**נושא הפרויקט: מערכת לזיהוי חריגות סביבתיות אצל תינוקות**



שמות מגישים: דרור רוסין 301795142, שגיא ראובן 308446160, יהודה יחיאל שכטר 312256050.

שם המנחה: ד"ר אופיר דן.

**תוכן עניינים**

תקציר הפרויקט.......................................................................................

מטרת הפרויקט........................................................................................

קהל יעד.................................................................................................

אלגוריתמים............................................................................................

האתגר הטכנולוגי.....................................................................................

אופן מימוש אב טיפוס................................................................................

רכיבי הפרויקט......................................................................................

דרישות האפליקציה..................................................................................

תיאור הניסוי.........................................................................................

מודולים עיקריים......................................................................................

איורים והמחשות.....................................................................................

אילוצים ונקודות תורפה..........................................................................

יכולות האפליקציה................................................................................

תוצאות ומסקנות..................................................................................

מדריך למשתמש והתקנה...........................................................................

סקר ספרות............................................................................................

**תקציר**

מערכות לזיהוי חריגות סביבתיות קיימות אצלנו בתעשייה זמן רב. בפרויקט שלנו התמקדנו במערכת שתעבוד בקרב תינוקות בפגייה. הסיבה לכך היא, כי במקום זה קיימים מספר תינוקות הגדול ממספר האחיות באופן משמעותי, דבר זה יכול להקשות על האחיות לשים לב לאירועים שונים הקורים לתינוק, ולטפל בו בהתאם. אי לכך ובהתאם לזאת קיים צורך למערכת, שתתריע על מקרי חרום. בזה נוכל להיעזר במערכת שלנו. נתרכז בפרמטרים הבאים:

1. **טמפרטורה**

2. **לחות**

3. **דום** **לב, הפסקת נשימה**

הסיבות המרכזיות: אבטחתו של הילוד וחסכון בכוח אדם.

האמצעים בהם נשתמש הם:

1. **רספברי** **פאי**: אנו נשתמש בגרסה ה- 2 שלו מודל B+. המכשיר הינו מחשב לוח יחיד קטן וזול עם חיבורים מתאימים אשר דרכו יהיה אפשר לסנכרן את החומרה לתוכנת מחשב הרלוונטית.
2. **סנסור טמפרטורה**-**לחות** מותאם, חישן דופק לב (babycare) , חישן רעש.
3. **מטריצת חיבורים** שעליו מולבש החיישן.
4. **חומרה אלקטרונית**: כבלים של מוליכים, נגדים, נורות.

התוכנה תשלב בין האמצעים הנ"ל בעזרת המודולים הבאים:

1. תקשורת: אינטרנט, Wi-Fi, שליחת מידע (מסרונים).
2. מערכות מסדי נתונים: היסטוריה של אירועים מלווים בנתונים סטטיסטיים.
3. אלגוריתמים: חישוב הרכב צוות אופטימאלי לאחזקת המערכת (אחיות, טכנאים). מודול אחר ימפה את מרחב החישנים במחשב.
4. בצוע סימולציה לצורך debugging ולפני הוצאת הפרויקט לציבור הרחב.
5. קביעת קריטריונים למקרי חירום למיניהם כגון: טמפרטורה מעל סף מסוים, רעשי רקע.

מימוש התוכנה התבצע בשפת Python והייתה יעילה יותר לסנכרון המודולים השונים.

במידה ונרצה בעתיד למסחר את המערכת, יהיה צורך להעביר את המוצר בתהליך הכשרה ומתן תו תקן והיתר שימוש.

**מטרות הפרויקט**

אבטחתו של הילוד ,עדכון על מצבו הנוכחי בכל עת להורים ולצוות הרפואי, חסכון בכוח אדם.

נתינת מענה מיידי על ידי הגעה לילוד הנכון, נעשה על ידי רכיב גי' פי אס בלוח.

בנוסף את מוצר זה ניתן להרחיב להשמיש אותו לדברים אחרים כגון: מערכת השקיה, חממה,

טמפרטורה בחוות שרתים ובכל מקום אפשרי אחר.

הפקת תועלת משנית מהפרויקט על ידי ניצול תוכנית הלימודים במיוחד מהקורסים הבאים:

מסדי נתונים, מבנה תוכנה, מבנה נתונים ואלגוריתמים.

במידה ונרצה בעתיד למסחר את המערכת, יהיה צורך להעביר את המוצר בתהליך הכשרה ומתן

תו תקן והיתר שימוש.

בנוסף, למידה והרחבת אופקים ברספברי-פאי, תחום החיישנים, שפת פייתון, שימוש במערכת

ההפעלה לינוקס, עבודה נכונה בצוות, עמידה בייעדי הפרויקט.

עבודה על פרויקט בסטנדרטים של תעשיה.

**קהל יעד**

קהל היעד הינם הורים טריים, רופאים והצוות הרפואי האחראי במקום המיועד.

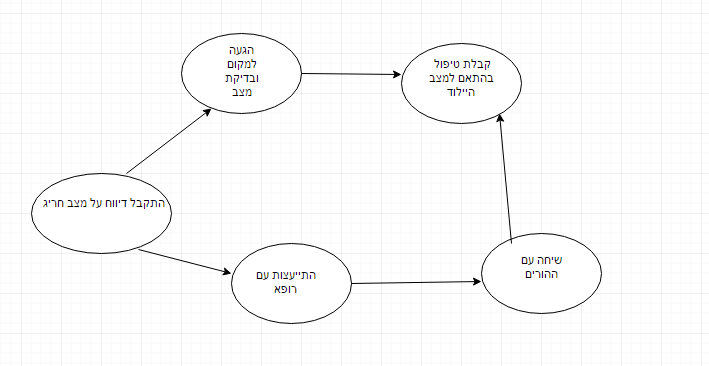
כמו שצוין במטרת הפרויקט לעיל, יכול להתאים גם למגזר ציבורי ועסקי מבחינת שימושי

מערכות השקיה וחוות שרתים ועוד...

**אלגוריתמים**

1. **חישוב הרכב צוות רפואי אופטימאלי** לאחזקת המערכת. לעשות תצפית וסימולציה לשכיחות קריאות לאחיות, להכפיל במשך ממוצע של טיפול, להתחשב בקריאות מקבילות. נשתמש בבניית דיאגראמת (CPM) PERT, לחישוב הנתיב הקריטי.

דיאגרמת CPM: לחישוב הנתיב הקריטי כולל זמנים משוערים



T = 0.5 min

דיאגרמת CPM לחישוב הנתיב הקריטי

T = 1 min

דיאגרמת CPM לחישוב הנתיב הקריטי

T = 0.5 min

דיאגרמת CPM לחישוב הנתיב הקריטי

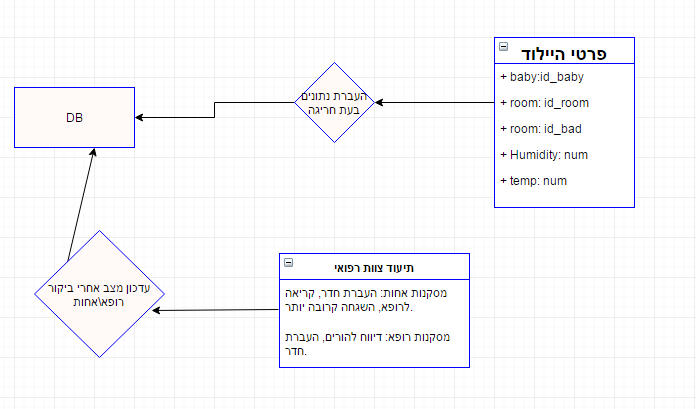
T = 3 min

דיאגרמת CPM לחישוב הנתיב הקריטי

T = 1 min

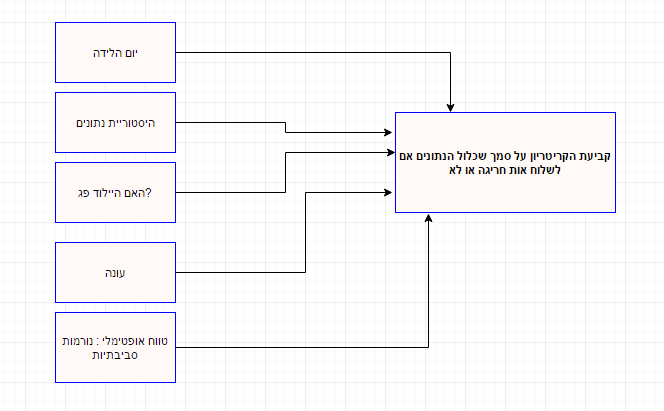
דיאגרמת CPM לחישוב הנתיב הקריטי

2. **הכנסת נתונים אופטימאלית** למסד הנתונים כלומר, למנוע כניסת נתונים מיותרים, ומצד שני למנוע התעלמות מנתונים חשובים.



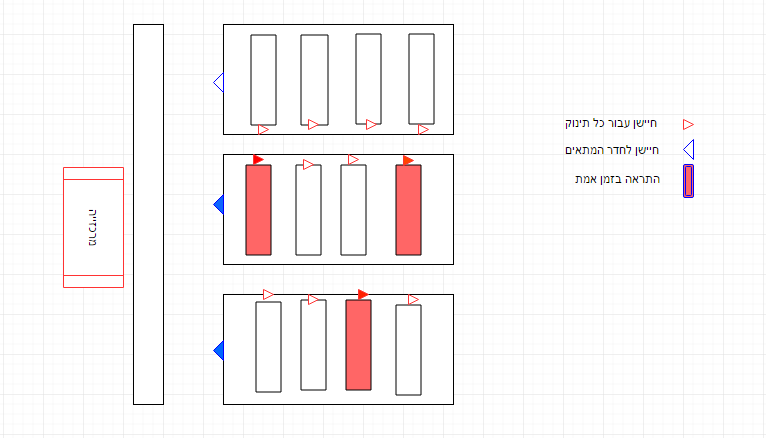
תרשים זרימה של הכנסת הנתונים למסד הנתונים

1. **חישוב תחומי חריגות** לפי הקריטריונים.



תרשים זרימה להצגת הרעיון של הגורמים המשפיעים על קביעת תחום הקריטריון

4. **מיפוי** – בניית מפת מיקום הילדים בבית הילדים (מעון, בית חולים, וכו') שיאפשר גישה מהירה של הצוות בעת מצוקה. ילדים הגורמים לחריגים לעתים תכופות יותר ימקמו קרוב יותר לצוות (לקצר "ריצות").



דוגמא להמחשה של מרחב חיישנים והתראות בזמן אמת.

**האתגר הטכנולוגי**

המערכת כוללת היבטים שונים כגון: ניהול בסיסי נתונים באופן יעיל, תקשורת אינטרנט, מיפוי

מרחב חיישנים, אנליזת נתונים לצורך קביעת קריטריון מתאים, מבחינת ההיבט של הציוד

האלקטרוני. האינטגרציה בין גורמים אלו מהווה אתגר טכנולוגי, היות ושאיפתנו לגרום

לעבודה מתוזמנת ויעילה של כל הרכיבים יחדיו.

האפליקציה כוללת מאגרי מידע שונים אשר מוצגים למשתמש בהתאם למצב בו הוא נמצא.

האפליקציה כוללת מחלקות רבות, חיבור של כל המחלקות הללו יחדיו המשתמשים במידע

משותף ושמירה על סדר התהליכים דורש הבנה מעמיקה של שלבי האפליקציה בכדי למנוע

גלישה של המשתמש אל מידע שכבר לא רלוונטי עבורו בהתאם לשלבי המשתמש.

חלק ממאגרי המידע הם קבצים הנשמרים על הראספברי פאי שבכדי לבצע שמירת קבצים כאלה

אנו נדרשים לקבל הרשאות גישה של הלוח.

אתגר טכנולוגי נוסף מולו אנו עומדים הוא שליחת מסרונים לקבוצה המתאימה.

התאמת הממשק עבור מערכת ההפעלה והפלטפורמה בו המשתמש עובד. חיסכון וניצול זיכרון

יעיל יותר.

**אופן מימוש אב טיפוס**

האפליקציה מומשה על גבי לוח ראספברי פאי, מערכת ההפעלה שלו היא Raspbian, ניתן

למצוא דמיון למערכת הפעלה לינוקס, שימוש במסד נתונים MySQL, שפת מימוש כאמור

, Python במאגרי המידע נעשה שימוש בקבצי CSV (אקסל) וDB לכתיבת המידע הדרוש ואלו

נשמרים על גבי הלוח לצורך פתיחה, עריכה, שינוי ובקרה.

1. **Python** **-** המערכת כוללת אפליקציה שבה כל הפונקציות הרלוונטיות. האפליקציה תמומש בשפת Python**.**
2. **MySQL-** לצורך ריכוז הנתונים והמידע, ננהל את בסיס הנתונים תחת MySQL.
3. **RP2 -** בצד החומרתי נשתמש בלוח של RP2 כאשר המערכת הפעלה שצרובה עליו מפעילה את האפליקציה שלנו. חיישן הטמפרטורה-לחות מורכב על המטריצת חיבורים יחד עם המוליכים עליו.

יצרנו UI (ממשק GUI למשתמש) מתאים וידידותי למשתמש לצורך שליטה במערכת וניהול

שלה.

**רכיבי הפרויקט**

המערכת מכילה את המודולים העיקריים הבאים:

1. תקשורת: אינטרנט על ידי Wi-Fi או חיבור קווי, שליחת מידע (מסרונים): המערכת תחובר לרכיב חומרה של אינטרנט אלחוטי מקומי שתאפשר לשלוח את ההודעות.
2. מערכת מסדי נתונים: היסטוריה של אירועים מלווים בנתונים סטטיסטיים בהתאם ליילוד: רעש, טמפרטורה ולחות, תאריך, שעה, דופק.
3. אלגוריתמים: חישוב הרכב צוות אופטימאלי לאחזקת המערכת (אחיות, וטכנאים), אלגוריתמי הכנסת נתונים אידיאלית למסד נתונים וחישוב תחומי חריגות לפי הקריטריונים המתאימים.
4. מיפוי מרחב החיישנים: המערכת תראה את מפת פריסת החיישנים המקומית, דבר העוזר להגיע למקום הנכון.
5. בצוע סימולציה לצורך הרצת בדיקות שפיות, פונקציונליות, מערכת, רגרסיה לפני הוצאת הפרויקט לציבור הרחב.
6. קביעת קריטריונים למקרי חירום למיניהם כגון: טמפרטורה/לחות מעל סף מסוים או תחום נורמטיבי מסוים בהתאם ליום הלידה, עונה, אם הוא פג או לא, נורמות סביבתיות, נורמות שהוגדרו על סמך היסטוריית נתונים.

**חומרה**

1. **רספברי** **פאי** **(ראה איור מס' 5)**: אנו נשתמש בגרסה ה- 2 שלו מודל B+. המכשיר הינו מחשב לוח יחיד קטן וזול עם חיבורים מתאימים אשר דרכו יהיה אפשר לסנכרן את החומרה לתוכנת מחשב הרלוונטית.
2. **סנסורים: טמפרטורה**-**לחות (ראה איור מס' 6)** מותאם, חיישן דופק, חיישן רעש.
3. **מטריצת חיבורים** **(ראה איור מס' 4)** שעליה מולבשים החיישנים.
4. **חומרה אלקטרונית**: כבלים של מוליכים, נגדים ונורות.

**תוכנה**

פלטפורמת הפיתוח היא סביבת העבודה PyCharm בשפת Python על גבי מערכת ההפעלה

של Raspbian המדמה לינוקס, שנצרבה על הלוח של RP2 בהתאם לפרויקט.

השתמשנו גם בPython Shell לצורך הרצה ודיבאג נוח יותר. גרסה 3.

האתגר הטכנולוגי יהיה כאשר התוכנה תשלב ותסנכרן בין המודולים לעיל לבין החומרה עצמה.

אופן העבודה חולק בין השותפים השונים כך שידוע בדיוק מה כל אחד עושה ועל מה הוא

אחראי וכמה זמן נותר לסיום העבודה על אותו החלק על מנת להתקדם למודול הבא במערכת.

בקורס מבנה תוכנה רכשנו ניסיון בשפה הסקריפטית Python על הסביבה PyCharm

ועכשיו נרצה יותר להתמקצע בשפה ולממש פונקציות מורכבות הרלוונטיות לנו (ראה [3]).

הספרייה המרכזית נקראת: Adafruit Python DHT Sensor Library שמאפשרת לנו את

קבלת הנתונים מהחיישן של טמפרטורה/לחות לתוכנה (ראה [5]).

ספריית נורה נקראת: Pigpio. נעזרנו בה לצורך הפעלת הנורה בזמנים הנכונים והתראות.

ספריית מפות:

ספריית מיקרופון/רעשים: נקראת pyalsaaudio. נעזרנו בה לצורך קבלת קולות/רעשים/תזוזה מהילוד.

**דרישות האפליקציה**

1. המערכת תעבוד על גבי החומרה המתאימה, תעבוד באופן רציף וביעילות.
2. המערכת תתמוך בזיהוי היילודים באופן מאובטח ונוח.
3. המערכת מחייבת קישוריות לאינטרנט ((Wi-Fi.
4. המערכת תאפשר ניתוח תוצאות בזמן אמת והצגתם לפי פרופיל גמיש.
5. כלל התוצאות תשמרנה בבסיס נתונים מאובטח ומגובה בענן.
6. המערכת תהיה קלה ופשוטה לפיתוח ותאפשר הגעה לגרסה ראשונית בזמן קצר ובעלות נמוכה.
7. המערכת תשלח התרעה של הודעות בזמן אמת לגורמים הרלוונטיים.
8. המערכת תהיה פשוטה וקלה לתפעול על ידי האחיות. ניתן יהיה על ידי חפיפה קצרה להבין את תפעול המערכת.
9. המערכת תלווה בתיעוד ממצה וקל להבנה.

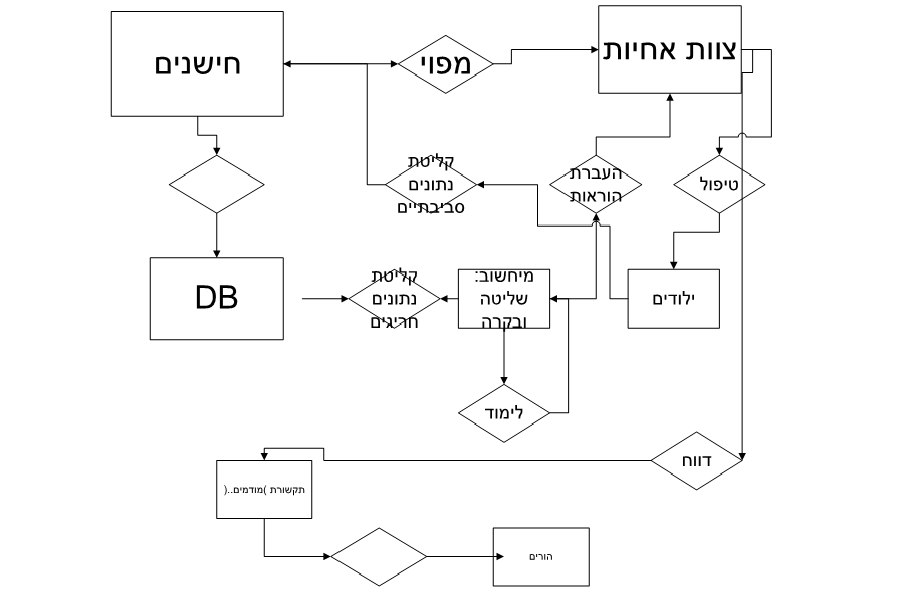
**תיאור הניסוי**

במהלך העבודה ביצענו את התהליכים הבאים:

1. ***חיישנים*** - סימולציות שונות על מנת לדמות טמפרטורה ולחות סביבתיים במצבים שונים, וזאת על ידי שינוי הטמפרטורה במזגן המקומי למשל. בעזרת ניסוי זה, ניתן יהיה לדעת שהמערכת עובדת כמצופה.
2. ***מסדי נתונים*** - בדיקת השפעת התראות מתאימות על מסדי נתונים חריגים למסד הנתונים וכו'. בנוסף, נבדוק שאין דריסה של נתונים בטבלאות ושהכל נשמר בצורה תקינה. יתר על כן, נעשה גם ניסוי כאשר הקריטריון של תחום חריגות השתנה. בוצעו בדיקות מערכת על מנת לוודא כי אין כפילויות של הודעות.
3. ***תקשורת*** - יבוצע גם ניסוי על מקום שהקליטה האלחוטית בו ירודה, נשתמש ב-redundancy, ובמקום זה מפת החיישנים תתקבל ע"י לווין עד כשהקליטה תשתפר ותחזור להיות יציבה.
4. ***ממשק משתמש*** – הפעלת המערכת ממספר מחשבים בו זמנית על מנת לבדוק עומסים ושאינה גוררת דריסת נתונים עקב כך. בדיקת פונקציונליות של הלחצנים ושהם עובדים כמו שרצינו.
5. ***הודעות***- נבדק העניין של כפילות הודעות לאדם מסוים, כלומר עבור על התראה הודעה אחת בלבד.

**מודולים עיקריים**

להלן דיאגרמת ERDלהצגת המודולים בפרויקט.



העברת מידע

שליחת הודעות

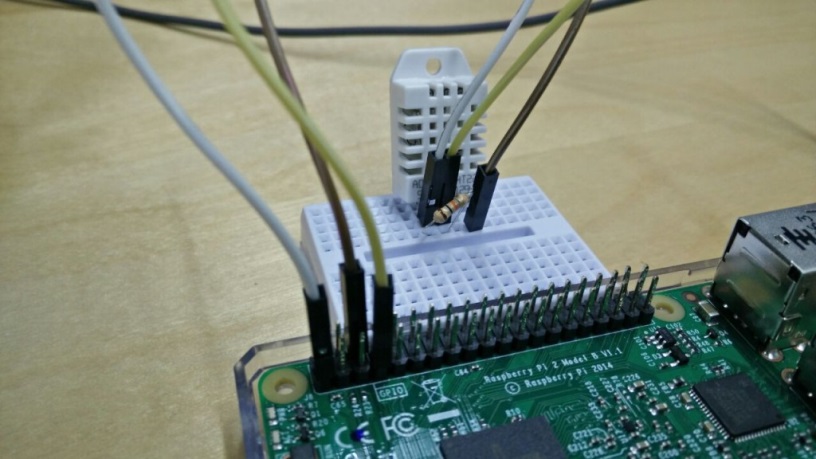
לפי הדיאגרמה הנ''ל ניתן להבין את הישויות הפועלות במערכת, וכן את היחסים ביניהן.

**איורים והמחשות**

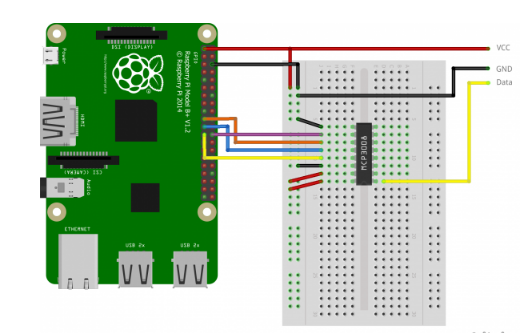
טמפרטורה ולחות:

****

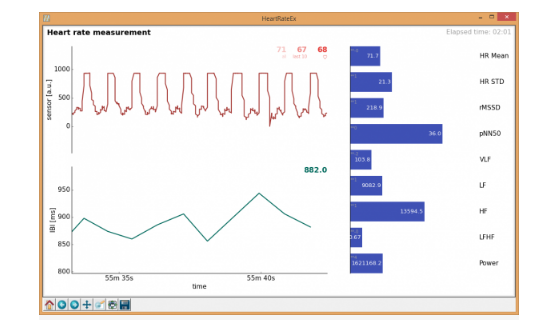
המחשת חיבור ראשוני של הרכיבים לראספברי פאי 2:



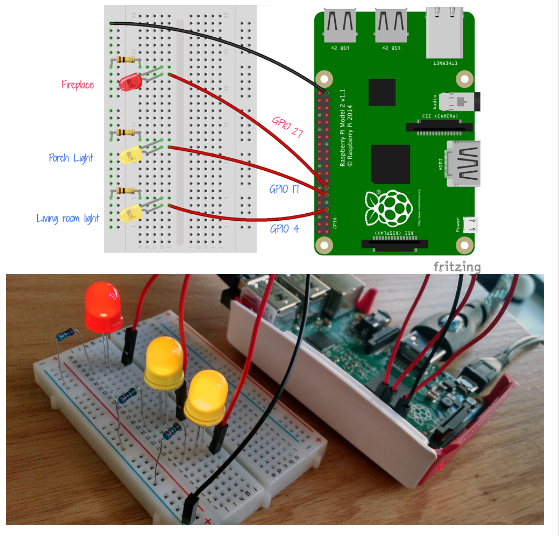
המחשת חיבור חיישן דופק לב ל RPI



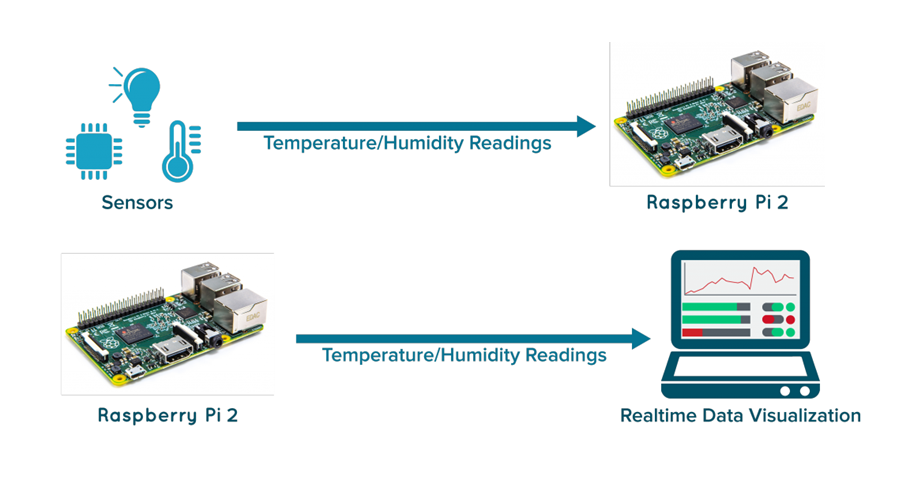
: ניתן לראות תוצאות נתונים של הדופק והיסטוריה



המחשת חיבור נורה והפעלתה:

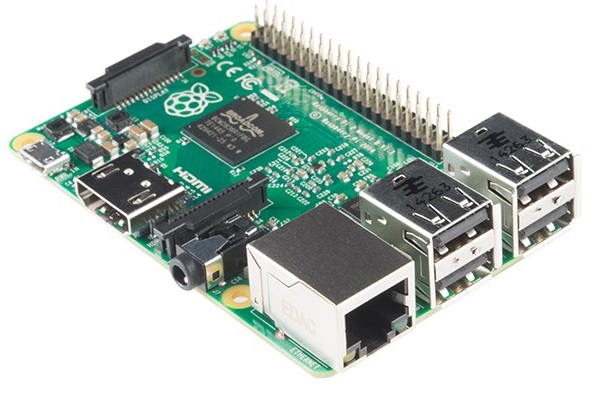


דיאגרמה להבנת הקונספט של העברת הנתונים.

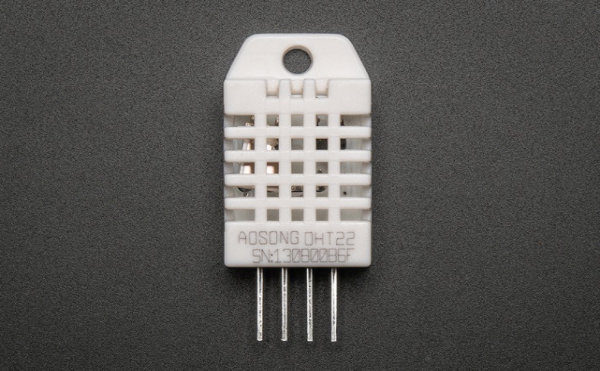


מטריצת חיבורים:





RP2



חיישן טמפרטורה-לחות DHT22

**אילוצים ונקודות תורפה**

המערכת מתבססת על טכנולוגיית קוד פתוח. נצרכנו לבצע שינויים והתאמות כדי להרכיב את המערכת שלנו. מכאן עלולות לנבוע בעיות של גרסאות בחלקים השונים אשר נצטרך להתגבר עליהן. את המיקרופון וחיישן התנועה לא הוספנו להדגמה אלא רק לצורך שימוש וירטואלי עם הספריות המתאימות שמצאנו להן.

**יכולות האפליקציה**

האפליקציה תבוא לידי ביטוי ותופעל לאחר הלידה של האם. הילוד יקבל מספר סידורי עוקב על מנת לעקוב אחריו וייכנס מייד למסד הנתונים של המקום. כך ניתן יהיה לבצע מוניטורינג נכון שיביא לנו תוצאות אודותיו. האפליקציה מציעה מפה של מרחב חיישנים על מנת להתמצא במקום. שליחת מסרונים של אותות מצוקה חריגים לגורמים המתאימים: הורים וצוות רפואי על מנת להאיר את תשומת ליבם ולגשת מיידית. הדפסת דו"ח מעקב של היילוד. כניסה למערכת שליחת הודעות והכנסת למאגר נתונים שם. האפליקציה בנוסף מאפשרת לעקוב אחרי מספר ילדים במקביל וזו על ידי הרחבה של המערכת לחדרים אחרים. ההורים יוכלו להסיר את ההתראות שהם מקבלים בכל עת.

**תוצאות ומסקנות**

מבחינת האתגר הטכנולוגי הצלחנו ליצור אינטגרציה בין הצד החומרתי לצד התוכנתי בצורה טובה: שליטה מרחוק על המערכת שבנינו, תפעול של הספריות השונות בפייתון, שימוש בתהליכונים (Threads) לתזמון נכון והצלחנו לקבל תוצאות כמו שציפינו. צריבה של מערכת ההפעלה על גבי rpi-2 ,יצירת בסיס נתונים בצד השרת, הבסיס נתונים מתנהל מול כרטיס זיכרון מוגבל אבל ניתן להרחבה. כמובן שבבתי חולים יוחזק storage הרבה יותר גדול .

**מדריך למשתמש והתקנה**

**סקר ספרות**

**Academic papers:**

[1]  Hendrickson, Chris; Tung, Au (2008). ["11. Advanced Scheduling Techniques"](http://pmbook.ce.cmu.edu/11_Advanced_Scheduling_Techniques.html). [Project Management for Construction](http://pmbook.ce.cmu.edu/). cmu.edu (2.2 ed.). Prentice Hall. [ISBN](https://en.wikipedia.org/wiki/International_Standard_Book_Number) [0-13-731266-0](https://en.wikipedia.org/wiki/Special:BookSources/0-13-731266-0). Retrieved October 27, 2011.

-We used this article to gain knowledge of building a PERT or CPM (Critical Path Method) diagram and understand its meaning. It's easier to understand the algorithm according to this chart.

# [2] Correct Humidity Level- <https://www.askdrsears.com/news/sears-family-blog/correct-humidity-level> - מאמר שמציג ממצאים על לחות נורמטיבית תקינה של תינוק

# [3] - טמפרטורה/חום התינוק http: //www.tipa.co.il/articlePage.asp?articleId=66

מאמר של טיפת חלב שמציג נתונים עדכניים אודות טמפרטורה סביבתית של תינוק

**Books:**

[2] Basic book of Electrical Engineering: <http://www.school.kotar.co.il/kotarapp/index/Book.aspx?nBookID=94920013>.

-We used this book to acquire basic information about our hardware devices and their electrical connection.

[3] Python Crash Course: A Hands-On, Project-Based Introduction to Programming.

- Excellent learning book. Guide to learn Python language in a quick, efficient and convenient.

**Existing Software:**

[4] PubNub: <https://github.com/pubnub/pi-house>. Here is the open source used in the application before editing. The code is in the Python language.

# [5] Adafruit Python DHT Sensor Library: <https://github.com/adafruit/Adafruit_Python_DHT>. Here is the library we used to synchronize between the sensor and the software and receiving data.

# [6] [pyalsaaudio](https://github.com/larsimmisch/pyalsaaudio) : <https://github.com/larsimmisch/pyalsaaudio/>

# We used to synchronize between the microphone and the software and receiving data by sound.

# [] Pigpio - <https://github.com/joan2937/pigpio>

# We used this library to turn the bulb at the right times.

# [7] Raspberry Pi Heartbeat / Pulse measuring:

# https://tutorials-raspberrypi.com/raspberry-pi-heartbeat-pulse-measuring/

# [8] MySQL: <https://github.com/PyMySQL/PyMySQL>. We took ideas form here in order to set up a database. The functions are update, delete and add entries.

**Similar projects:**

<http://www.upfile.co.il/file/611049344.html>. In this project, we used to learn about wireless sensor network, correct assembly its efficient implementation.

[10] “Angel Care”: https://www.angelcarebaby.com

Angelcare wants parents to understand they're not alone in this new adventure of parenthood. We're always here using innovative, research-based design and advanced technology to help keep babies safe, supported and squeaky clean. Angelcare products have received countless awards for their ease of use, seamless user experience and overall quality. Our goal is to reduce the inevitable stresses of infant care with 24/7 peace of mind. With Angelcare by your side, you'll be able to savor every joyful moment of your new life.

**Our project links :**

**Github:**

<https://github.com/drorruss/Final_Project>

אתר הפרויקט בפלטפורמת גיטהאב: כאן ניתן לראות את קבצי הפרוייקט כולל מצגת, דוחות ופוסטר.

**Youtube:**

<https://www.youtube.com/watch?v=by_GR2tYN24http://moodle.ariel.ac.il/mod/assign/view.php&t=15s> סרטון להצגת אב טיפוס עובד

**Model projects:**

<http://moodle.ariel.ac.il/course/view.php?id=322> נמצאים כאן שאר הדוחות בפרויקט