**טכנולוגיות אינטרנט מתקדמות (WEB)**

**הגשת פרויקט**

קבוצה 11 Journalism Interview Helper

|  |  |
| --- | --- |
| שם חבר.ת הצוות | תז |
| קרן קזצ'ינסקי | 322845504 |
| אוריאן חזיזה | 211921028 |
| רחל אבידן-קינג | 315329508 |
| דניאל שחר | 314983214 |
| ליהי בן עמרה | 208338152 |

**תקציר הפרויקט:**הפרויקט שלנו מהווה אתר לעיתונאים העורכים ראיונות ועשוי להיות הפלטפורמה החדשה לניהול יעיל של הראיונות שלהם.   
באתר יש סוג משתמש יחיד מסוג מראיין ובהתחברותו לאתר יכול לבצע מספר דברים. ראשית, המראיין יכול ליצור ראיון חדש באתר ולמלא בו את הפרטים לגביו: המרואיין, תאריך, הנושא.  
האתר מציע התממשקות לGemini שמטרתה לייצר שאלות חדשות למראיין בהתאמה לנושא שיבחר. בנוסף, קיים אופציה למראיין להוסיף גם שאלות משל עצמו.   
כמו כן, חלק עיקרי באתר הינו עמוד ההיסטוריה של המראיין שהוא מהווה מאגר של כל הראיונות שערך באתר, וגם קיימת לו אופציה להעלות לטבלה זו ראיון אשר ערך בקובץ וורד ולא באתר. פונקציונליות זו מאפשרת למראיין ארגון מסודר של המידע.   
פונקציונליות נוספת הינה סינון בטבלת ההיסטוריה של הראיונות, על מנת להקל על המראיין במציאת ראיון ספציפי מרשימת הראיונות שערך ולאפשר גישה מהירה למידע.   
הסינון יכול להתבצע לפי אחת מהדרכים הבאות: נושא, תאריך, שם מרואיין.   
האתר בעל ממשק רספונסיבי מה שמאפשר שימוש נוח מכל מכשיר, ובנוסף בעל אופציה לDark mode על מנת לענות לצורכי המשתמשים.

**הטכנולוגיות המרכזיות:**Backend – node js deployed on VERCEL, Firebase

Frontend – React

DB – Firebase- interviews/users (Real Time DB)

Styling – Tailwind Css

**קישור לGIT :** [**https://github.com/RachelAvidanKing/2024WEB**](https://github.com/RachelAvidanKing/2024WEB)

**קישור לאתר :** [**https://2024-web-rachelavidankings-projects.vercel.app/**](https://2024-web-rachelavidankings-projects.vercel.app/)

תמונה שמכילה טקסט, צילום מסך, מערכת הפעלה, תוכנה

התיאור נוצר באופן אוטומטי

מהנדס מערכת – ליהי בן עמרה

חלוקת העבודה מול הצוות נעשה בשיתוף פעולה עם הצוות, כל אחת לקחה על עצמה את החלק שידעה שתבצע על הצד הטוב ביותר. היה ממשק בין חברי הצוות וכל חבר צוות קיבל אסמכתה לסיום המשימה שלו מכל חברי הצוות על מנת שהתוצאה תהיה מיטבית ותהיה ביקורת על חלקי העבודה. נקבעו זמני עבודה משותפים, וכל המשימות מולאו עד לתאריך ההגשה.

|  |  |  |
| --- | --- | --- |
| **שם חבר הצוות** | **משימות שהוקצו** | **משימות שהושלמו** |
| אוריאן חזיזה | דיאגרמות ארכיטקטורת אתר | הכל |
| קרן קזצ'ינסקי | תיקון קוד- הוספת אופציה לסינון של ראיונות קודמים, סידור פונט ושיפור רספונסיביות | הכל |
| ליהי בן עמרה | כתיבת תיק מתכנת | הכל |
| רחל אבידן-קינג | תיקון דרישות + דיאגרמת DB | הכל |
| דניאל שחר | כתיבת תיק משתמש | הכל |

**דרישות**

**דרישות פונקציונליות:**

1. המערכת תאפשר הרשמה והתחברות אישית למראיין.

2. המערכת תאפשר מילוי פרטי ראיון.

3. המערכת תאפשר הורדה והעלאה של הראיונות המתומללים.

4. המערכת תאפשר שמירת נתונים של הראיון.

5. המערכת תאפשר צפייה בהיסטוריה של הראיונות הקודמים.

6. המערכת תאפשר שינוי ערכת נושא- יום/לילה.

7. המערכת תאפשר למשתמש להקליד שאלות בעצמו.

**דרישות לא פונקציונליות:**

**אבטחת מידע (Security)**

1. בהקלדת הסיסמה יש הצפנה

**נתונים ושמירה (Data integrity)**

2. הפרטים שישמרו לכל משתמש בהרשמה הם: שם, סיסמה, שם משתמש ועיתון שבו הוא עובד.

3. הנתונים שנשמרים במערכת עבור כל ראיון הם: שם המרואיין, תאריך, נושא ושאלות/תשובות הראיון.

**גמישות (Flexibility)**

4. המערכת הינה רספונסיבית ומתאימה לכל גדלי המסכים.

**שימושיות (Usability)**

5. המערכת כוללת הסברים על פעולות משתמש (למשל: העלאת קובץ).

6.המערכת כוללת מנגנונים למניעת טעויות משתמש

**דיאגרמת ארכיטקטורת אתר – High level**

Frontend

React Framework

Connection  
General  
History  
NewInterview

Backend

API  
Layer

React Components

Business  
Logic

React  
Hooks

Tailwind  
Css

Interview  
Management

Authentication

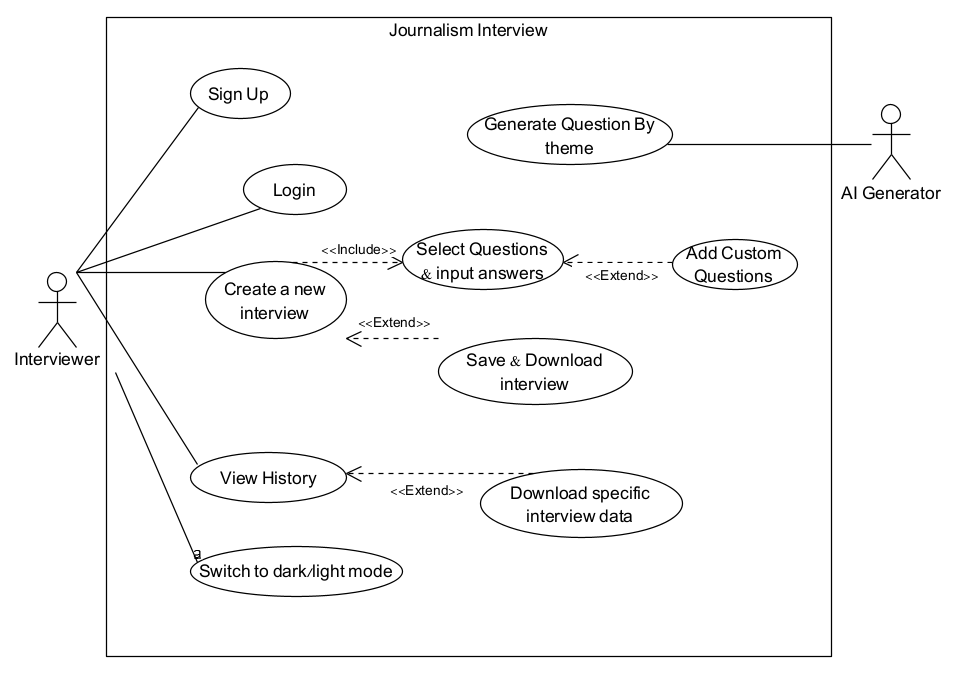
External services

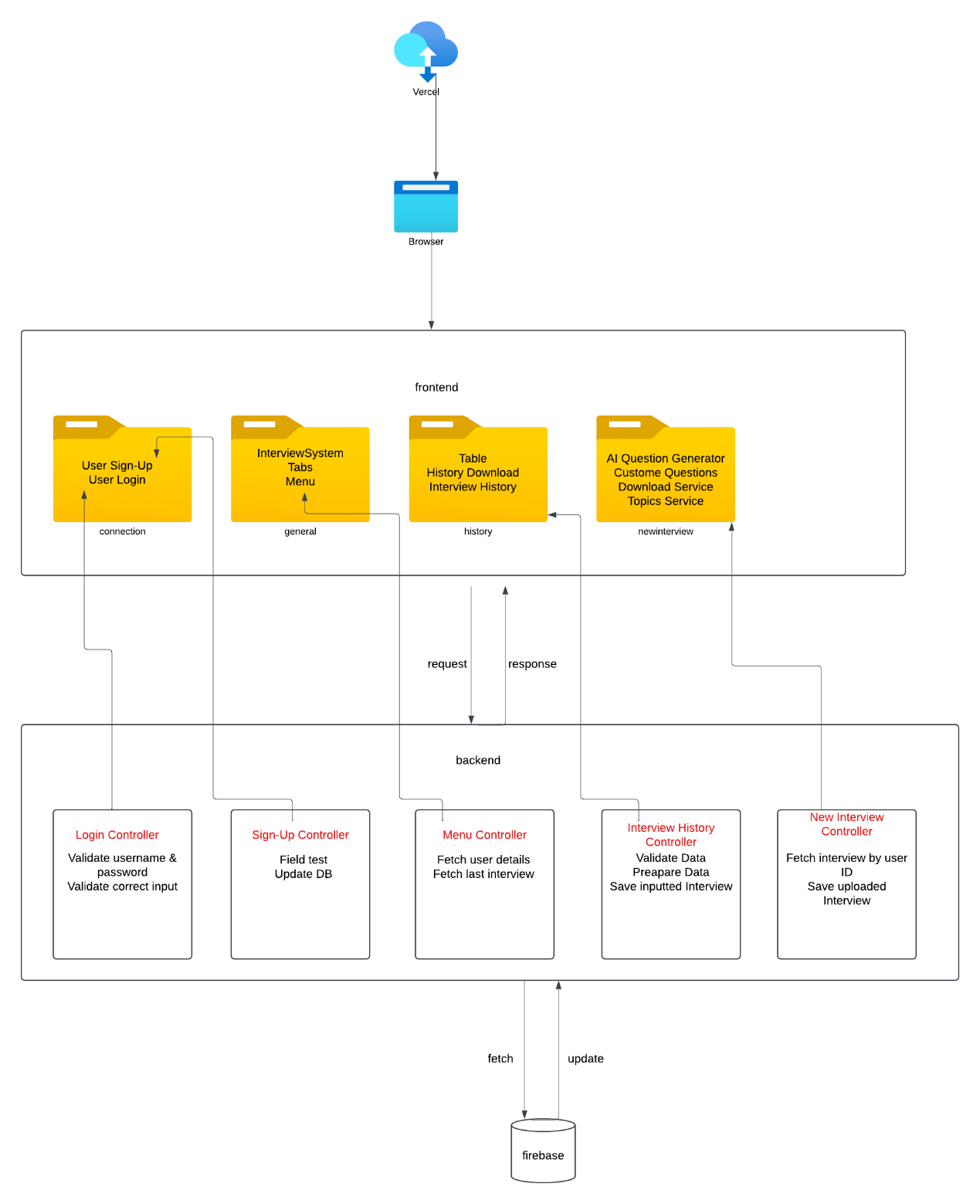
Database

Gemini   
API

Firebase

**דיאגרמת Use case**



******דיאגרמת ארכיטקטורת אתר- מבנה סופי כולל קומפוננטות ותיקיות**

newinterview (Interview Page)

history (History Page)

backend

frontend

Connection (Login & Sign-Up pages)

General for website

דיאגרמת **DB**

Date

intervieweeName

Interview\_id

Topic

**questionsAndAnswers**

**Interview**

Has

UserId

belongs

Interview\_id

Answer

question\_id

**Interviewer**

question

password

username

UserId

newspaper

email

**דיאגרמה זו מתארת את מסד הנתונים שבנינו בFirebase   
שממנו נשלף רוב המידע בהתאם לצרכים הרלוונטיים.   
בנוסף מתבצעת שליפת מידע מ- API של Gemini שאופציה זו מפורטת בתיק מתכנת וממנו שלפנו את השאלות שהצאט ייצר עבור המראיין שלנו בהתאם לנושא שבחר.**

**תמונת מסך של המסד-**

**תמונה שמכילה טקסט, צילום מסך, עיצוב

התיאור נוצר באופן אוטומטי**

תיק מתכנת

פונקציות מרכזיות בקוד:

**general**

**InterviewSystem.jsx –** קומפוננטה האחראית לניהול של כל האתר. ניהול התפריט, סרגל הכלים והרינדור בין הדפים.

**MenuController.jsx –** קומפוננטה שאחראית על הבאת פרטי משתמש מהDB והראיון הכי עדכני שנעשה (מוצג בתפריט הנשלף)

**history**

**HistoryDownload.jsx-** קומפוננטה האחראית להורדת הראיון הנוצר באתר ונשמר בDB כקובץ WORD

**InterviewHistoryController.jsx –** קומפוננטה שאחראית על שמירת ושליפת המידע של הראיונות בDB

**InterviewHistory.jsx –** קומפוננטה שמציגה את טבלת ההיסטוריה של ראיונות המראיין וכל ההסברים הנדרשים לצורך העלאת ראיון הנוצר חיצונית לאתר.   
בנוסף, נותנת אופציה לשמירה, מחיקה, הורדת הראיון הנוצר באתר וסינון של הראיונות הקיימים.

**newinterview**

**AIQuestionGenerator.jsx –** קומפוננטה שאחראית על הממשק עם הAPI של הבינה המלאכותית ועל הייצור של השאלות לפי הנושא המבוקש

**CustomQuestionInput.jsx-** קומפוננטה שאחראית על האופציה של הוספת שאלות מותאמות של המראיין לראיון הנוצר באתר

**DownloadService.jsx-** קומפוננטה שאחראית על הורדת הראיון שנוצר באתר ומוודאת שהמשתמש הכניס את כל השדות הרצויים

**InterviewController.jsx-** קומפוננטה שאחראית על בדיקת הקלט של המראיין עבור שמירת הראיוןומבצע את השמירה לDB**.** בנוסף מסדר את הקלט שישמר בצורה מיטבית

**TopicsService.jsx –** קומפוננטה שאחראית על כל הניהול של יצירת ראיון באתר ועיצוב הדף

**firebase.jsx-** קומפוננטה המכילה מתודות הקשורות להתממשקות הDB לאתר

קטעי קוד מיוחדים שלקחנו ממקורות שונים:

-**נעזרנו בקטע קוד הלקוח מהקישור לאתר על מנת להוריד את הראיונות שערך המראיין באתר כקובץ וורד:** <https://docx.js.org/#/>

-**נעזרנו בקטע קוד הלקוח מקישור ה**GitHub **המצורף שמפצל בין השורות של ה**Word **בשביל להעלות אותם לטבלת הראיונות:** <https://github.com/mwilliamson/mammoth.js/?utm_source>

**:API שימוש בKEY API של Gemini-בינה מלאכותית שנעזרנו בה על מנת לייצר שאלות לפי הנושא שהזין המשתמש.  
קטע הקוד בשפת** JS **נלקח מאתר ליצירת מפתחות** API **:**

[https://ai.google.dev/gemini-api/docs/get-started/tutorial?lang=node&\_gl=1\*1b2laq4\*\_ga\*MTI4NTM0MDIxOS4xNzM2MzQ0NjIz\*\_ga\_P1DBVKWT6V\*MTczNzM2NzY0NS42LjEuMTczNzM2NzgwNS41Ny4wLjE0NjI4MDU0ODA.&hl=he](https://ai.google.dev/gemini-api/docs/get-started/tutorial?lang=node&_gl=1*1b2laq4*_ga*MTI4NTM0MDIxOS4xNzM2MzQ0NjIz*_ga_P1DBVKWT6V*MTczNzM2NzY0NS42LjEuMTczNzM2NzgwNS41Ny4wLjE0NjI4MDU0ODA.&hl=he)

**סביבות מיוחדות שהתקנתם:**

Vite – סביבת עבודה לייצור פרויקטים בREACT באופן מהיר

Tailwind css – סביבת עבודה לעיצוב הפרויקט

React Dom – סביבות עבודה לרינדור רכיבי REACT לתוך הDOM בדפדפן

ספריית lucide-react – שימוש לייבוא אייקונים של התפריט ומצב יום/לילה

ספריית react-icons/fa – שימוש לייבוא אייקונים של הורדה ושמירה של הראיונות

ספריית mammoth – ספרייה להעלאת קבצי הוורד של הראיונות

ספריית docx – ספרייה ליצירה והורדת קובץ וורד

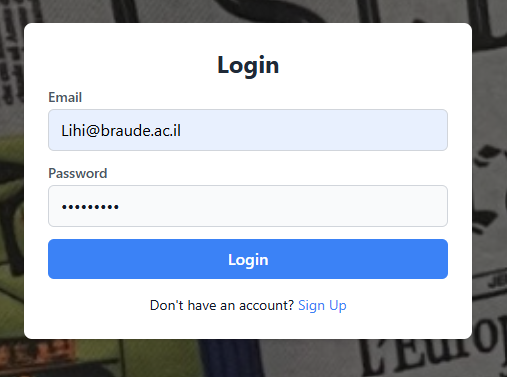
**פרומפטים שהשתמשתם בהם בכלי AI:**

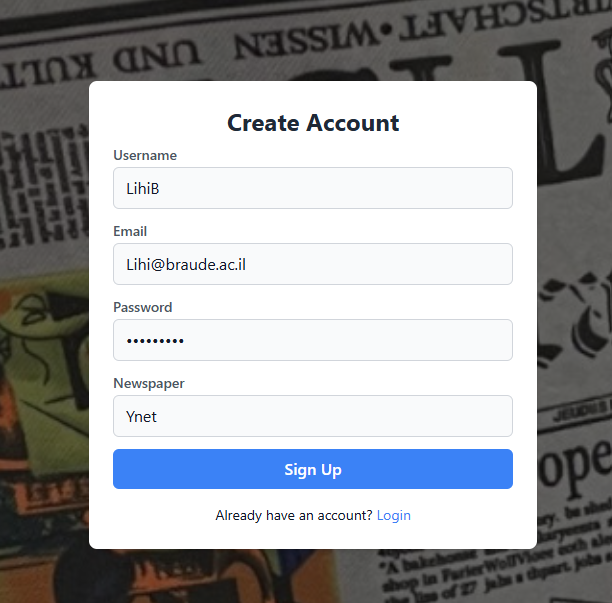
"Im building a website for interviewers. i want the website to be written in js react and be responsive. there needs to be a main page with tabs at the top - each tab transfers me to a different part of the website. the first tab is for the interview and questions- it should display possible questions for the interview based on interview topic (politics, sports etc). the questions come from connection to Google's Gemini. next to each question there should be a checkbox to choose the question- if the interviewer chose the question, an answer box needs to be displayed, making it possible for them to write the answer. an interviewer can also write their own questions. they also need to fill interviewee details (name, background) and intewrview date. the interview is saved to a firebase db by interviewer details, interviewee details and date. 2nd page- interview history. interviewer can see a table of all the interviews they made by interviewee name and date. when pressing a row in the table, the interview needs to be displayed"  
"Show navbar after login page only."  
"Build me a table using row and column separate components"  
"I want the AI questions to be selected using checkboxes. Each time the user clicks generate I want the option to select the new generated questions too"  
"Give me the option to add questions & answers by clicking on custom questions"  
"Please make the system more responsive for smaller screens"

"Please put the tabs inside the menu when website displayed in phone"  
"Give me the option to filter the table results by interviewee name, date or topic"  
"Please do the history page more user friendly and add instructions of how to upload word to the website"  
"Please separate the component from its backend functionality. If the method has a reference to the DB please put it in a different component"  
"Please sort the table by dates. The most recent interview should appear at the first row"  
"I want to add a menu (Hamburger) to the website with the users details and last interview that was conducted. I want it to open at the right side of the screen"

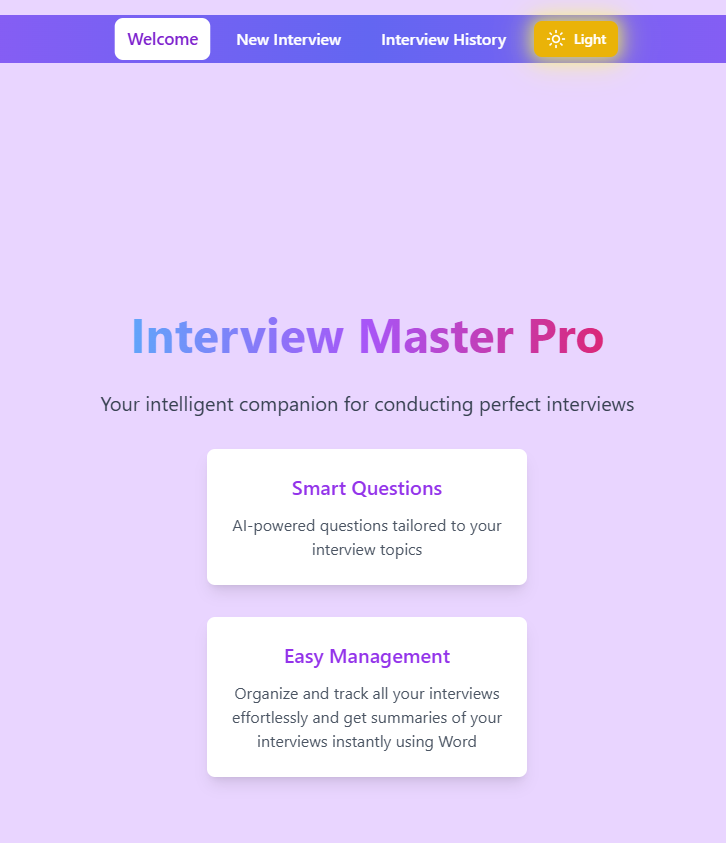
**תיק משתמש**

1. אם המראיין כבר רשום לאתר, יש לבצע כניסה ע"י מייל וסיסמה



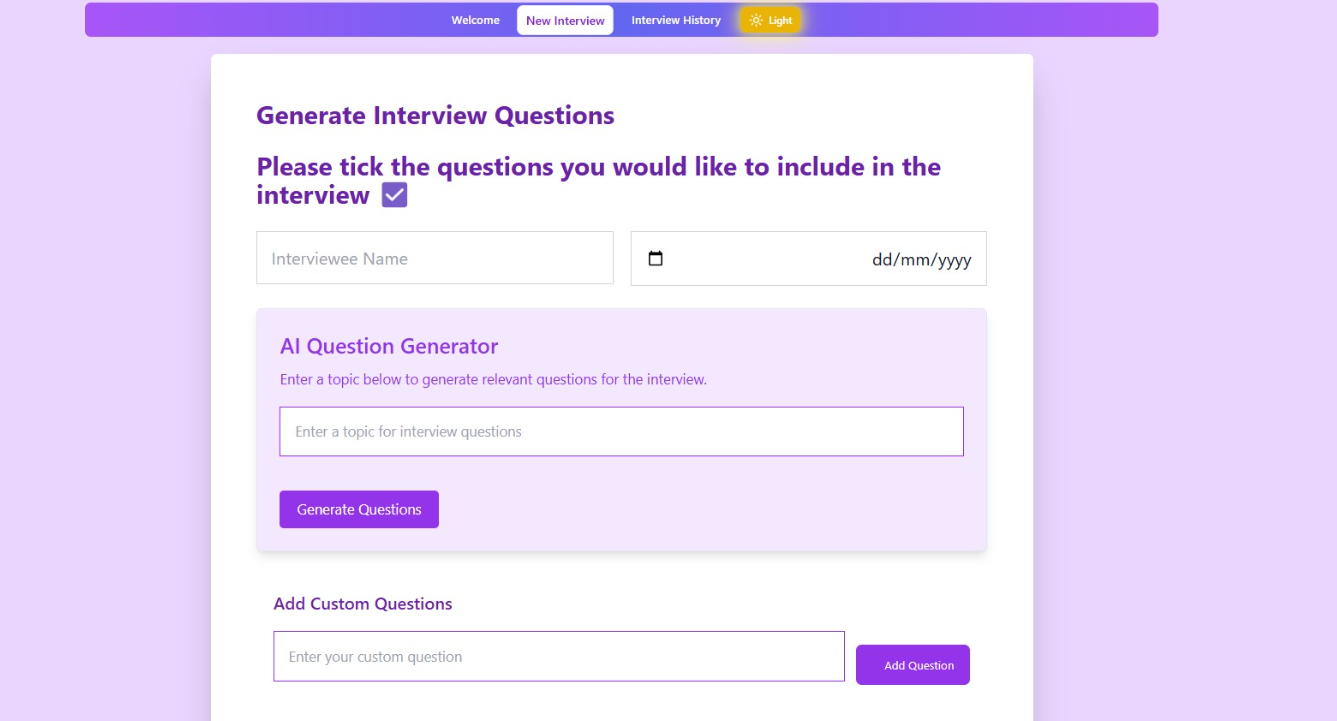
אחרת, אם המראיין נכנס פעם ראשונה לאתר המשתמש נדרש לבצע הרשמה   


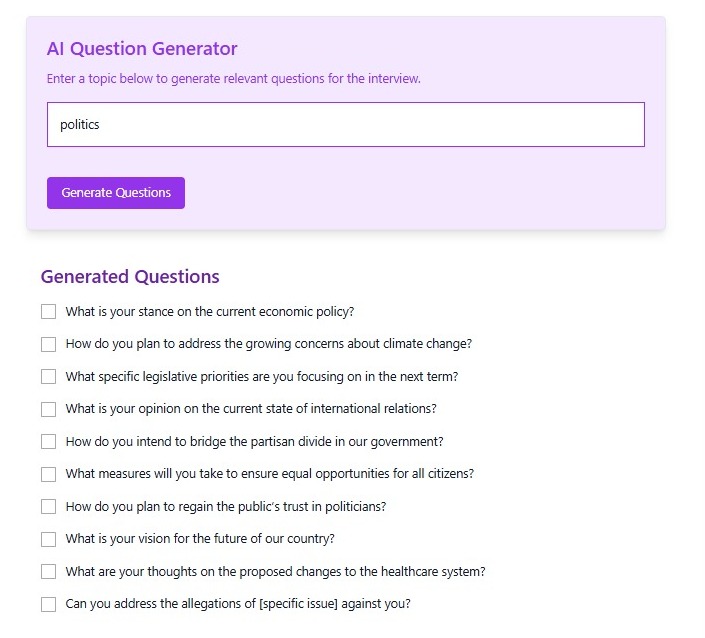
1. בדף הראשי המראיין יכול לבחור 2 אופציות:  
   1) Smart Questions - לבצע תיעוד ראיון באתר  
   2) Easy Management – ניהול כלל הראיונות

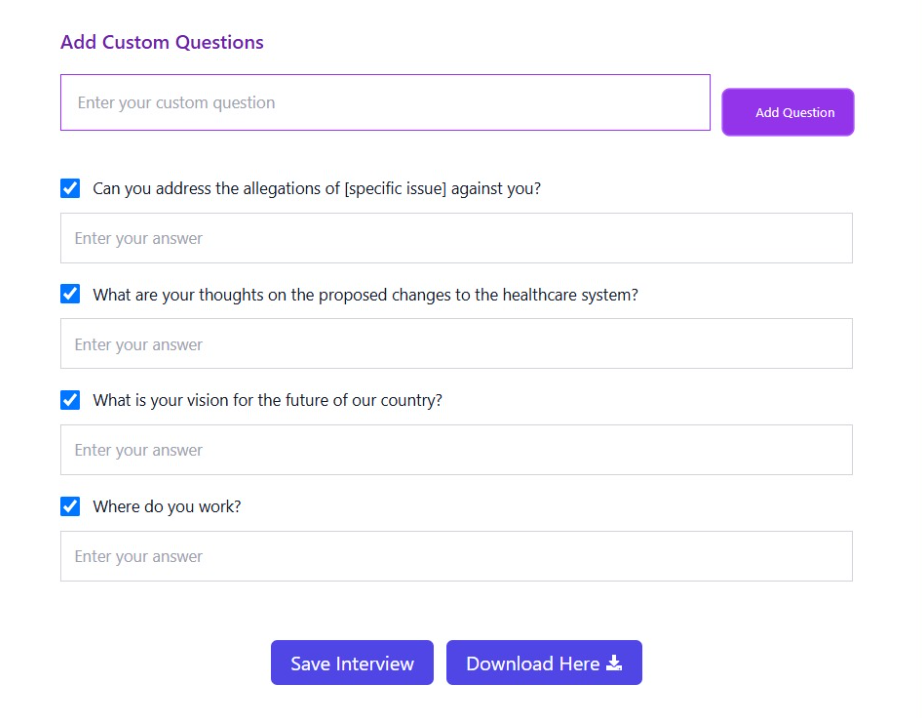


1. Smart Questions / New Interview (בסרגל כלים למעלה)

המשתמש צריך להכניס את שם המרואיין ותאריך קיום הראיון.  
לאחר מכן, המשתמש יכול לבקש לייצר שאלות דרך Gemini לפי נושא הראיון. בנוסף, יכול המראיין לכתוב שאלות שהוא מעוניין לשאול.

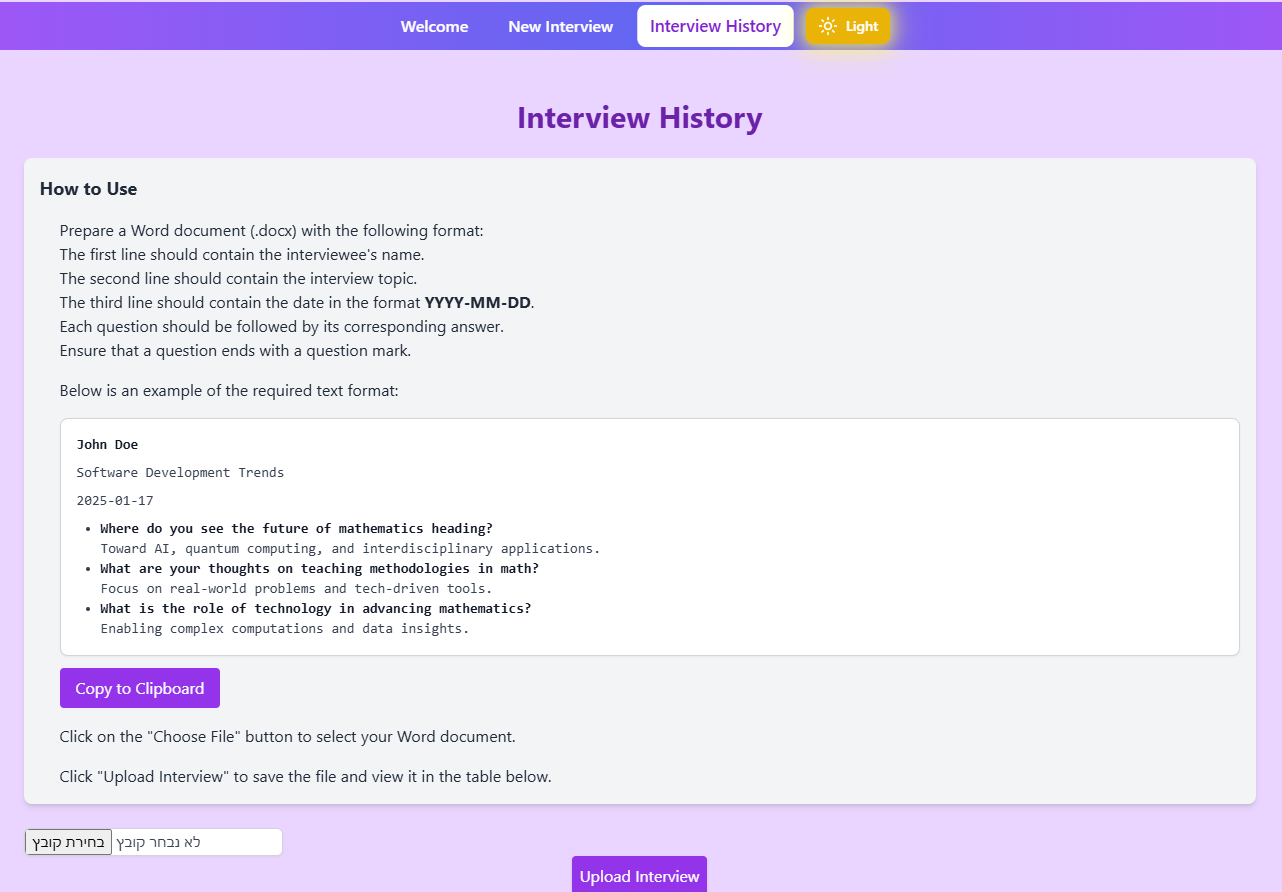


* אם המשתמש החליט לייצר שאלות דרך Gemini עליו להכניס את הנושא הראיון  
  
* לאחר בחירת השאלות, המראיין מזין את התשובות עליהם ולבסוף יכול להוריד את זה כקובץ וורד או לשמור באתר עצמו (במידה ולא מילא את כלל התיבות, זה ימנע ממנו להוריד/לשמור את הקובץ)

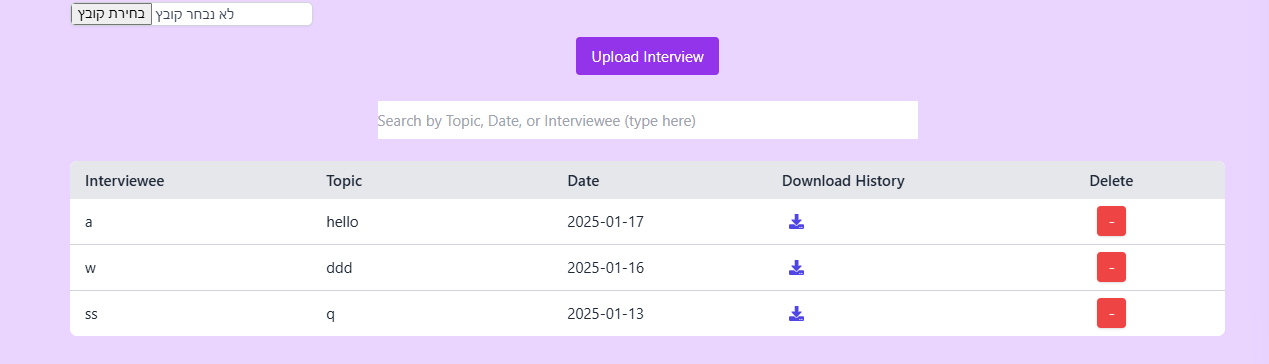


* אם המשתמש החליט שבנוסף/במקום הוא מעוניין לכתוב שאלות שהוא רוצה לשאול, הוא מקליד את השאלה בתיבת "Add Custom Question" ומזין תשובה לשאלה זו.

1. Interview History / Easy Management (בסרגל כלים למעלה)

המשתמש צריך להעתיק את מבנה הראיון הנדרש, כדי לשמור על אחידות בין כלל הראיונות (הראיונות שמיוצרים באתר והראיונות שהוא יכול להעלות כקובץ וורד).   
העלאת ראיונות חיצוניים תתבצע בלחיצת "בחירת קובץ" או “Choose File” (תלוי שפת המערכת של המשתמש) ולאחר מכן על "Upload Interview"

1. המשתמש יכול לראות את כלל הראיונות שהוא העלה/יצר באתר:



בנוסף, דרך לשונית החיפוש הוא יכול לבצע סינון למציאת ראיונות ספציפיים לפי שם המרואיין, תאריך הראיון או נושא הראיון

תמונה שמכילה צילום מסך, טקסט, קו

התיאור נוצר באופן אוטומטי

1. יש אפשרות להפוך את המסך לתצוגת לילה לפי לחצן "Light"   
     
   תמונה שמכילה טקסט, צילום מסך, כחול חשמלי, גופן

   התיאור נוצר באופן אוטומטי

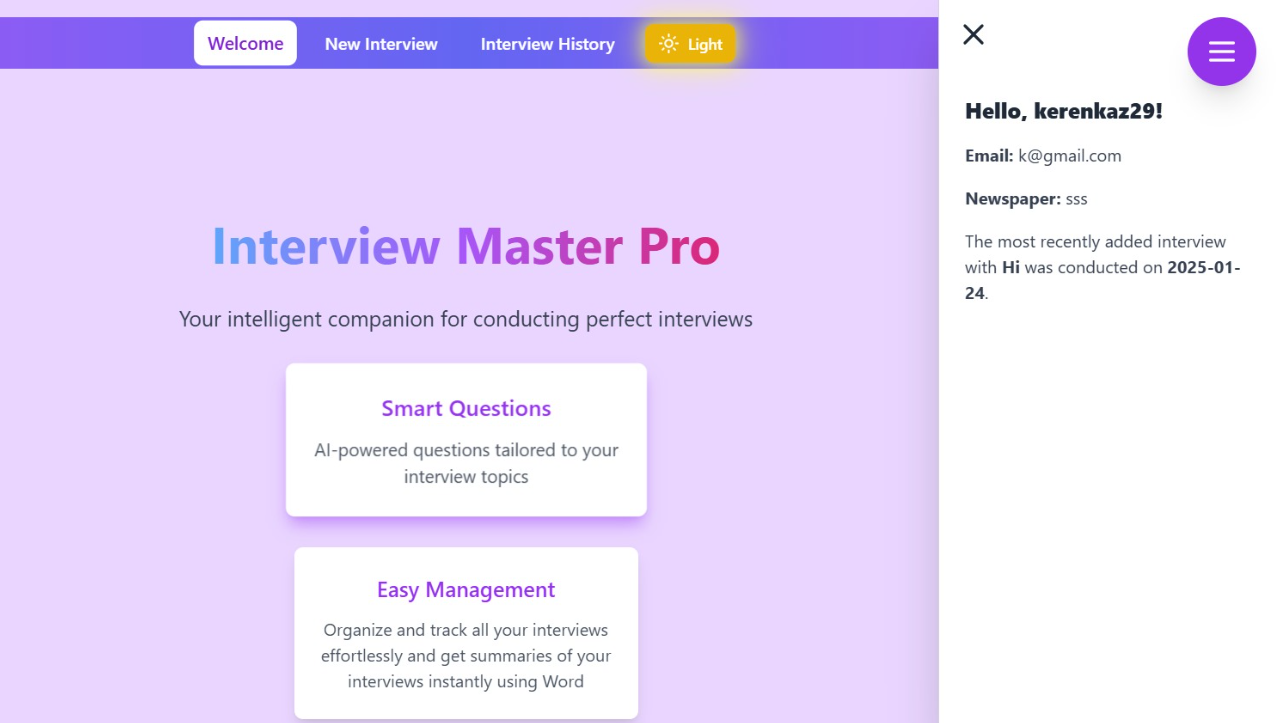
וההפך: יש אפשרות להפוך את המסך לתצוגת יום לפי לחצן "Dark"   
  
תמונה שמכילה טקסט, צילום מסך, גופן, כחול חשמלי

התיאור נוצר באופן אוטומטי

1. מבחינת רספונסיביות, האתר בעל פונקציונליות זהה בכלל המכשירים, מלבד סרגל הכלים שנמצא בצד למעלה במסך כתפריט נשלף.

באתר במחשב, בתפריט הנשלף, מופיעים רק פרטי המשתמש (שם משתמש, מייל, עיתון שבו הוא עובד) ופרטי הראיון האחרון שאותו מראיין עשה.

תצוגה במסכים גדולים:



תצוגה במסכים קטנים:  
תמונה שמכילה טקסט, צילום מסך, גופן, אתר

התיאור נוצר באופן אוטומטי