרון של מסד נתונים מסוג NoSQL הוא ב-Scaling. הוא יכול יחסית בקלות לפצל את העבודה על כמה מכונות. המשמעות היא שמסד נתונים כזה יכול לטפל בכמות רבה יותר של נתונים מאשר מסד נתונים רציונלי.

Firestore הוא real time data base, כלומר הוא מתעדכן בזמן אמת ומציג את הנתונים מיד כאשר הוא מתעדכן בזמן אמת.

Firestore שימושים שונים כגון:

שירותי אימות ואירוח

Firestore Authentication היא כלי אימות עוצמתי מבית גוגל המותאם לאפליקציות הפועלות על בסיס Firestore.

ניתן להשתמש בכלים מוגדרים מראש או לבנות ממשק משתמש ייעודי בהתאמה אישית שיאפשר לאמת משתמשים בקלות ולהנפיק אישורים בהתאמה אישית דרך מייל או ברשתות החברתיות.

בנוסף, Firestore מציעה אירוח אתרים סטטי ליישומים הבנויים על בסיס HTML, CSS ו-JavaScript. על השרתים המארחים נוכל ליישם פרוטוקולי HTTPS ו-SSL סטנדרטיים כדי לשמור על האבטחה.

יישומים ללא שרת

אינטגרציה נוספת של גוגל הזמינה ב-Firebase מגולמת בשלל יישומי ענן שמאפשרים להפעיל קוד Backend על בסיס אירוע. הפעלת היישומים באפליקציה נעשית על בסיס ארכיטקטורה ללא שרת.

כלומר, ניתן לבנות יישומים כמערכת של פקודות ופונקציות נפרדות המבודדות זו מזו בענן ולחבר אותן באמצעות ממשקי API. הפופולריות של ארכיטקטורות אלו הולכת ועולה עקב הגמישות הרבה שהן מאפשרות והן מיושמות בפתרון התוכנה הזה באופן נוח וידידותי.

מסד נתונים יציב ונוח לשימוש

מסד הנתונים של Firestore המאפשר ניהול בזמן אמת הוא הכלי הראשון שפותח במסגרת הפתרון הזה, מה שהופך אותו גם לשירות היציב והבדוק ביותר.

ספריות API

face-api.js : ספרייה זו בנויה על גבי TensorFlow.js ומספקת מודלים מאומנים מראש לזיהוי פנים, זיהוי ציוני דרך וזיהוי פנים. זה מאפשר למפתחים להוסיף בקלות יכולות זיהוי וזיהוי פנים ליישומי אינטרנט.

tensorflow : ספרייה זו היא יישום JavaScript של TensorFlow, מסגרת למידת מכונה בקוד פתוח שפותחה על ידי Google. זה מאפשר למפתחים לבנות ולאמן מודלים של למידת מכונה ב-JavaScript, ולפרוס מודלים אלה ביישומי אינטרנט.

face-api.min.js : זוהי גרסה ממוזערת של ספריית face-api.js. הוא משמש כאשר מפתחים רוצים לייעל את מהירות הטעינה של יישום האינטרנט שלהם ולהקטין את גודל קבצי ה-JavaScript שמשתמשים צריכים להוריד.

על ידי הכללת ספריות אלו באפליקציית האינטרנט, ניתן להשתמש במודלים מאומנים מראש כדי לזהות ולזהות פרצופים בתמונות או בזרמי וידאו, ואפילו לבצע משימות כגון זיהוי רגשות או זיהוי תכונות פנים.

על מנת להוסיף את הספריות האלה לאתר יש להעתיק ולהדביק את שלוש תגיות הסקריפט בתוך תגית הhead. לאחר מכן אנו נוכל להשתמש ביכולות של הספריות האלה בצד הלקוח.

```
<script src="https://cdn.jsdelivr.net/npm/face-api.js@0.22.2/dist/face-api.js">  
</script>
```

```
<script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@3.12.0/dist/tf.js">  
</script>
```

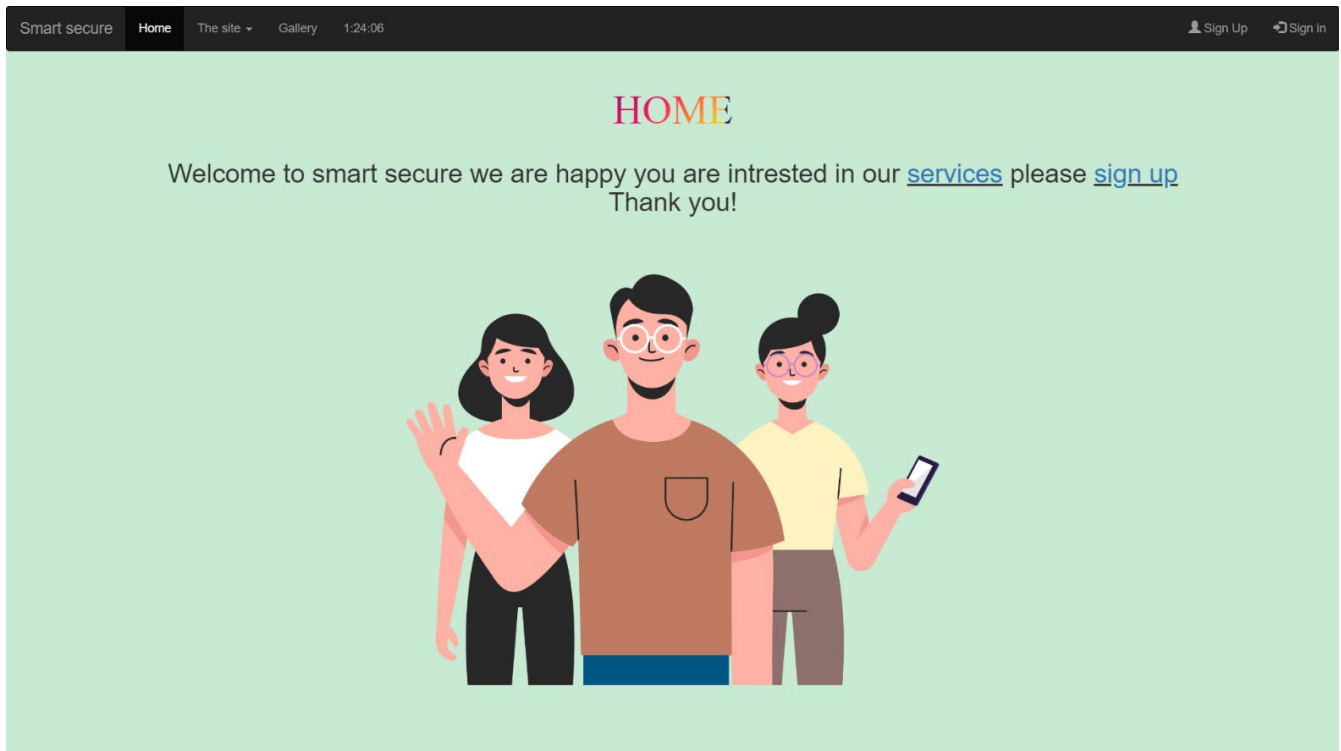
```
<script src="https://cdn.jsdelivr.net/npm/face-api.js@0.22.2/dist/face-api.min.js">  
</script>
```

העמודים באתר

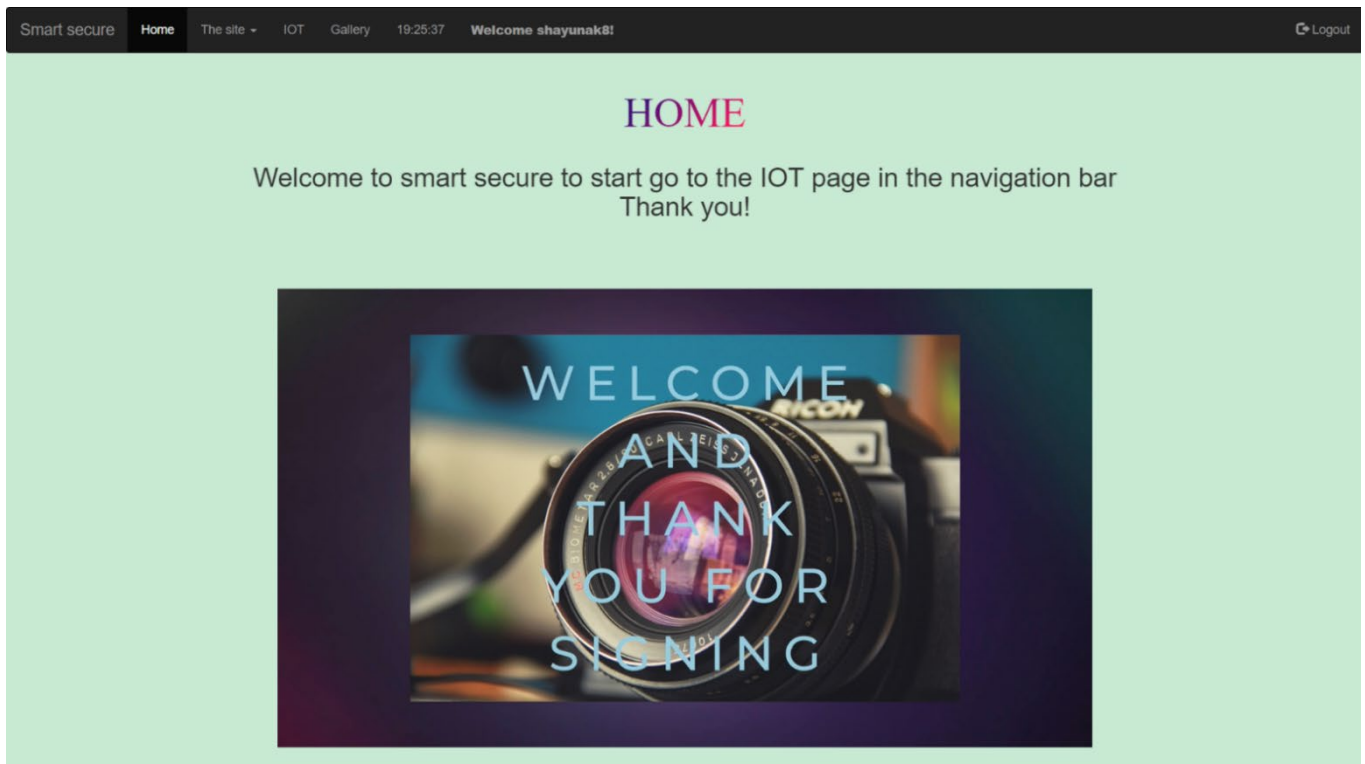
בכל עמוד שקיים אצלי באתר אציג תמונה של איך הוא נראה ואסביר למה הוא משמש. ישנם עמודים אשר משתנים לאחר כניסה למשתמש ולכן אשים בעמודים כאלה שתי תמונות.

עמוד הבית

עמוד הבית לפני כניסה למשתמש



עמוד הבית לאחר כניסה למשתמש



עמוד זה הוא עמוד הקבלה, הדף הראשון אותו רואים כאשר נכנסים לאתר. בתמונה הראשונה ניתן לראות כיצד העמוד נראה לפני כניסה למשתמש באתר וניתן לראות כי ישנן אופציות שונות בסרגל הניווט לפני ואחרי התחברות למערכת. כמו כן תוכן הדף משתנה בעת התחברות.

עמוד הרשמה

Smart secure Home The site Gallery 19:31:53 Sign Up Sign

SIGN UP

Please fill in this form to create an account.

Enter your username

Enter your email

Enter your password

Enter your password

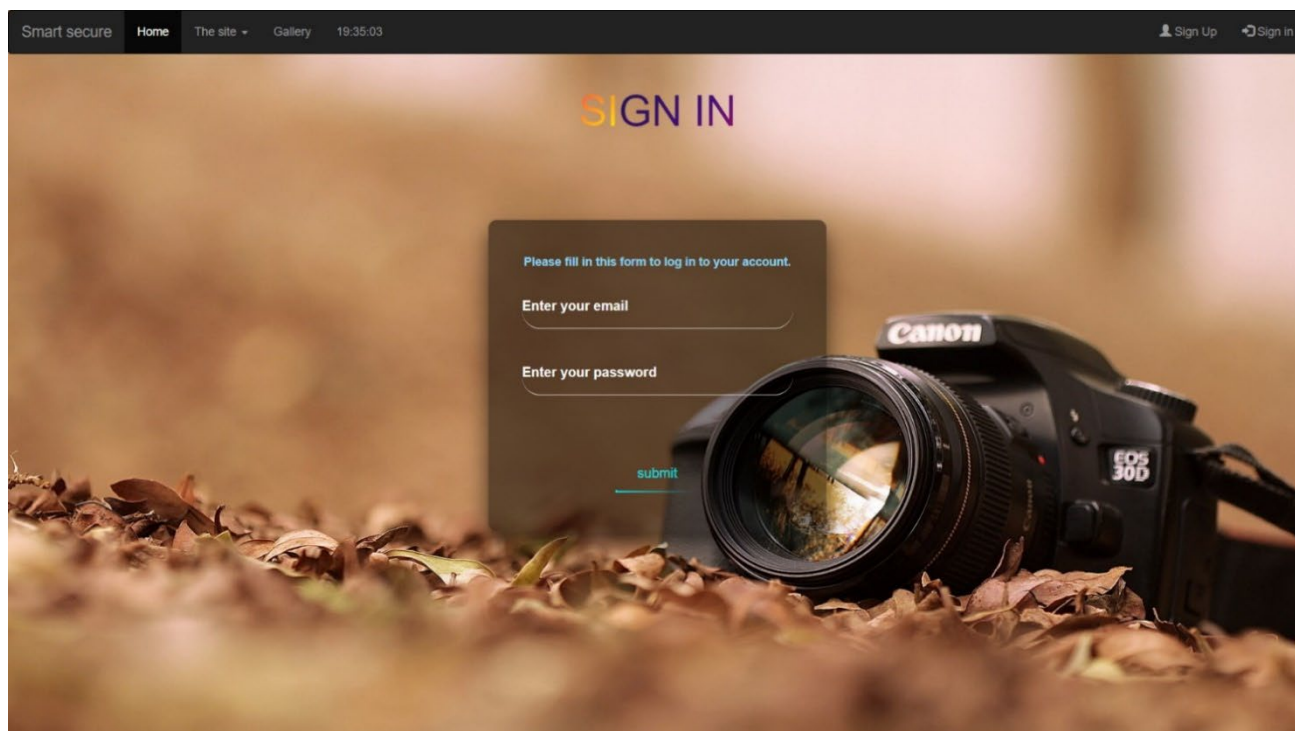
By creating an account you agree to our

[TERMS & PRIVACY](#)

submit

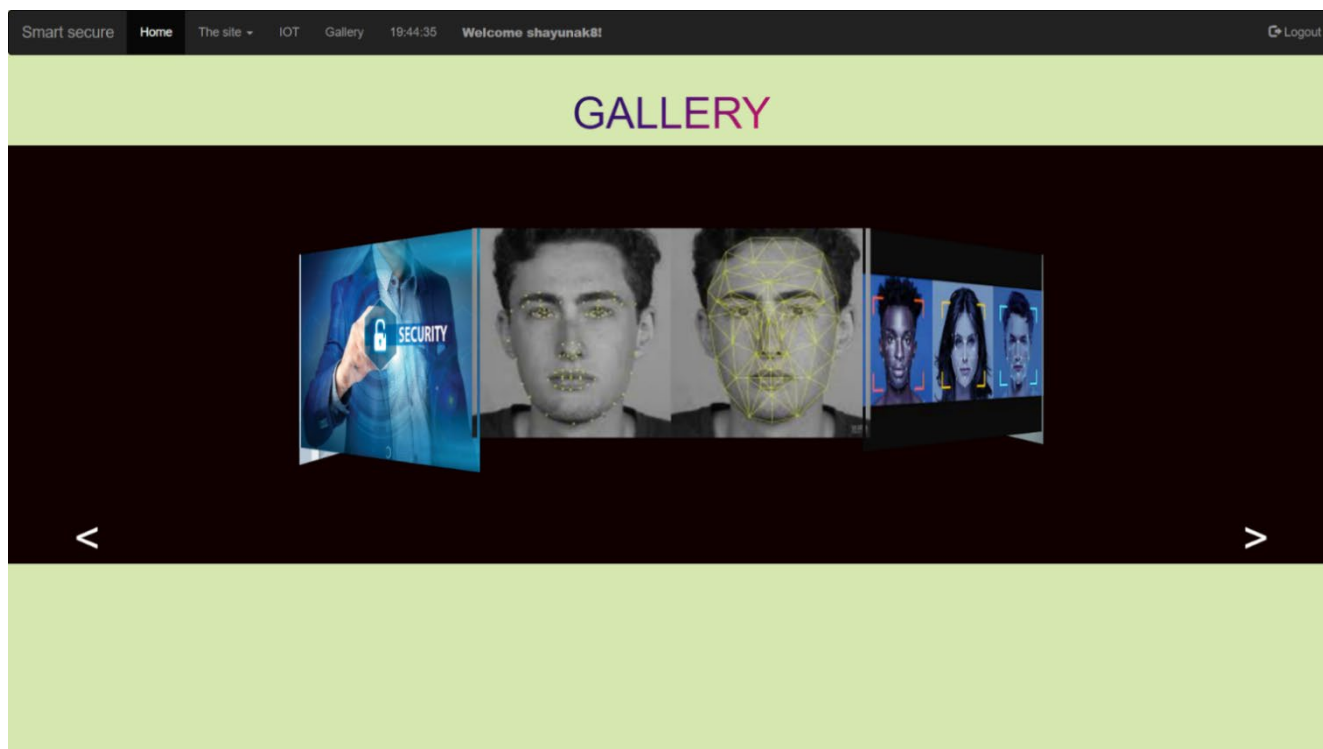
עמוד זה הוא עמוד ההרשמה לאתר, בעמוד זה מכניסים את שם המשתמש, אימייל וסיסמא וכך נרשמים לאתר. לאחר ההרשמה לאתר אנו מועברים לעמוד הבית ומתחברים אוטומטית לחשבון שיצרנו. בקוד JS מאחורי עמוד זה ישנן מספר בדיקות למשל: בדיקה ששם המשתמש ארוך מ-6 תווים, בדיקה שהסיסמא ואימות הסיסמא זהים.

עמוד כניסה



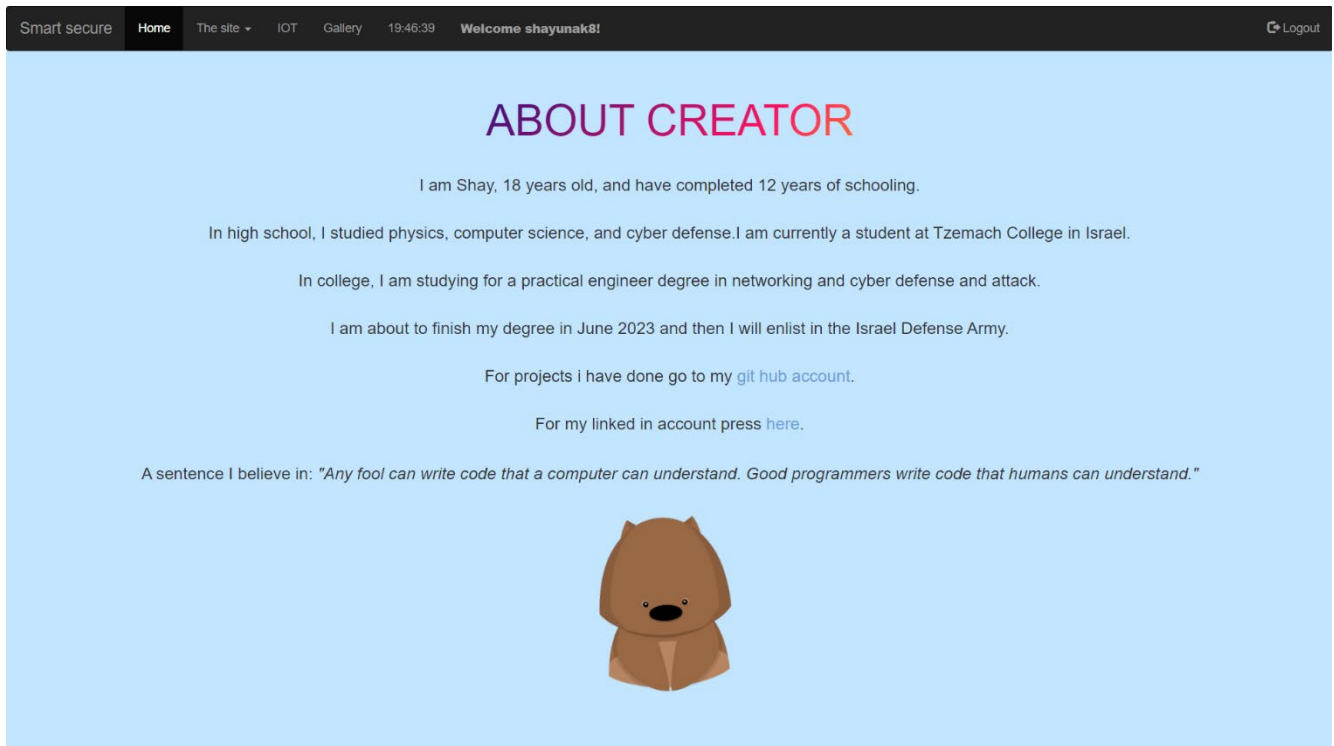
עמוד זה הוא עמוד הכניסה למשתמשים רשומים. משתמשים שכבר נרשמו לאתר ורוצים להיכנס לחשבונם עושים זאת דרך עמוד זה. לאחר הכנסת הפרטים במידה ומתאימים למשתמש קיים המשתמש מועבר לעמוד הבית ומתחבר לחשבון שלו. במידה והוכנסו פרטים לא נכונים תוצג הודעה מתאימה במסך.

עמוד גלריה



עמוד זה הוא עמוד הגלריה המציג תמונות בתוך גלגל המסתובב במרחב לפי בחירת המשתמש.
עמוד זה מוצג למשתמשים רשומים וגם כן לא רשומים בסרגל הניווט.

עמוד על יוצר האתר



בעמוד זה יש הסבר על יוצר האתר (שי פרידמן) ולינקים לחשבון ה GitHub וה LinkedIn. עמוד זה מוצג למשתמשים רשומים וגם כן לא רשומים בסרגל הניווט.

עמוד על האתר



בעמוד זה ישנם הסברים על האתר ועל מה יש לו להציע למשתמשים שקנו את השירות שלו. דף זה מוצג למשתמשים רשומים וגם כן כאלו שלא נכנסו למשתמש שלהם.

עמוד יצירת קשר

עמוד זה הוא עמוד יצירת קשר, ניתן ללחוץ על הקישורים השונים בצד ימין אשר ישלחו אותך למקום האמיתי למשל קישור האימייל ישלח אותך לאתר גימייל כיך הודעה חדשה עם הנמען מוכן. קישור הטלפון יחייג אל המספר האמיתי שלי במידה ומתחברים מהטלפון לאתר.

IOT עמוד

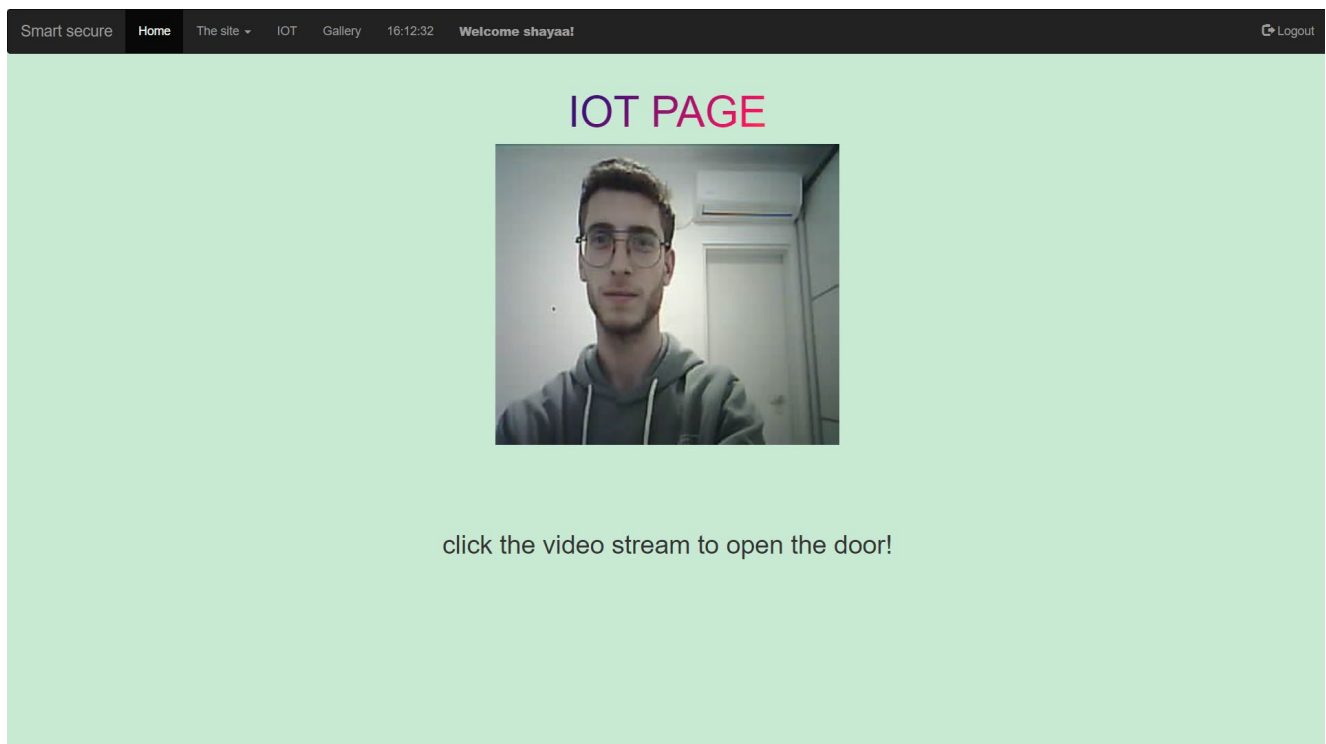
עמוד זה מופיע בסרגל הניווט רק לאחר כניסה למשתמש באתר וכך זה נראה לפני התחברות:



אחרי התחברות:



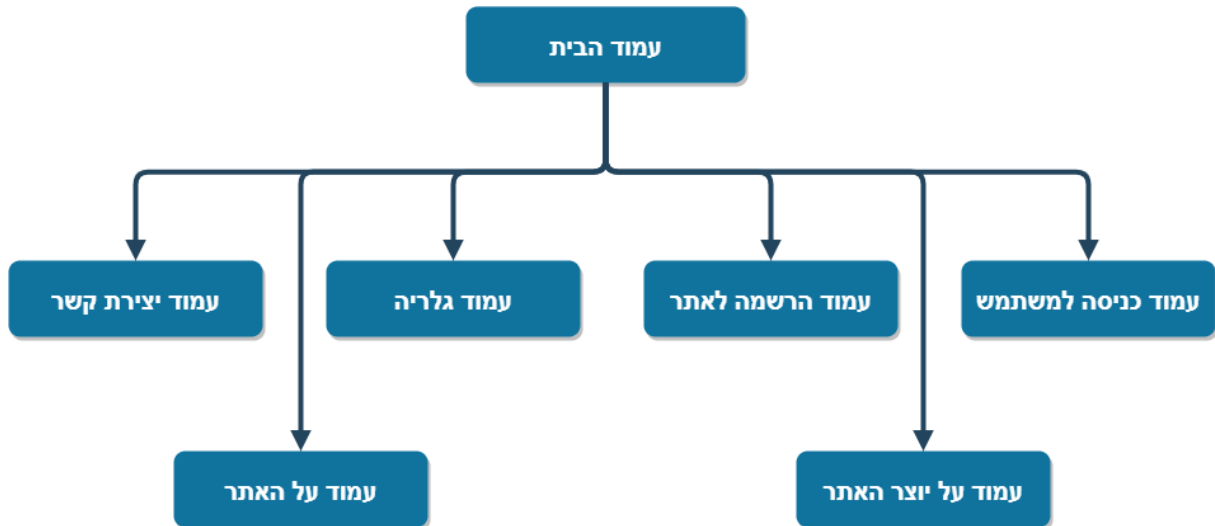
כך נראה העמוד



בעמוד זה אנו יכולים לראות מה המצלמה רואה ברגעים אלו וכמו כן בעמוד אנו מקבלים הודעה במידה ונמצא אדם בתוך השטח הסטרילי שהגדרנו.

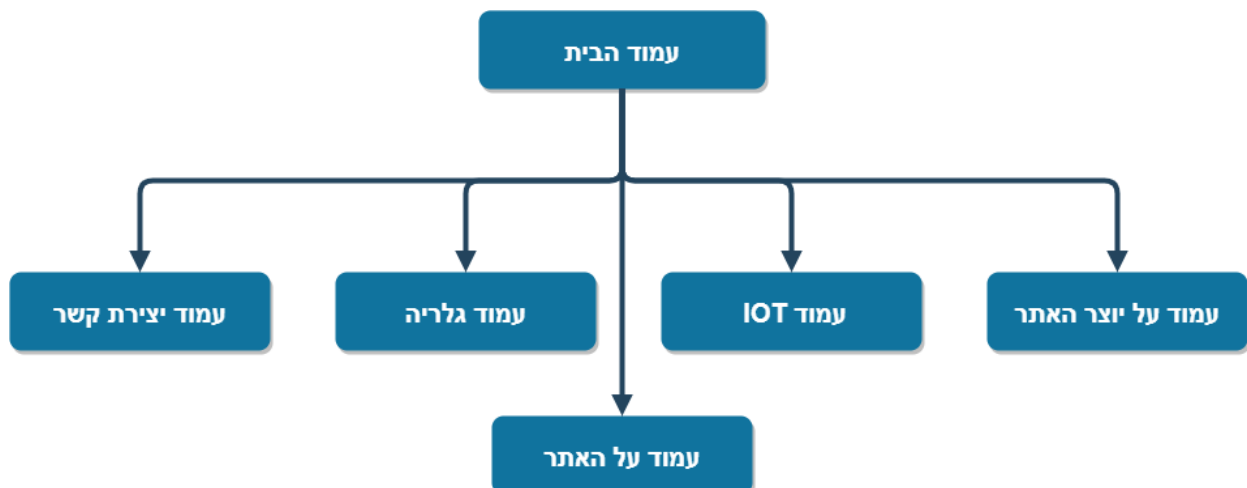
תרשימי זרימה של הפרויקט

תרשים זרימה של האתר לפני התחברות למשתמש



לפני ההתחברות לא ניתן להיכנס לכל העמודים באתר אך ישנם עמודים שמופיעים רק לפני ההתחברות למשל עמוד כניסה למשתמש ועמוד ההרשמה. בין כל העמודים האלו ניתן לעבור בחופשיות.

תרשים זרימה של האתר לאחר התחברות למשתמש



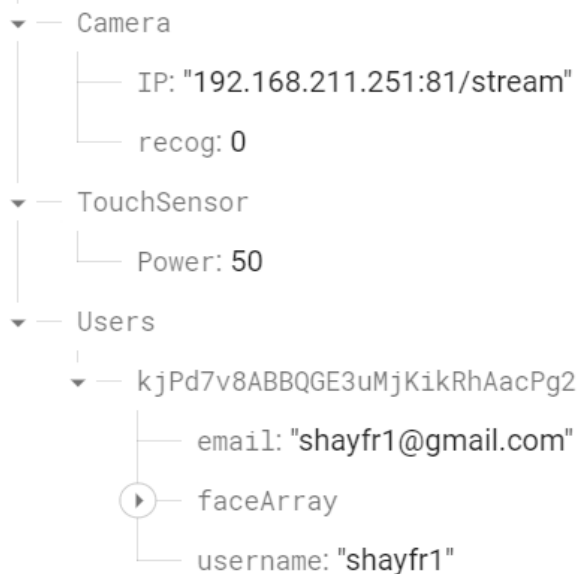
לאחר ההתחברות נוסף לנו את עמוד ה-IOT אשר ממנו מקבלים הודעות מהחיישנים. בנוסף לא יהיה ניתן לעבור לעמודי ההתחברות וההרשמה לאתר. כמו כן מכל דף ניתן לעבור לכל דף אחר.

תרשים זרימה בין כל הרכיבים בפרויקט



כיצד נראה עץ ה-fire base בפרויקט

<https://final-project-b3ef7-default-rtdb.europe-west1.firebaseio.com/>



לכל חיישן ישנו ענף, לחיישן המצלמה יש שני תתי ענפים אחד מכיל את כתובת ה-IP של המצלמה והשני מסמן למנוע מתי לפתוח את הדלת. לחיישן השטח הסטרילי יש תת ענף המסמן מה חוזק הנגיעה בו. וענף שלישי הוא של המשתמשים הקיימים באתר. לכל משתמש שלושה תתי ענפים, אחד לאימייל השני למערך הפנים אותו סרק בהרשמה והשלישי הוא שם המשתמש.

החיישנים בפרויקט

בפרויקט אני משתמש בחיישן מצלמה מובנה על הESP. מטרת המצלמה היא לצלם בכל רגע ולמצוא פנים. כאשר נמצאו פנים, הESP ישווה את הפנים שנקלטו במצלמה לפנים שהוכנסו לדטה בייס מבעוד מועד. אם הESP רואה כי מדובר באותו בן אדם דלת הכניסה לבית שעליה מותקנת המצלמה תפתח. ולעומת זאת אם הESP לא יזהה את הפנים שנקלטו הוא ישלח הודעה למשתמש ותודיע לו כי נמצא אדם זר מחוץ לבית.

זאת ועוד יש שימוש בחיישן נגיעה אשר בודק דריכה באיזור סטרילי אשר לא אמור לדרוך שם אדם. אם החיישן מרגיש כי יש אדם על המשטח המשתמש מקבל התרעה שמודיע לו שישנו אדם בשטח הסטרילי.

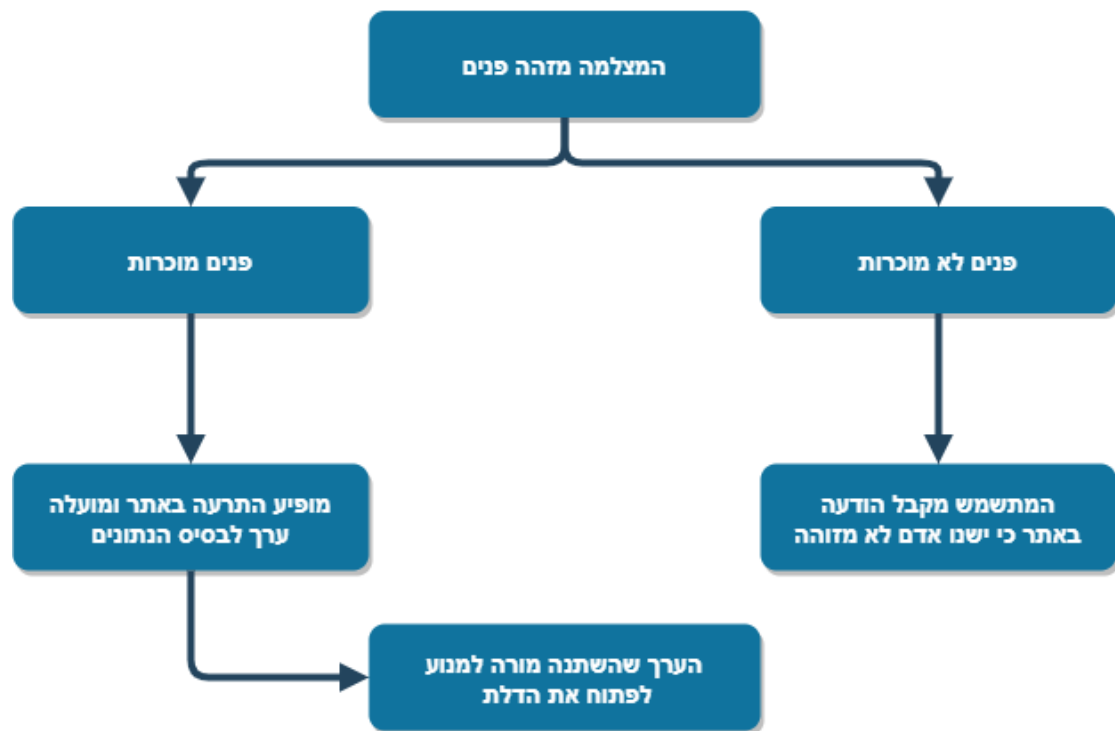
בעזרת שימוש בכל הטכנולוגיות ביחד (בסיס נתונים בזמן אמת, חיישן המתריע על ממצא אדם חשוד ו ESP אשר ישמש כחיישן מצלמה ובדיקת זהות חשוד) נוכל לתת את האבטחה הטובה ביותר לבית בזמן אמת ובכך להבטיח את בטיחות האנשים הגרים בבית. בנוסף תמיד ניתן יהיה לראות את מה שהמצלמה רואה והיא בעצם תשמש גם כמצלמת אבטחה.

תרשים זרימה של מערכת חיישן הלחץ



לחיישן זה אין מעגל חשמלי, אלא הכל נעשה ברגל D4 ב-ESP. כאשר מורגשת נגיעה ברגל זו החוזה של הנגיעה מועלה לבסיס הנתונים. הערך נע בין 1-106 כאשר 1 זו נגיעה עוצמתית ו-106 משמעו אין אדם באיזור.

תרשים זרימה כיצד המצלמה פועלת



למצלמה אין מעגל חשמלי אלא היא מולחמת על ה-ESP ומחוברת עליו ישירות. המצלמה פועלת ללא הפסקה כאשר מחוברת לספק מתח ומעלה את השידור לכתובת ה-IP שאותה העלתה גם כן לבסיס הנתונים.

חיישן המנוע

בפרויקט השתמשתי במנוע שפותח וסוגר את הדלת לאחר שמזוהות פנים מוכרות. המנוע פועל כאשר משתנה ערך בבסיס הנתונים, כאשר המצלמה מזהה פנים מוכרות היא מעלה לבסיס הנתונים את הספרה 1. כאשר ה-ESP מבחין שהערך שונה מ-0 ל-1 הוא פותח וסוגר את דלת הכניסה ומחזיר את הערך בבסיס הנתונים ל-0. וכך בכל פעם שמשתנה הערך בבסיס הנתונים (כאשר מזוהות פנים מוכרות) המנוע פותח וסוגר את הדלת.

קוד החיישנים בפרויקט

חיישן touch sensor

```
#define VERSION "1.1"
#include <WiFi.h>
#include <inttypes.h>
#include <Wire.h>
#include <Firebase_ESP_Client.h>
#include "addons/TokenHelper.h"
//Provide the RTDB payload printing info and other helper functions.
#include "addons/RTDBHelper.h"

#define WIFI_SSID "POCO F3"
#define WIFI_PASSWORD ""
// Insert Firebase project API Key
#define API_KEY "AIzaSyCKccNAw3hgB3vGN7x47RRFgd2iY5iLNLU"

// Insert RTDB URLdefine the RTDB URL */
#define DATABASE_URL "https://final-project-b3ef7-default-
rtdb.europe-west1.firebaseio.com/"

//Define Firebase Data object
FirebaseData fbdo;

FirebaseAuth auth;
FirebaseConfig config;

unsigned long sendDataPrevMillis = 0;
int count = 0;
bool signupOK = false;

// ESP32 Touch Test
// Just test touch pin - Touch0 is T0 which is on GPIO 4.

void setup() {
    Serial.begin(115200);
    Serial.println("Start");
    delay(1000); // give me time to bring up serial monitor
    Serial.println("ESP32 Touch Test");
    WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
    Serial.print("Connecting to Wi-Fi");
    while (WiFi.status() != WL_CONNECTED){
        Serial.print(".");
        delay(1000);
    }
    Serial.println();
    Serial.print("Connected with IP: ");
    Serial.println(WiFi.localIP());
    Serial.println();

    config.api_key = API_KEY;

    /* Assign the RTDB URL (required) */
    config.database_url = DATABASE_URL;

    /* Sign up */
    if (Firebase.signUp(&config, &auth, "", "")){
        Serial.println("ok");
    }
}
```

```

    signupOK = true;
}
else{
    Serial.printf("%s\n", config.signer.signupError.message.c_str());
}

/* Assign the callback function for the long running token generation
task */
config.token_status_callback = tokenStatusCallback; //see
addons/TokenHelper.h

Firebase.begin(&config, &auth);
Firebase.reconnectWiFi(true);
}

void loop() {
    int value = touchRead(4);
    Serial.println(touchRead(4)); // get value of Touch 0 pin = GPIO 4
    if (Firebase.ready() && signupOK && (millis() - sendDataPrevMillis
> 3000 || sendDataPrevMillis == 0)){
        sendDataPrevMillis = millis();
        if (Firebase.RTDB.setFloat(&fbdo, "TouchSensor/Power", value)){
            Serial.println("PASSED");
            Serial.println("PATH: " + fbdo.dataPath());
            Serial.println("TYPE: " + fbdo.dataType());
        }
        else {
            Serial.println("FAILED");
            Serial.println("REASON: " + fbdo.errorReason());
        }
    }
    delay(1000);
}

```

בקוד זה יש בדיקה שחוזרת שוב ושוב ובודקת האם יש נגיעה ברגל gpio4 של הESP. רגל זו מרגישה אם נוגעים בה ובאיזה עוצמה נוגעים בה (נגיעה רכה עם האצבע או לחיצה חזקה על הרגל). כאשר הרגל חשה בנגיעה אפילו הכי קטנה היא מעלה לFB את ערך העוצמה נע בין 1-120. באתר עשיתי שכאשר הESP מרגיש נגיעה החזקה מ70 תוצג הודעה מתאימה למשתמש שאומרת כי יש אדם בתוך השטח הסטרילי.

```
#include "esp_camera.h"
#include <WiFi.h>
#include <Arduino.h>
#include <Firebase_ESP_Client.h>

//Provide the token generation process info.
#include "addons/TokenHelper.h"
//Provide the RTDB payload printing info and other helper functions.
#include "addons/RTDBHelper.h"

#define CAMERA_MODEL_AI_THINKER // Has PSRAM

#include "camera_pins.h"

// =====
// Enter your WiFi credentials
// =====
const char* ssid = "POCO F3";
const char* password = "";

void startCameraServer();

#define API_KEY "AIzaSyCKccNAw3hgB3vGN7x47RRFgd2iY5iLNLU"

// Insert RTDB URLdefine the RTDB URL */
#define DATABASE_URL "https://final-project-b3ef7-default-
rtdb.europe-west1.firebaseio.com/"

//Define Firebase Data object
FirebaseData fbdo;

FirebaseAuth auth;
FirebaseConfig fbconfig;

unsigned long sendDataPrevMillis = 0;
int count = 0;
bool signupOK = false;

void setup() {
    Serial.begin(115200);
    Serial.setDebugOutput(true);
    Serial.println();

    camera_config_t config;
    config.ledc_channel = LEDC_CHANNEL_0;
    config.ledc_timer = LEDC_TIMER_0;
    config.pin_d0 = Y2_GPIO_NUM;
    config.pin_d1 = Y3_GPIO_NUM;
    config.pin_d2 = Y4_GPIO_NUM;
    config.pin_d3 = Y5_GPIO_NUM;
    config.pin_d4 = Y6_GPIO_NUM;
    config.pin_d5 = Y7_GPIO_NUM;
    config.pin_d6 = Y8_GPIO_NUM;
    config.pin_d7 = Y9_GPIO_NUM;
    config.pin_xclk = XCLK_GPIO_NUM;
    config.pin_pclk = PCLK_GPIO_NUM;
```

```

config.pin_vsync = VSYNC_GPIO_NUM;
config.pin_href = HREF_GPIO_NUM;
config.pin_sscb_sda = SIOD_GPIO_NUM;
config.pin_sscb_scl = SIOC_GPIO_NUM;
config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.frame_size = FRAMESIZE_UXGA;
config.pixel_format = PIXFORMAT_JPEG; // for streaming
//config.pixel_format = PIXFORMAT_RGB565; // for face
detection/recognition
config.grab_mode = CAMERA_GRAB_WHEN_EMPTY;
config.fb_location = CAMERA_FB_IN_PSRAM;
config.jpeg_quality = 12;
config.fb_count = 1;

// if PSRAM IC present, init with UXGA resolution and higher JPEG
quality
//                               for larger pre-allocated frame buffer.
if(config.pixel_format == PIXFORMAT_JPEG){
    if(psramFound()){
        config.jpeg_quality = 10;
        config.fb_count = 2;
        config.grab_mode = CAMERA_GRAB_LATEST;
    } else {
        // Limit the frame size when PSRAM is not available
        config.frame_size = FRAMESIZE_SVGA;
        config.fb_location = CAMERA_FB_IN_DRAM;
    }
} else {
    // Best option for face detection/recognition
    config.frame_size = FRAMESIZE_240X240;
}
#if CONFIG_IDF_TARGET_ESP32S3
    config.fb_count = 2;
#endif

#if defined(CAMERA_MODEL_ESP_EYE)
    pinMode(13, INPUT_PULLUP);
    pinMode(14, INPUT_PULLUP);
#endif

// camera init
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
    Serial.printf("Camera init failed with error 0x%x", err);
    return;
}

sensor_t * s = esp_camera_sensor_get();
// initial sensors are flipped vertically and colors are a bit
saturated
if (s->id.PID == OV3660_PID) {
    s->set_vflip(s, 1); // flip it back
    s->set_brightness(s, 1); // up the brightness just a bit
    s->set_saturation(s, -2); // lower the saturation
}
// drop down frame size for higher initial frame rate
if(config.pixel_format == PIXFORMAT_JPEG){
    s->set_framesize(s, FRAMESIZE_QVGA);
}

```

```

#ifdef CAMERA_MODEL_M5STACK_WIDE ||
defined(CAMERA_MODEL_M5STACK_ESP32CAM)
    s->set_vflip(s, 1);
    s->set_hmirror(s, 1);
#endif

#ifdef CAMERA_MODEL_ESP32S3_EYE
    s->set_vflip(s, 1);
#endif

    WiFi.begin(ssid, password);
    WiFi.setSleep(false);

    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");

    startCameraServer();

    Serial.print("Camera Ready! Use 'http://");
    Serial.print(WiFi.localIP());
    Serial.println("' to connect");

    fbconfig.api_key = API_KEY;

    /* Assign the RTDB URL (required) */
    fbconfig.database_url = DATABASE_URL;

    /* Sign up */
    if (Firebase.signUp(&fbconfig, &auth, "", "")){
        Serial.println("ok");
        signupOK = true;
    }
    else{
        Serial.printf("%s\n",
fbconfig.signer.signupError.message.c_str());
    }

    /* Assign the callback function for the long running token
generation task */
    fbconfig.token_status_callback = tokenStatusCallback; //see
addons/TokenHelper.h

    Firebase.begin(&fbconfig, &auth);
    Firebase.reconnectWiFi(true);
}

void loop() {
    if (Firebase.ready() && signupOK && (millis() -
sendDataPrevMillis > 15000 || sendDataPrevMillis == 0)){
        sendDataPrevMillis = millis();
        // Write an Int number on the database path test/int
        if
(Firebase.RTDB.setString(&fbdo, "Camera/IP", WiFi.localIP().toString() +
":81/stream")){
            Serial.println("PASSED");
            Serial.println("PATH: " + fbdo.dataPath());

```



```
        Serial.println("TYPE: " + fbdo.dataType());
    }
    else {
        Serial.println("FAILED");
        Serial.println("REASON: " + fbdo.errorReason());
    }
}
}
```

בקוד זה אנו עלים את כתובת הIP המתקבלת מהרשת לFB ולאחר מכן עושים שידור לכתובת הIP ומציג את מה שרואים במצלמה.

רפלקציה

במהלך השנתיים בהן למדתי במכללת כנרת עבדתי על הפרויקט רבות, השקעתי בו שעות רבות מזמני על מנת להגיע אתו לרמה הגבוהה ביותר.

בתהליך העבודה למדתי על עצמי, וכמובן הרחבתי את הידע שלי בכל תחומי המחשבים למשל רשתות ותכנות. את העבודה על הפרויקט עשיתי לבד ולמדתי בעזרת סרטונים ופורומים שונים את החומר הנדרש להכנתו, חלק מן הידע למדתי במהלך לימודי במכללה וחלקו כפי שאמרתי למדתי לבד. למרות שחלק מן הידע הנדרש היה לי כבר אני נתקלתי בקשיים רבים, חלקם גדולים וחלקם פחות ולדעתי הצלחתי להתמודד עם כולם בצורה הטובה ביותר.

האתגר הגדול ביותר מבחינתי היה ללמוד את החומר לבד מפני שהצורה הנוחה ביותר בשבילי ללמידת חומר חדש היא הרצאה פרונטלית כמו בכיתה ולכן היה לי קושי בלמידה העצמית בבית.

מהפרויקט קיבלתי המון כלים לחיים, למדתי להסתדר עם למידה עצמאית שכפי שכתבתי לא פשוטה בשבילי. בנוסף למדתי כיצד לממש את החומר שלמדתי בשנות לימודי במכללה ואני בטוח שזה יעזור לי גם בהמשך לימודי ובחיים עצמם.

בנוסף, מהפרויקט הסקתי מסקנות רבות, אחת מן המסקנות שהסקתי היא שאני מסוגל כאשר אני רוצה. אני משקיע המון שעות בשבוע בלימודים וזה מאוד לא פשוט לי אך בזמן העבודה על הפרויקט הבנתי שכאשר אני רוצה משהו (במקרה שלי רציתי להגיע לפרויקט ברמה גבוהה מן הנדרש) אני מסוגל לעמוד כמה שצריך על מנת להגיע לתוצאות הרצויות.

אילו הייתי צריך להתחיל את הפרויקט בימים אלו הייתי עובד באותה צורה מפני שאני חושב שמהדרך בה עבדתי למדתי על עצמי רבות והתפתחתי מאוד בזכותה. זאת ועוד בדרך שבה עבדתי אני למדתי כיצד לחפש "חכם יותר" בגוגל מפני שבימינו כאשר כל פיסת מידע קטנה נמצאת באינטרנט אפשר להיאבד בו במהירות.

לסיכום אני חושב שהפרויקט פיתח אותי הן מבחינת ידע והן מבחינה אישית ואני שמח מאוד שעברתי את זה. זאת הייתה חוויה בשבילי ואני בטוח שאזכור את הפרויקט והעבודה עליו גם בהמשך חיי.

ביבליוגרפיה

<https://www.w3schools.com>

<https://getbootstrap.com>

<https://codepen.io>

<https://chat.openai.com/chat>

<https://www.wikipedia.org>

<https://randomnerdtutorials.com>

<https://console.firebase.google.com>