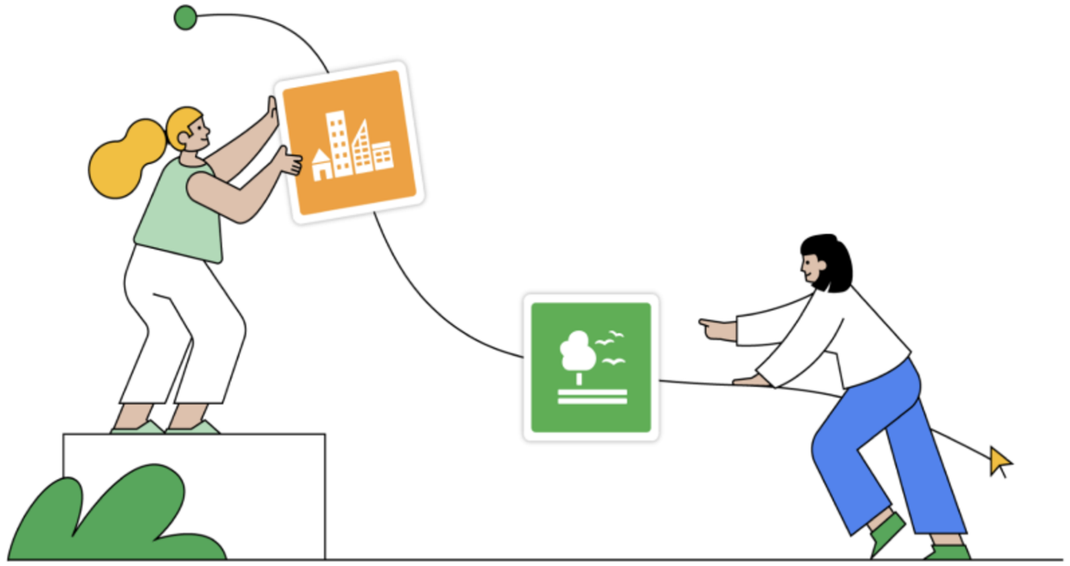
Powered by H2S

Solution Challenge



Team Details :

Team name : DataStreamer

Team leader name : Ayush Khaire

Problem Statement: Ensuring Web Accessibility for Visually Impaired Users Through Interactive Web technologies and AI-Powered Navigation.

Solution Challenge

Powered by H2S

Our solution is a Django-based web application designed to improve web accessibility for visually impaired users. It utilizes Selenium to capture and render complete webpages, ensuring that users can interact with them accurately. By implementing coordinate mapping, the system enables users to click on an image representation of a webpage, with precise synchronization to the actual site. The solution accounts for different screen sizes, making it adaptable across devices. sockets ensures smooth, real-time interactions, enhancing usability. This project is open-source, aiming to promote inclusivity and provide a more accessible browsing experience for people with disabilities . The project ensures some inovative features for using web with help of Gemini , and different Google technologies along with web .

Opportunities

How different is it from any of the other existing ideas?

Currently, accessing websites efficiently remains a challenge for visually impaired users. While screen readers and zoom features exist, they come with limitations—excessive zoom distorts website layouts, making navigation difficult, and screen readers read every word, even irrelevant details, increasing cognitive load. Our solution overcomes these issues by integrating AI, braille displays, and an ultra-zoom lens. Unlike traditional tools, it enables users to interact with entire webpages dynamically through coordinate-based clicks, ensuring better usability even for complex websites. This approach enhances accessibility by making web interactions more intuitive and efficient.

Opportunities

How will it solve the problem?

With the zoom lens, users can access websites in ultra-zoom mode without disrupting the webpage structure, making navigation smoother and reducing the effort required to swipe around. AI-powered features allow users to navigate efficiently with a single prompt, eliminating the need to manually search for buttons or elements. The integration of a braille display enhances accessibility for users with sensitive eyes by allowing them to read braille directly on the monitor, using color contrast instead of traditional touch-based methods—making reading faster than existing sound and vibration-based solutions. Additionally, for users who prefer sign-based interaction, the system provides a more intuitive way to view and type, significantly boosting productivity and ease of use.

Opportunities

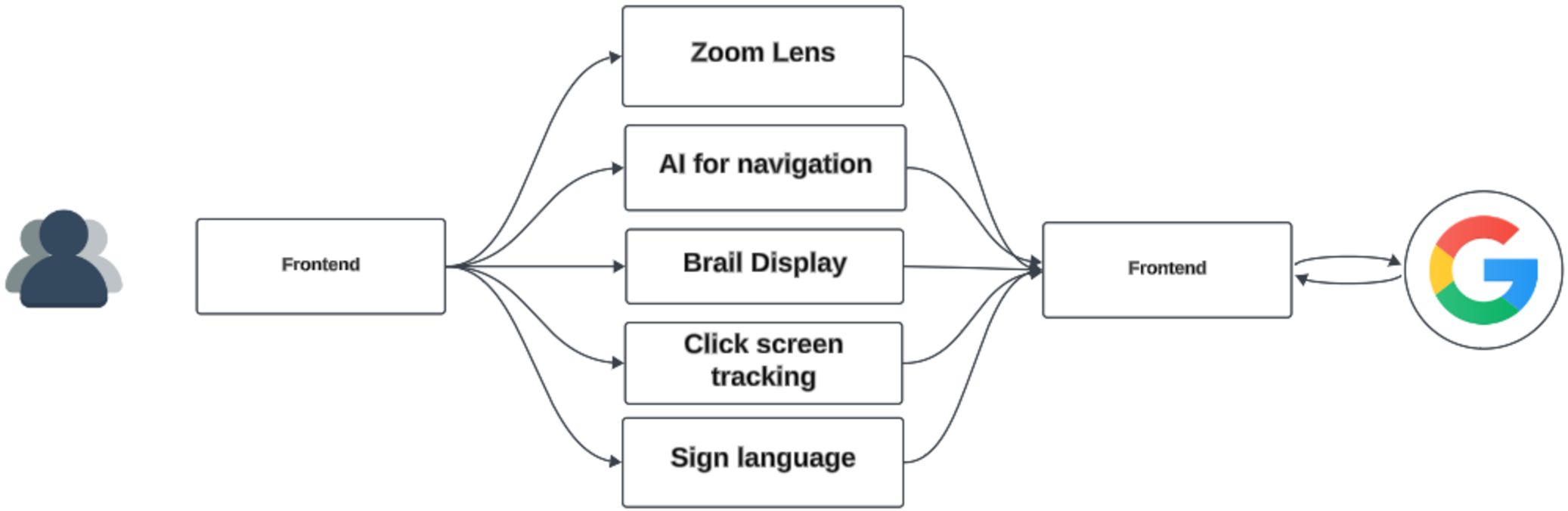
USP of the proposed solution

- Ultra Zoom Without Distortion: Unlike regular zooming, your solution maintains webpage structure while providing extreme magnification.
- AI-Powered Navigation: Users can find elements and navigate with a single prompt instead of manually searching.
- Braille Display on Screen: Instead of relying on touch-based braille devices, users can read braille directly on their monitor using color contrast.
- Multiple Accessibility Modes: Supports sign-based input, AI-assisted navigation, and braille, catering to different user needs.

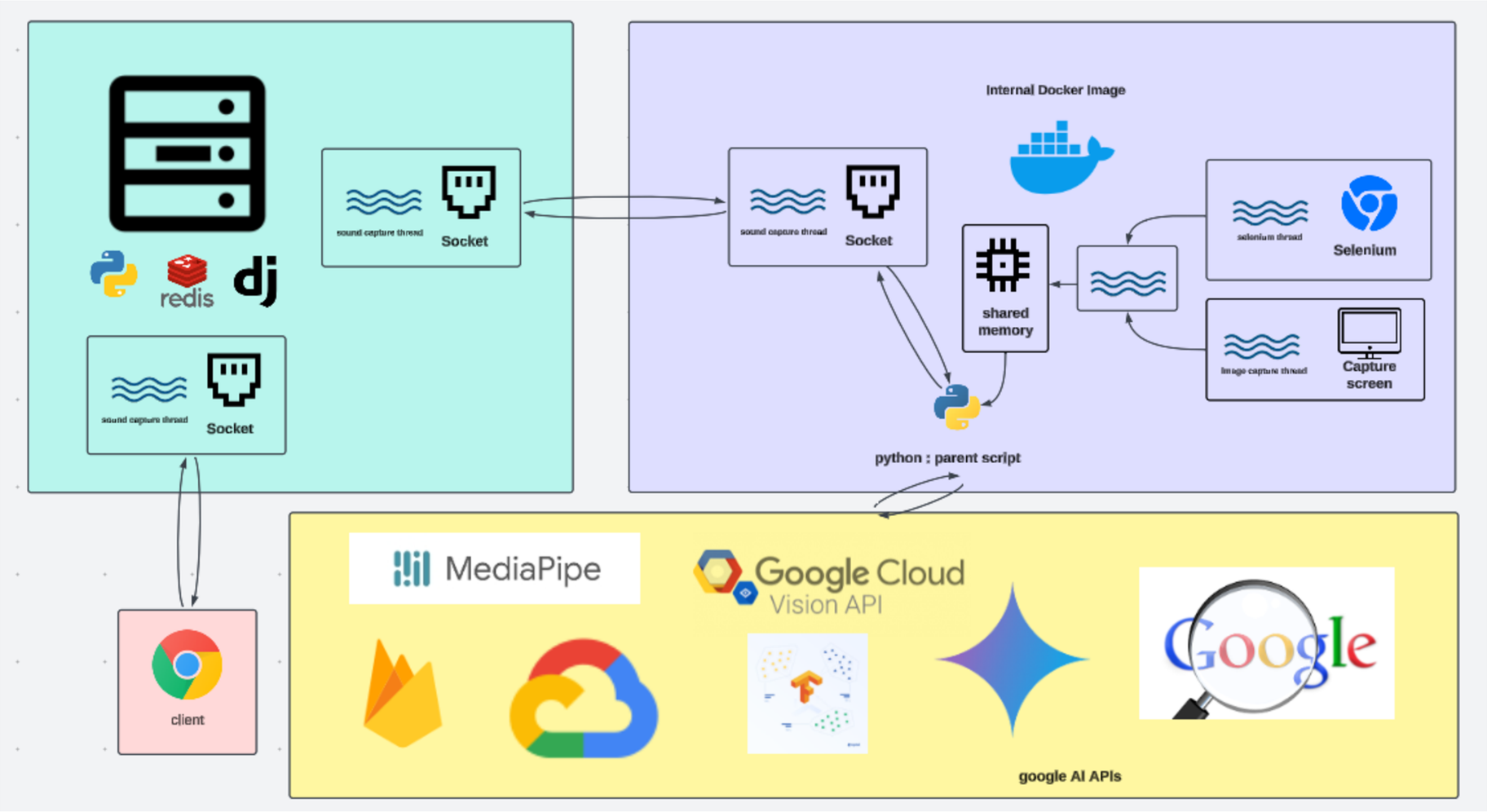
List of Features we offer

- **Real-Time Session & Stream** – Users can interact with web pages dynamically without delays.
- **Ultra Zoom for Web Pages** – Maintains webpage structure even with extreme zooming.
- **AI-Powered Navigation** – Allows users to navigate efficiently with minimal effort.
- **Braille Light Contrast-Based Display** – Enables visually impaired users to read braille directly on the screen.
- **User Activity Monitoring** – Learns user behavior to provide better assistance in future sessions.
- **Click Screen Data for Navigation** – Uses past interactions to create optimized navigation routes.
- **Sign Language Interpretation** – Helps users who prefer sign-based input and communication.

Process flow diagram



Architecture diagram



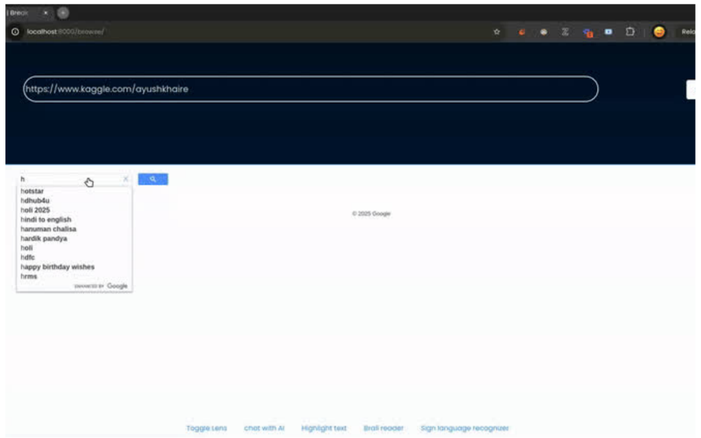
Technologies to be used

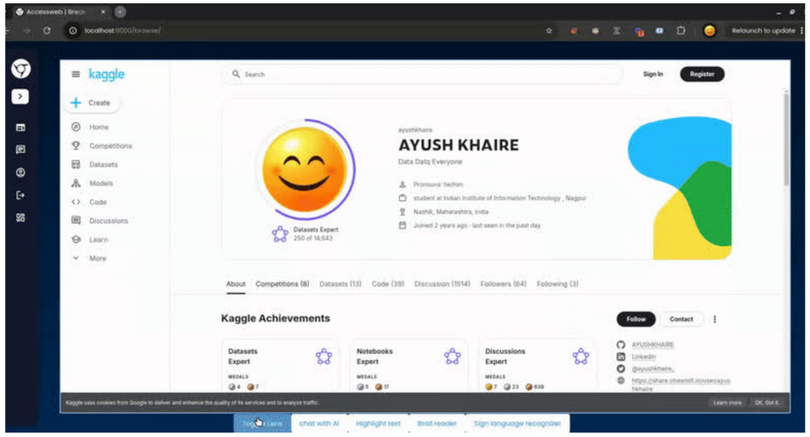
- Python
- Django
- HTTML + CSS + JS
- Websockets
- Docker
- Google Search
- Gemini APIs
- Vertex AI
- Google cloud
- Vision API
- TensorFlow
- Firebase + Redis
- Kubernetes (if needed)
- Kafka (if needed)

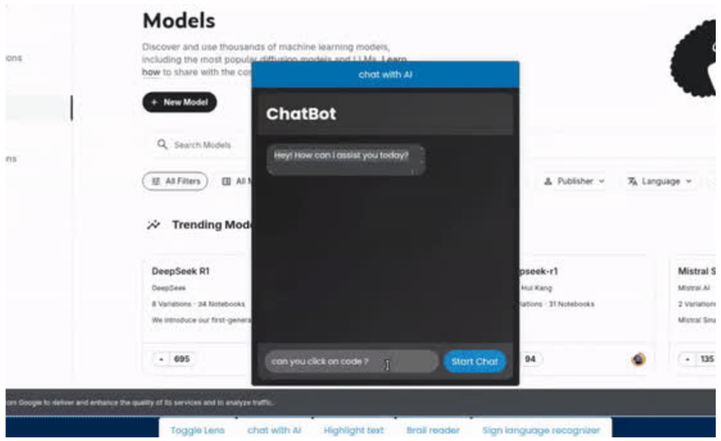
Estimated implementation cost

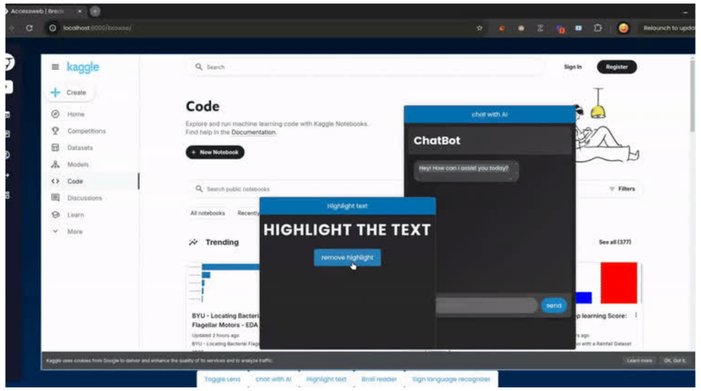
- To run our solution efficiently, we need a backend powered by Django, along with a server to handle requests. Firebase and Redis will manage real-time data and user sessions, ensuring smooth interactions. For AI processing, we leverage Vertex AI, Gemini APIs, and Vision API, optimizing accessibility features. Docker containers will be used for scaling, ensuring that the system can handle multiple users simultaneously. Additionally, Google Cloud and Kubernetes (if needed) will help in managing deployments efficiently. The architecture is designed to be cost-effective while ensuring high performance and accessibility. It de[ends completely on users - But while google is already managing a lot of services for free , this cost is resoanable .

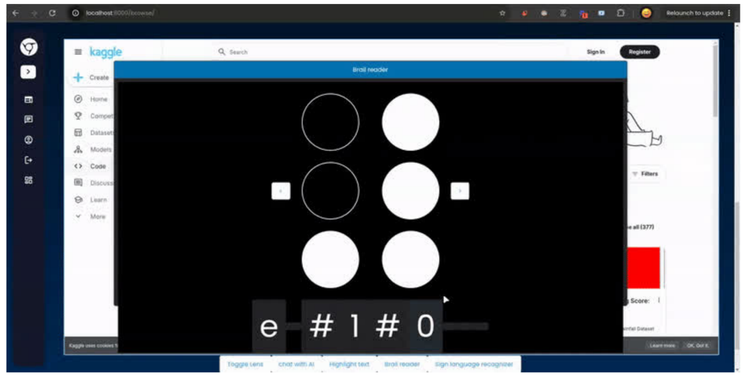
Snapshots of the MVP

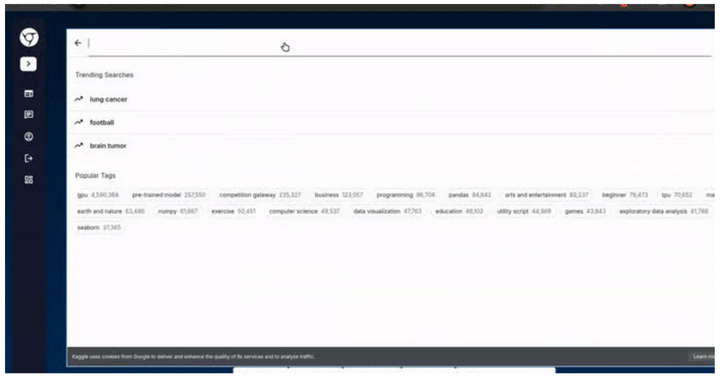








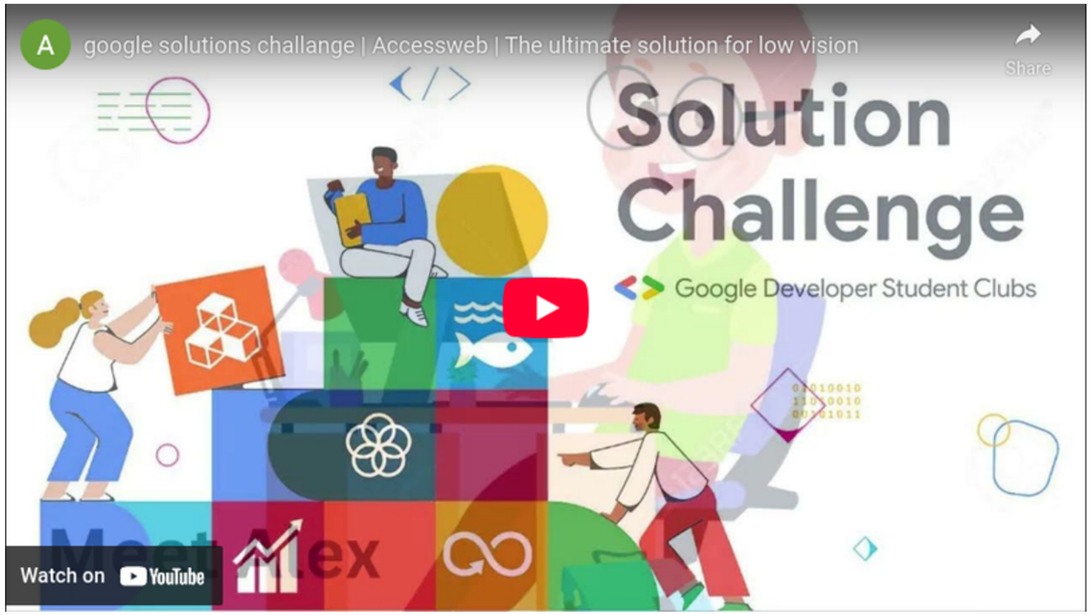




Solution Challenge

Powered by H2S

Explore more ……



View on github :
https://github.com/AYUSHKHAIRE/web-view
Send Feedback: or Feature requests :
ayushkhaire.dev@gmail.com

github.com/AYUSHKHAIRE/web-view

I apologize for not hhaving a proper deployment . This project is too much complex , please consider folowing youtube video and github repository as a MVP . Thanks !

Powered by H2S

Solution Challenge

Thank you