

UnitedGen

An Engineering Project in Community Service

Phase – II Report

Submitted by

21BCY10071 - PRANJALI SHRIVASTAVA
 21BCY10077 - SOUMYA NAIDU
 21BCE11563 - AARIDHI VISHWAKARMA
 21BAI10361 - KETAN SHARMA
 21BHI10070 - ESHITA KHARE
 21BSA10036 - R.MALLIKA
 21BSA10021 - AYUSH VERMA
 21BSA10160 - PRIYANSHI JUNEJA

in partial fulfillment of the requirements for the degree of

Bachelor of Engineering and Technology



**VIT Bhopal University
 Bhopal
 Madhya Pradesh**

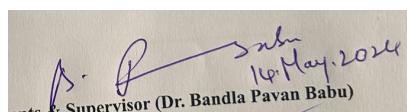
May, 2024



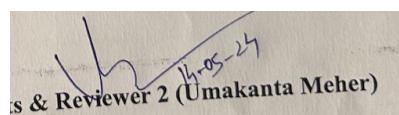
Bonafide Certificate

Certified that this project report titled “**UnitedGen**” is the bonafide work of “21BCY10071 Pranjali Shrivastava, 21BCY10077 Soumya Naidu, 21BCE11563 Aaridhi Vishwakarma, 21BAI10361 Ketan Sharma, 21BHI10070 Eshita Khare, 21BSA10036 R.Mallika, 21BSA10021 Ayush Verma, 21BSA10160 Priyanshi Juneja” who carried out the project work under my supervision.

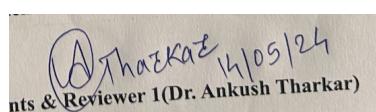
This project report (Phase II) is submitted for the Project Viva-Voce examination held on 10th May 2024.



Comments & Supervisor (Dr. Bandla Pavan Babu)



Comments & Reviewer 2 (Prof. Umakanta Meher)



Comments & Reviewer 1 (Dr. Ankush Tharkar)

CONTENTS

SN. NO.	TITLE	PAGE NO.
i.	Declaration of Originality	4
ii.	Acknowledgement	5
iii.	Abstract	6
iv.	List Of Figures	7
1.	Introduction	8-9
	1.1 Motivation	9
	1.2 Objective	9
2.	Existing work/ Literature Review	10
3.	Topic of Work	11
	3.1 System Architecture	11-16
	3.2 Working principle	17-25
	3.3 Expected results	26-31
	3.4 Code snippet	32-33
	3.5 Individual Contribution	34-37
4.	Conclusion	38
5.	References	39-41
6.	Biodata	42-43



Declaration of Originality

We, hereby declare that this report entitled UnitedGen represents our original work carried out for the EPICS project as a student of VIT Bhopal University and, to the best of our knowledge, it contains no material previously published or written by another person, nor any material presented for the award of any other degree or diploma of VIT Bhopal University or any other institution. Works of other authors cited in this report have been duly acknowledged under the section “References”.

Date: 10th May 2024

Reg No & Name: Ketan Sharma 21BAI10361

Acknowledgement

We extend our deepest gratitude to United Gen for their pioneering efforts in developing a web application aimed at bridging the gap between transgender, intersex individuals, and the citizens of India. Their commitment to fostering inclusivity and understanding within our society is truly commendable. Through their innovative platform, they are empowering communities and promoting meaningful dialogue, paving the way for a more equitable and accepting future for all.

Abstract

In contemporary societal discourse, intersex individuals often face misidentification and marginalization, frequently conflated with transgender persons. This research delves into the distinctive nature of intersex identities, where individuals embody a spectrum outside the conventional male/female sex binary. In contrast, transgender individuals experience a deviation between their gender identity and societal expectations associated with their assigned sex at birth. The intersex community in India, unfortunately, encounters derogatory labels and societal relegation, such as the terms "Chakke" or "Hijre."

This proposed initiative seeks to address the unique challenges faced by intersex and transgender individuals in India by creating a dedicated website. The platform aims to provide comprehensive information, support, and resources tailored to the needs of these communities. By bridging identities and fostering understanding, this initiative strives to empower individuals within the intersex and transgender communities, combating stigma and promoting inclusivity.

List of Figures

SN NO.	TITLE	PG NO.
1.	Architecture of Website Application	11
2.	Landing Screen	17
3.	Login and Sign Up Page	18
4.	User Authentication	20
5.	What to do Page	21
6.	To Learn Screen	21
7.	To Get a Job Screen	22
8.	Selection of Course Screen	22
9.	Attempting An Exam Screen	23
10.	Test Report Screen	24
11.	FrontEnd (Landing Page)	26
12.	FrontEnd (Problem Page)	27
13.	FrontEnd (Problems Form)	27
14.	FrontEnd (Problems Details)	28
15.	Table (Results)	32
16.	Code Snippet 1	33
17.	Code Snippet 2	34

1. INTRODUCTION

In contemporary society, there exists a common misconception that often conflates intersex individuals with transgender persons. It is crucial to recognize the distinctive nature of these identities. Intersex individuals embody a spectrum wherein their genitals, chromosomes, or reproductive organs do not conform to the conventional male/female sex binary. This incongruence may manifest as a mismatch between genitals and reproductive organs or the presence of both male and female traits. Regrettably, in certain societies, such as in India, individuals in the intersex community are sometimes derogatorily labeled as "chakke" or "hijre."

In contrast, transgender individuals navigate a different experience, where their gender identity deviates from the societal expectations associated with the sex assigned to them at birth. For example, a person assigned male at birth may identify and transition to a female gender identity.

It is crucial to note that individuals who undergo gender transitions (transgender individuals) are an integral part of the broader intersex community. Unfortunately, in India, many in the intersex community find themselves relegated to societal roles like begging or performing at events to sustain their livelihoods.

With this imperative in mind, our proposed initiative envisions the creation of a website tailored to the unique needs of these individuals. This website serves users by providing information and answering their queries to assist them.

1.1 Motivation

The motivation behind a project centered on the discrimination against intersex inferiority and unawareness, typically stems from a desire to address inequalities, promote social justice, amplify voices that are often marginalized, foster inclusivity, and create a safer, more supportive environment. It focus on working towards a more equitable society where everyone can thrive regardless of their gender identity.

1.2. Objective

The main objective of a project is to fill the gap between transgender/intersex and the world. Focusing on the transgender/intersex community could be multifaceted, aiming to promote inclusivity, raise awareness about their challenges, advocate for equal rights, provide support services, foster understanding, and create a more accepting society. The specific goal may vary based on the project's scope, whether it's educational, advocacy-driven, or aimed at providing tangible support.

2. Existing Work / Literature Review

"Queer AI" refers to the intersection of queer theory and artificial intelligence (AI). It involves examining how AI technologies intersect with issues of gender identity, sexual orientation, and queer experiences. Queer AI seeks to explore and address the ways in which AI systems can perpetuate or challenge existing biases, stereotypes, and inequalities related to LGBTQ+ individuals.

The Crisis Contact Simulator is a virtual tool designed to simulate and facilitate training for individuals who may find themselves in crisis intervention or support roles. It provides a realistic and immersive environment for users to practice their communication and de-escalation skills in various crisis scenarios.

The TelePrEP Navigator is a digital platform designed to facilitate access to pre-exposure prophylaxis (PrEP) for HIV prevention through telehealth services. It serves as a comprehensive tool to support individuals in navigating the process of initiating and adhering to PrEP medication regimens, particularly in areas where access to in-person healthcare services may be limited or challenging.

Angels Balaguers et al addresses that the RAG[2] pipeline consists of multiple stages, including extracting information from PDFs, generating questions and answers, using them for fine-tuning, and leveraging GPT-4 for evaluating the results. to the intersection of queer theory and artificial intelligence (AI). There are two common ways in which developers incorporate proprietary and domain-specific data when building applications of Large Language Models (LLMs): Retrieval-Augmented Generation (RAG) and Fine-Tuning. RAG augments the prompt with the external data, while fine-Tuning incorporates the additional knowledge into the model itself.

Aditi Singh (2024) et al introduces MindGuide[1] leverages the capabilities of LangChain and its ChatModels, specifically ChatOpenAI, as the bedrock of its reasoning engine. The system incorporates key features such as LangChain's ChatPrompt Template, HumanMessage Prompt Template, ConversationBufferMemory, and LLMChain, creating an advanced solution for early detection and comprehensive support within the field of mental health.

3. Topic of the work

(3.1) System Design / Architecture

Architecture of Website Application

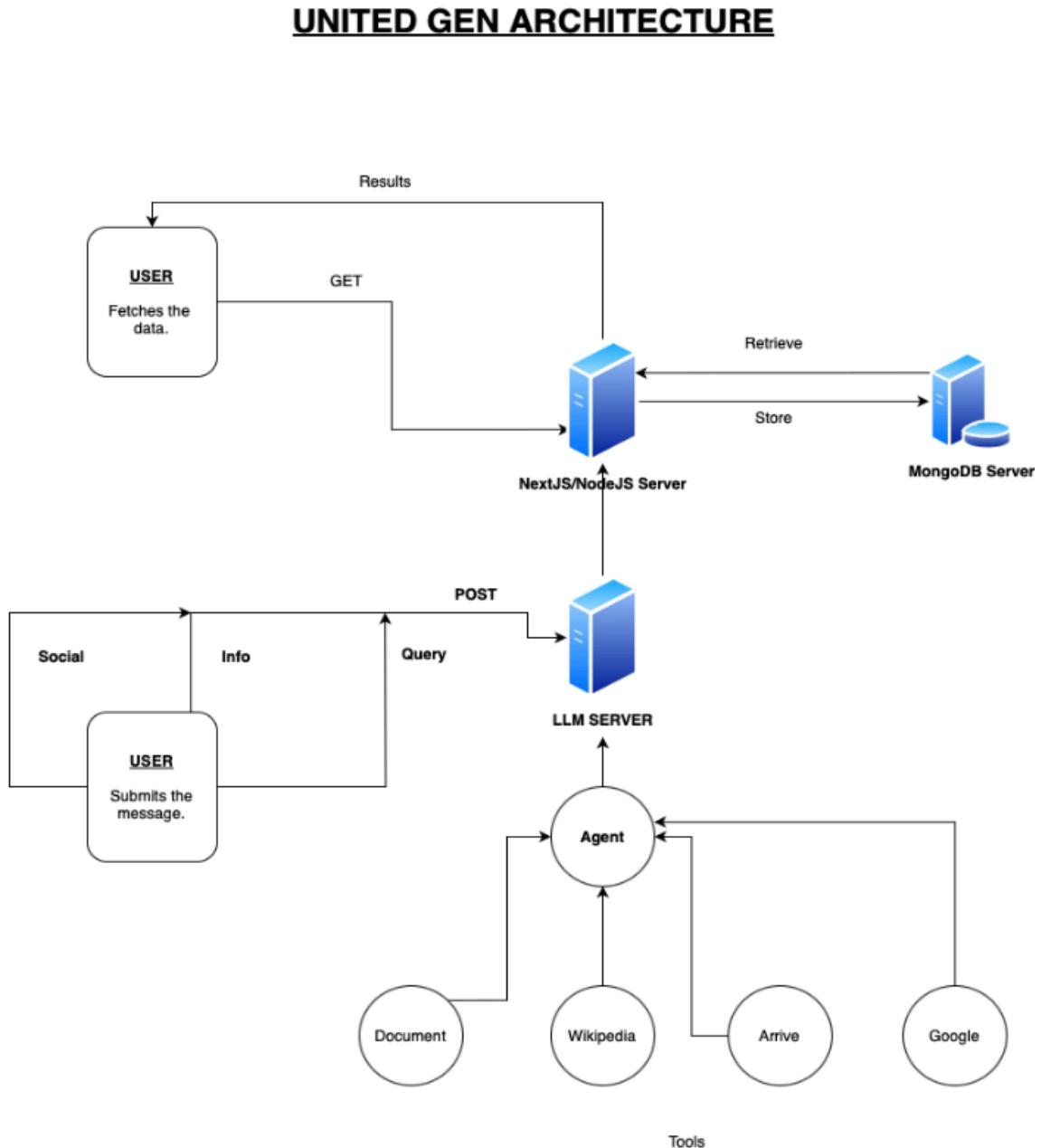


Fig. 3.1.1

Description: The figure above demonstrates how the working of our application would be done. The user would submit the data with (social, info or query) to the LLM server and then output from the LLM server would be sent to NodeJS server which stores and fetches data from mongodb server to user display.

Figma

Figma is a cloud-based design tool used for user interface (UI) and user experience (UX) design, prototyping, and collaboration. It's widely used by designers, developers, product managers, and other professionals involved in the design process. Here's a detailed explanation of how Figma works and its components:

1. Cloud-Based Platform: Figma operates entirely in the cloud, which means users don't need to download or install any software. They can access their projects from any device with an internet connection, making collaboration seamless.

2. User Interface: Figma's user interface is intuitive and user-friendly. It consists of various panels and tools organized to facilitate the design process. The main components of the user interface include:

- **Canvas:** This is where users create their designs. It's a blank space where they can add and arrange design elements.
- **Layers Panel:** This panel displays all the layers in the design, allowing users to organize and manipulate them easily.
- **Toolbar:** The toolbar contains various tools for creating shapes, text, images, and other design elements. Users can also find options for selecting, editing, and arranging objects.
- **Properties Panel:** This panel provides options to customize the properties of selected objects, such as fill color, stroke, opacity, etc.
- **Prototyping Panel:** Figma allows users to create interactive prototypes directly within the tool. The prototyping panel provides options for linking frames, adding interactions, and previewing the prototype.

3. Components: One of Figma's most powerful features is its component system. Components are reusable design elements that can be used across multiple screens or projects. When users edit a component, all instances of that component update automatically, ensuring consistency throughout the design. Components can also have multiple states, allowing for the creation of dynamic and interactive designs.

4. Collaboration: Collaboration is at the core of Figma's design philosophy. Multiple users can work on the same document simultaneously, seeing each other's changes in real-time. Figma provides tools for commenting, suggesting edits, and reviewing designs, making it easy for teams to communicate and collaborate effectively.

5. Version History: Figma automatically saves a version history of each design, allowing users to revert to previous versions if needed. This feature ensures that no work is ever lost and provides a way to track changes over time.

6. Plugins: Figma supports plugins that extend its functionality. Users can install plugins to automate repetitive tasks, integrate with other tools, or add new features to Figma's interface.

Overall, Figma is a versatile and powerful design tool that streamlines the entire design process, from ideation to prototyping to collaboration. Its cloud-based nature, intuitive interface, component system, and collaboration features make it a popular choice among designers and design teams worldwide.

Figma has been a great help in our project for the following reasons:

1. Wireframing

- Low-Fidelity Wireframes: We began by creating low-fidelity wireframes in Figma, using basic shapes and placeholders to outline the structure and content of each page.
- Grid Layout Design: Leveraging Figma's grid layout functionality, we structured wireframes with precision, aligning elements to a defined grid for visual consistency.
- Clickable Prototypes: Figma's prototyping features enabled us to link wireframes together, creating clickable prototypes that simulated user interactions and allowed for usability testing.

2. Designing the UI

- Vector Graphic Creation: Using Figma's vector drawing tools, we crafted custom icons and graphics that aligned with the project's visual identity and conveyed key messages effectively.
- Typography Selection: We curated typography styles within Figma, selecting fonts that balanced readability with visual appeal and ensuring consistency across the website.
- Color Palette Definition: Figma's color picker facilitated the creation of a cohesive color palette, with hues chosen to evoke specific emotions and reinforce brand identity.

3. Creating Components

- Component Libraries: We established component libraries within Figma, housing reusable UI elements such as buttons, forms, and navigation bars. These libraries facilitated rapid design iteration and consistency maintenance.

- Auto Layout Implementation: Leveraging Figma's Auto Layout feature, we created dynamic components that adjusted in size and layout based on content, optimizing responsiveness and reducing manual effort.
- Nested Components: By nesting components within one another, we built complex UI structures in Figma that remained editable and scalable, enhancing design flexibility and efficiency.

4. Prototyping

- Interactive Component Integration: Figma's interactive components allowed us to incorporate complex interactions, such as hover states and animated transitions, into our prototypes, enhancing realism and user engagement.
- Smart Animate Utilization: We utilized Figma's Smart Animate feature to create smooth transitions between screens, providing users with a seamless browsing experience and reinforcing the website's professionalism.
- Prototype Versioning: Figma's version history feature enabled us to track changes and revert to previous iterations of our prototypes, facilitating collaboration and mitigating the risk of design regressions.

5. Chatbot Interface

- Interactive Chatbot Simulation: Using Figma's prototyping capabilities, we simulated interactive chatbot conversations, allowing stakeholders to experience and provide feedback on the chatbot's functionality and user flow.
- Component States: We defined component states within Figma to represent different stages of the chatbot interaction, such as idle, typing, and response, ensuring visual clarity and enhancing the user experience.
- Voice User Interface (VUI) Design: Leveraging Figma's design tools, we crafted voice user interface elements for the chatbot, including speech bubbles and microphone icons, to communicate the conversational nature of the interaction.

The development process for the app:

1. Research and Planning

- We conducted comprehensive research to identify the diverse needs of the transgender and intersex community in accessing educational resources and support services.

- Through stakeholder interviews and user surveys, we gathered insights into the specific challenges faced by individuals seeking educational opportunities and support.
- Our planning phase involved mapping out the features and functionalities required to address these needs, including integration with educational platforms, provision of study materials, implementation of a chatbot for support, and the addition of features like tests and quizzes.

2. Designing the UI/UX

- We crafted an intuitive and visually appealing user interface using design tools such as Figma, ensuring that the app would be accessible and easy to navigate for users of all backgrounds.
- The user experience was carefully considered at every stage, with a focus on seamless navigation between educational resources, study materials, and support features.
- We incorporated feedback from user testing sessions to refine the design, ensuring that the app met the needs and preferences of its target audience.

3. Development

- Our development team utilized modern technologies and frameworks to build a robust and scalable app infrastructure.
- Integration with various educational platforms was achieved through API connections, allowing users to access a wide range of course materials and resources directly from the app.
- Features such as the chatbot were developed using natural language processing (NLP) algorithms to enable intelligent and context-aware interactions with users.
- Testing and quality assurance procedures were implemented throughout the development process to ensure a high level of reliability and performance.

4. Educational Platform Integration

- The app was designed to seamlessly integrate with popular educational platforms, such as online learning portals and digital libraries, to provide users with access to a wealth of educational resources.
- APIs were leveraged to establish connections with these platforms, enabling users to browse courses, access study materials, and engage with educational content without leaving the app.

- Personalization features were implemented to tailor recommendations and suggestions based on users' interests, preferences, and learning goals.

5. Study Materials and Resources

- A dedicated section within the app was created to curate study materials and resources relevant to the transgender and intersex community, covering a wide range of subjects and topics.
- Users could browse through textbooks, articles, videos, and other learning materials, with options to bookmark favorites and save progress for future reference.
- Features such as search functionality and content categorization were implemented to facilitate easy discovery and access to study materials based on users' interests and needs.

6. Chatbot Functionality

- The app included a chatbot feature designed to provide personalized support, guidance, and information to users.
- Utilizing natural language processing technology, the chatbot was able to understand user queries and provide relevant responses, whether it be answering questions, offering advice, or directing users to resources.
- Users could engage with the chatbot in real-time through text-based interactions, with the option to escalate to live support from human agents if needed.

7. Tests and Quizzes

- Interactive tests and quizzes were incorporated into the app to help users assess their knowledge, reinforce learning, and track progress.
- A variety of question formats, including multiple-choice, true/false, and fill-in-the-blank, were implemented to provide engaging and effective learning experiences.
- Users could receive instant feedback on their performance, view detailed explanations of correct answers, and track their scores over time to monitor their learning journey.

8. Feedback and Iteration

- Throughout the development process, we actively solicited feedback from users and stakeholders to identify areas for improvement and refinement.

- User testing sessions, surveys, and analytics data were used to gather insights into user behavior, preferences, and pain points, informing iterative updates and enhancements to the app.
- Continuous iteration and optimization were key principles guiding our development approach, ensuring that the app remained responsive to the evolving needs of its users and the broader community.

By following this approach, we were able to develop an app that not only provided access to educational resources and support services but also fostered a sense of community and empowerment among users.

App Prototype



Fig. 3.1.2 Landing Screen

Description: The figure above shows the screen which first appears when the user clicks on the app.

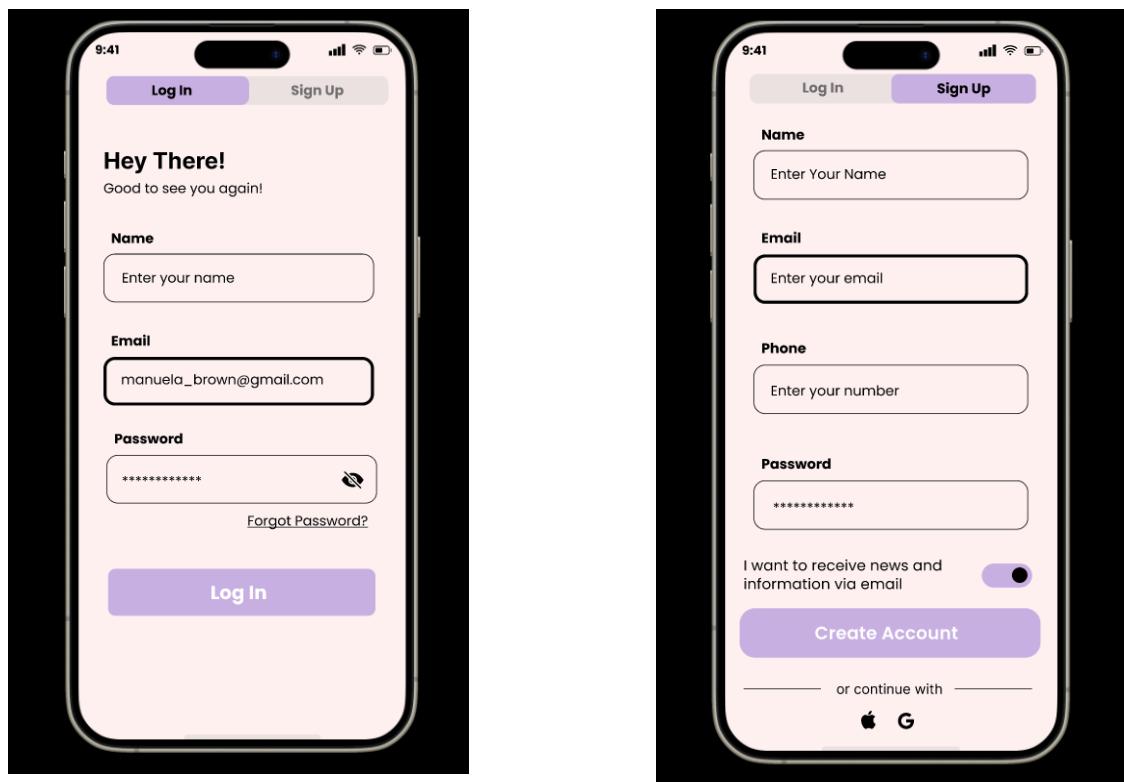


Fig. 3.1.3 Login and Signup Page

Description: This screen appears after the landing page. The users are given two options, i.e. Log in and Sign Up. The new users have the option of Sign up and create a new account and the existing users have the Log in option. The existing users only need to enter their name, email address and password for further use of the app.

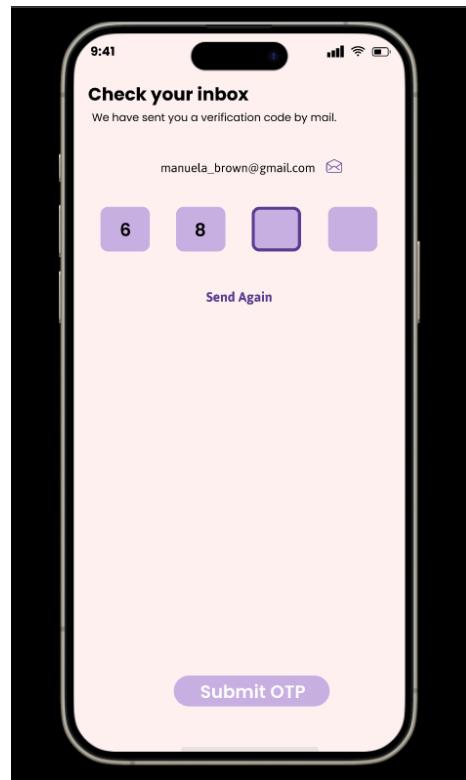


Fig. 3.1.4 User Authentication Page

Description: This screen appears after the user has created an account. This is a verification screen where OTP[One Time Password] is required for authentication of the user. This helps in maintaining the security of the app as well as the user's personal information, so that any unauthorized access is prevented.

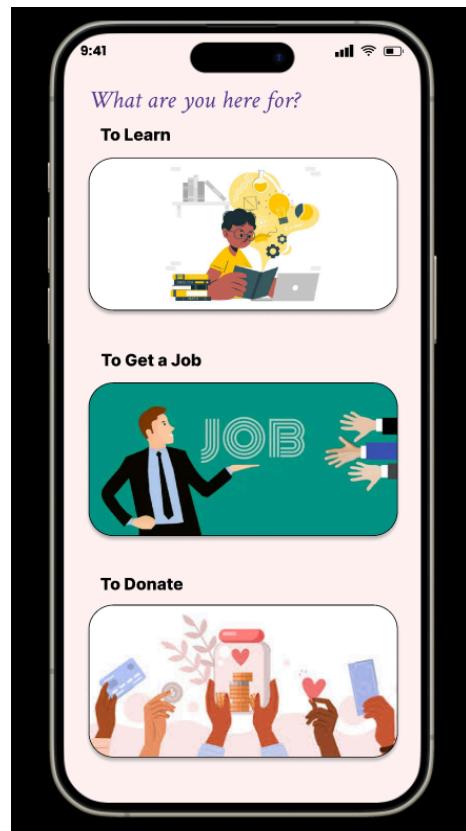


Fig. 3.1.5 To Choose what to do

Description: This screen appears after the Authentication has been successful . The user gets 3 options to choose what they want to do using the app, i.e., learn, get a job or donate for the community.

The image displays four screenshots of a mobile application interface:

- Choose subject:** A list of subjects with radio buttons: Chemistry, Math, English, and Biology. A purple 'Let's Go!' button is at the bottom.
- Course deets:** Details about a course titled 'Intermediate Full Class' (12 hours, 6 months). It shows a profile picture of a man, a 'Preview' button, and a 'Get Started' button.
- Playlists:** A list of playlists under the 'Maths' category. It includes 'Algebra Basic' (5 topics: 1. Algebraic Basic, 2. BOOMAS, 3. Multiplication, 4. Substitution, 5. Solving Inequalities), 'Geometry', 'Multiplication', 'Substitution', and 'Solving Inequalities'. Each topic has a preview button and a like/share icon.
- Overview:** An overview of the 'Maths' category by 'Jenny Francis, Math Teacher'. It shows a profile picture, a 'Preview' button, and a 'Topics' section listing: Exponent, Expression, Polynomials, Like and Unlike Terms, Constants, and Variables.

Fig. 3.1.6 To Learn

Description: The mobile app interface allows users to choose subjects like Chemistry, Math, English, and Biology. They can enroll in specific courses, such as the “Intermediate Full Class” or “Part Time Learning” and access detailed information about the topics covered. They can also choose their mentors. There are personalized playlists under each subject. Users can preview and like this playlist. They can also get an overview of the course and mentor beforehand.

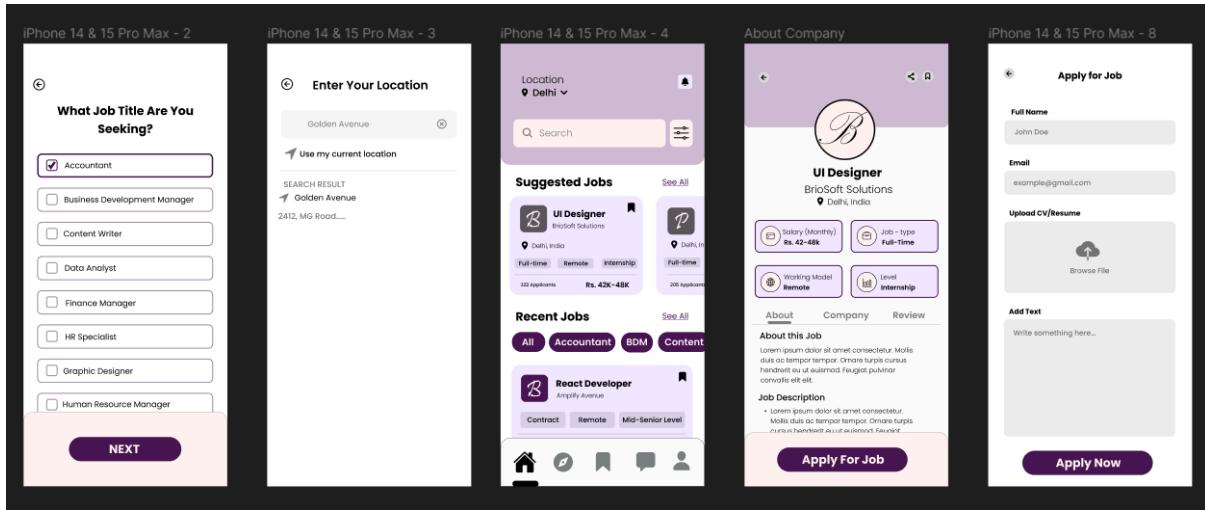


Fig. 3.1.7 To Get a Job

Description: The users first choose a job title from many options like Accountant, Business Development Manager, Content Writer, etc., and proceed with the "NEXT" button. Then, they enter their location manually or use their current location on a map. The app presents suggested and recent jobs along with the levels and timeline the user wants, such as UI Designer (full-time/remote/part-time), React Developer (intern level/intermediate/experienced), etc. Specific job details are provided, like salary, job type, and location for each position, for example, UI Designer at BrightSoft Solutions in Delhi. Users fill out a form with their name, email, and optionally upload a resume, along with additional text, and submit via the "Apply Now" button.

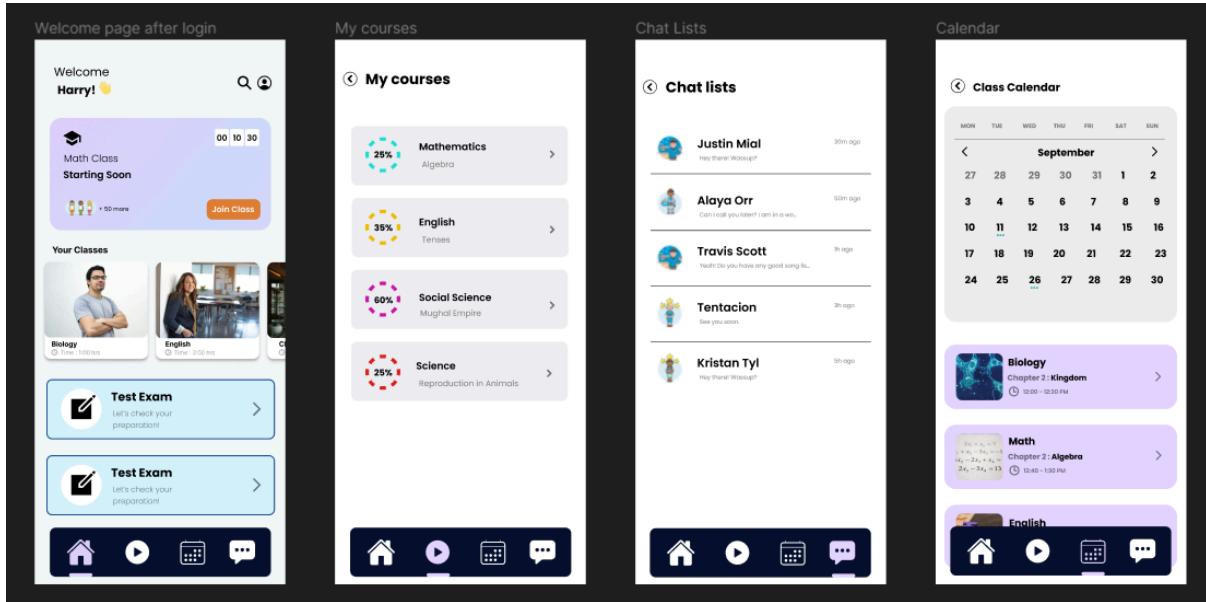


Fig. 3.1.8 Selection of course and class schedule

Description: The user is greeted upon logging in. At the top, there's a search bar. The upcoming scheduled class is displayed. There is also an option for taking an exam. In the my courses page list of all the courses appear. If we navigate to the chat lists all the conversations get loaded and new chats can be read. When we click on the calendar option all the classes scheduled throughout the month are displayed on a calendar.

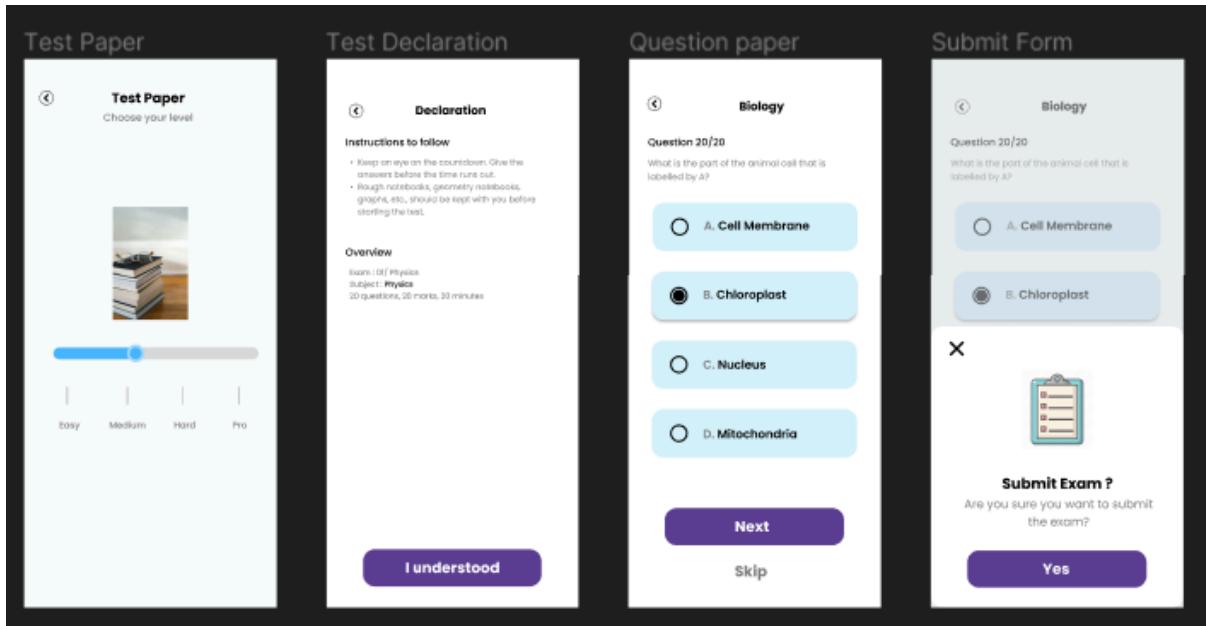


Fig. 3.1.9 Attempting an exam

Description: The user can attempt a test paper to check their preparation. A slider allows the user to choose their level of difficulty, ranging from easy to pro. Test Declaration contains instructions and an overview for the test. A question paper is displayed with multiple options. Click on buttons labeled "Next" and "Skip" after choosing the answer. A confirmation message asks if the user is sure they want to submit their exam.

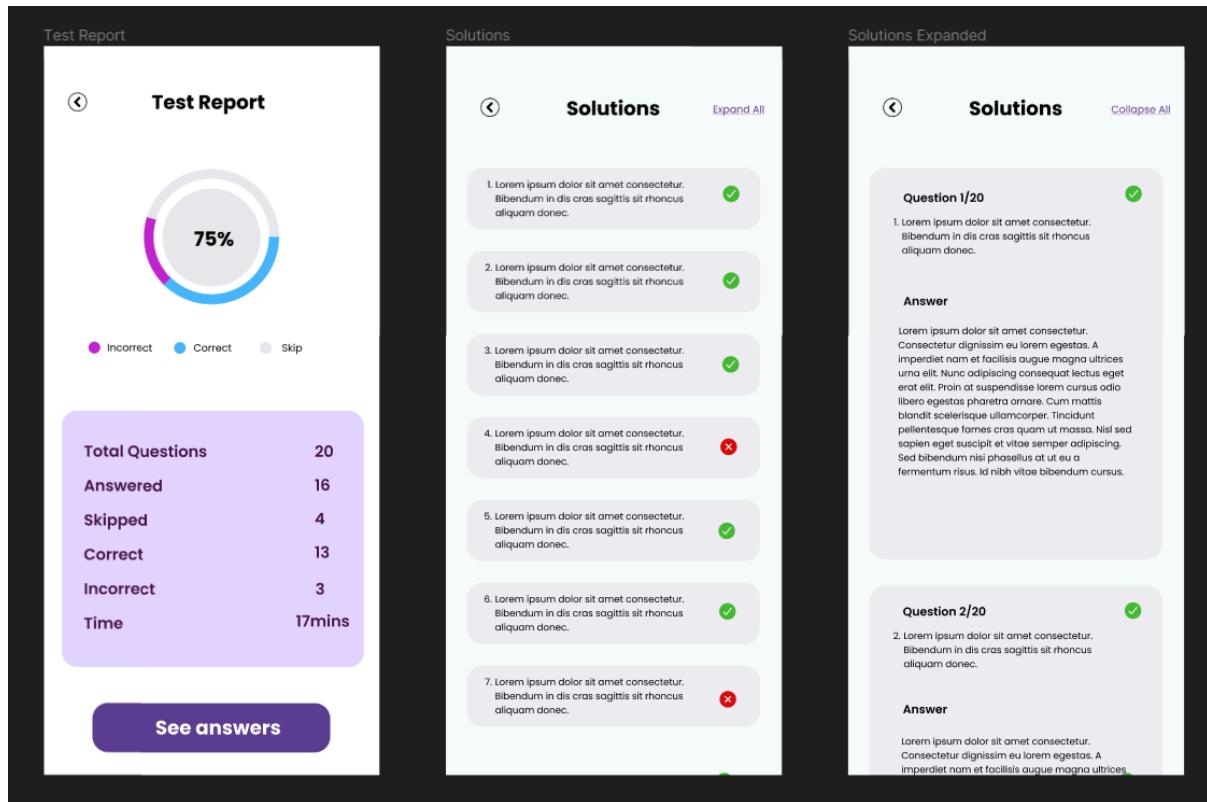


Fig. 3.1.10 Test Report Section

Description: This screen appears after the user has successfully completed the test. The users Full Test report details :- No of questions attempted; not answered; answered; correct; incorrect and the time taken by them. It also shows the solutions after the completion of the test so that the users can analyze where they went wrong and prepare better next time.

(3.2) Working Principle

The working principle of the app involves a multi-faceted approach to gather and process information.

- (i) Input Gathering: Users provide input in three different forms:
 - Queries: Direct questions or search terms.
 - Information: Data or details they wish to share.
 - Social Experiences: Insights or experiences shared by other users.
- (ii) Message Compilation: These inputs, along with any accompanying messages, are compiled into a cohesive format.
- (iii) Utilization of Language Model: The compiled message is then fed into a Language Model (LLM) for processing. This LLM likely performs tasks such as natural language understanding, contextual analysis, and generating appropriate responses.
- (iv) Data Sourcing: The app utilizes OLAMA (Open-source Language Model Aggregator), which integrates various sources of information including:
 - Wikipedia: A vast repository of knowledge on diverse topics.
 - Google Search: Provides real-time and comprehensive search results.
 - Reddit: Offers user-generated content and discussions on a wide range of subjects.
- (v) Information Retrieval and Analysis: OLAMA fetches relevant data from these sources based on the input message. It then analyzes this data to extract meaningful insights and up-to-date information.
- (vi) Response Generation: The processed information is used to generate a response to the user's query or input.

Website Design

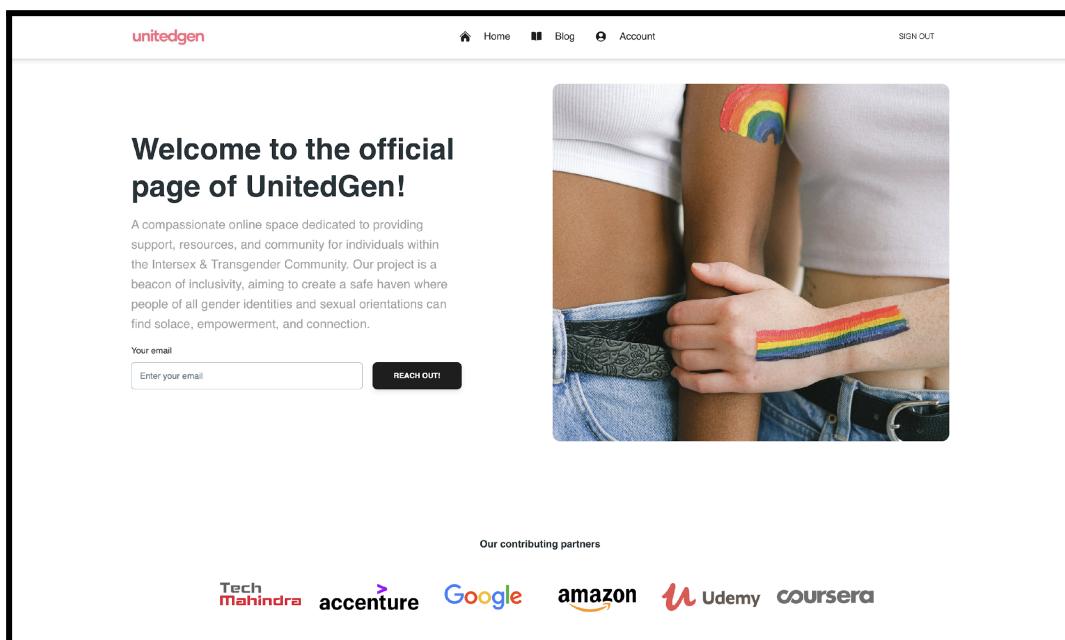


Fig. 3.2.1 FrontEnd (Landing Page)

Description: The figure above shows how the landing page of our website would look like. As it can be clearly seen the theme applied corresponds to LGBTQ section and also anyone can reach out to us by simply giving their email. We have also provided our contributing partners. The navbar above has three navigations and o-auth functionality has also been provided for the security and privacy of the user.

Fig. 3.2.2 FrontEnd (Problems Section)

Description: The figure above shows how the problems will be displayed which were sent by the user to us via problem form. We can see the author name, problem type and message that has been sent by the user to us. By clicking the read more button we can get more information about the problem. This screen will only be visible to the Employees handling the backend of our Website

The screenshot shows a web page titled "Send Us Your Problem" under the "unitedgen" header. The page includes a sidebar with options like "Post Problems", "Apply Job Opportunities", "See Problems", and "Log Out". The main form area contains fields for "Your Name" (with placeholder "First Name Second Name"), "Your Email" (with placeholder "name@mail.com"), "Type of Problem" (a dropdown menu showing "Select Problem Type"), "Title of Problem" (with placeholder "Overview"), and a large "Your Message" text area. A "SEND" button is at the bottom, and a note below it says "We will respond you within 24 hours."

Fig. 3.2.3 FrontEnd (Problems Form)

Description: The figure above shows the User Interface That will be shown on the User's screen where the Users through this section can share their problems with us.. We can see the author name, problem type, email, title and message that has been sent by the user to us. By clicking the send button the user can send their message to the LLM server working in the backend.

Problem Details	
Title	Transgender Rights
Name of sender	Rohit Sharma
Email of sender	rohit.sharma2021@gmail.com
Type	Information
Description	Tell me about transgender rights.
Solution	<p>Transgender Rights in India: India has a long history of recognizing the third gender, with many ancient texts and art forms referring to individuals who do not identify as male or female. However, despite this recognition, transgender people in India continue to face significant discrimination and marginalization. Here are some key issues related to transgender rights in India:</p> <ol style="list-style-type: none"> 1. Legal Recognition: The Transgender Persons (Protection of Rights) Act, 2019 is the first-ever law in India that recognizes transgender people as a distinct gender category. However, the law has faced criticism for being too narrow and not addressing the diverse needs of the transgender community. 2. Identity Documents: Transgender people in India often face difficulties when trying to obtain identity documents such as Aadhaar cards, passports, and driving licenses. These documents typically require gender markers that do not accurately reflect a person's gender identity. 3. Employment: Transgender people in India often face discrimination in the workplace, including unequal pay and limited job opportunities. Many transgender people are forced to work in marginalized industries such as sex work or begging. 4. Healthcare: Transgender people in India face significant barriers when seeking healthcare services, including lack of access to gender-sensitive healthcare facilities and discrimination by healthcare providers. 5. Education: Transgender children are often denied admission to schools or face harassment and bullying once they are enrolled. Many transgender people in India are illiterate or have limited access to education. 6. Housing: Transgender people in India often lack access to affordable and secure housing, leading many to live in slums or on the streets. 7. Police Brutality: Transgender people in India are at a high risk of police brutality, including harassment, violence, and arrest under laws that criminalize their identity. 8. Social Stigma: Transgender people in India face significant social stigma and discrimination, which can lead to mental health issues, substance abuse, and suicide. 9. Marriage and Family: Transgender people in India often face rejection from their families and communities when they come out as transgender. Many transgender people are forced into marriages with individuals of the opposite gender, which can lead to further marginalization and violence. 10. Lack of Data: There is a lack of comprehensive data on the transgender population in India, making it difficult to address their specific needs and challenges.

Fig. 3.2.4 FrontEnd (Problems Details)

Description: The figure above shows the detailed view of the problem when the user clicks the read more button on the problem card on the problem page. We can see the title senders' name, email, type description and the solution to the problem given by the LLM to the user.

Next.js

Next.js is a popular React framework that's primarily used for building server-side rendered (SSR) and statically generated (SSG) web applications. It's known for its simplicity, performance, and developer experience. Here are some key features and concepts:

1. Server-side rendering (SSR): Next.js allows you to render React components on the server side before sending them to the client, which improves initial load times and SEO. This is especially useful for content-heavy websites or applications.
2. Static site generation (SSG): Next.js can generate static HTML files at build time, which can then be served directly from a CDN. This approach is great for websites with relatively stable content that doesn't need to be generated dynamically on each request.
3. File-based routing: Next.js uses a file-based routing system, where each React component in the pages directory automatically corresponds to a route in the application. This simplifies the routing setup and makes it intuitive to organize your code.
4. API routes: Next.js allows you to create API routes alongside your pages, making it easy to build backend functionality within the same project. These API routes can handle requests like any other server-side code.
5. Built-in CSS and Sass support: Next.js provides built-in support for importing CSS and Sass files, allowing you to style your components with ease. It also supports CSS modules for scoped styles.
6. Image optimization: Next.js optimizes images automatically, reducing their size and improving performance without sacrificing quality. This is achieved through various techniques like lazy loading and responsive image support.
7. TypeScript support: Next.js has first-class support for TypeScript, enabling you to write type-safe React code and catch errors early in the development process.
8. Incremental static regeneration (ISR): This feature allows you to update static content without having to rebuild the entire site. It's useful for content that needs to be updated frequently but doesn't change on every request.
9. Built-in performance optimization: Next.js comes with several performance optimization features out of the box, such as automatic code splitting, prefetching, and client-side routing, to ensure fast page loads and smooth user experiences.
10. Automatic code splitting: Next.js automatically splits your code into smaller chunks, so only the necessary JavaScript is loaded for each page. This helps reduce initial load times and improve performance, especially on slower network connections.

Overall, Next.js provides a powerful and flexible framework for building modern web applications with React, offering a range of features to enhance developer productivity and user experience.

Backend (LangChain/RAG Server)

One of the most powerful applications enabled by LLMs is sophisticated question-answering (Q&A) chatbots. These are applications that can answer questions about specific source information. These applications use a technique known as Retrieval Augmented Generation, or RAG.

RAG is a technique for augmenting LLM knowledge with additional data.

LLMs can reason about wide-ranging topics, but their knowledge is limited to the public data up to a specific point in time that they were trained on. If you want to build AI applications that can reason about private data or data introduced after a model's cutoff date, you need to augment the knowledge of the model with the specific information it needs. The process of bringing the appropriate information and inserting it into the model prompt is known as Retrieval Augmented Generation (RAG).

LangChain has a number of components designed to help build Q&A applications, and RAG applications more generally.

The RAG (Retrieve, Answer, Generate) pipeline architecture is a framework used in natural language processing (NLP) and question-answering systems. It's particularly effective for tasks like open-domain question answering, where the system must understand and respond to questions without relying on a predefined database or knowledge graph.

Here's a breakdown of each step in the RAG pipeline:

1. Retrieve: In this step, the system retrieves relevant documents or passages from a large corpus of text based on the input query or question. Various retrieval techniques can be used, including keyword matching, semantic search, or neural retrieval models like BM25 or dense vector retrievers.
2. Answer: Once the relevant documents or passages are retrieved, the system identifies potential answers within these texts. This step involves understanding the context of the question and the content of the retrieved documents to extract relevant information. Techniques such as named entity recognition (NER), semantic parsing, or machine reading comprehension (MRC) are often employed to identify and extract answers.
3. Generate: In the final step, the system generates a coherent and concise answer based on the information extracted in the previous steps. This could involve paraphrasing or summarizing the extracted information to form a natural language response that directly addresses the input query.

The RAG pipeline architecture allows for modularization and specialization of different components, making it flexible and adaptable to various NLP tasks and

domains. It enables efficient processing of large amounts of text data and facilitates accurate and informative responses to user queries. Additionally, each step in the pipeline can be optimized independently, allowing for continuous improvement of the overall system's performance.

LLM (Ollama/LLama2)

"Llama2 Large Language Model" (LLM) could potentially refer to an advanced version or iteration of a language model inspired by llamas. In the realm of artificial intelligence and natural language processing (NLP), a large language model is a type of machine learning model trained on vast amounts of text data to understand and generate human-like text.

Here's what "Llama2 Large Language Model" could entail:

1. Training Data: The Llama2 LLM would be trained on extensive datasets comprising diverse text sources, including books, articles, websites, and other written materials. These datasets would likely include information about llamas and related topics, allowing the model to understand context-specific nuances and generate relevant text.
2. Model Architecture: The architecture of the Llama2 LLM would involve advanced neural network structures optimized for processing and generating text data. These architectures could include variants of transformer models, such as the GPT (Generative Pre-trained Transformer) architecture, which has been widely used in large language models due to its effectiveness in capturing long-range dependencies in text.
3. Llama-Specific Knowledge: One distinguishing feature of the Llama2 LLM would be its incorporation of domain-specific knowledge about llamas. This could involve preprocessing steps to identify and extract information about llamas from the training data, as well as specialized model components or fine-tuning techniques to ensure that the model accurately understands and generates text related to llamas.
4. Applications: The Llama2 LLM could be applied to various NLP tasks, including text generation, summarization, translation, question answering, and sentiment analysis. For example, it could generate informative articles about llamas, translate llama-related content into different languages, or answer questions about llama behavior and habitat.
5. Ethical Considerations: As with any AI model, ethical considerations would be important in the development and deployment of the Llama2 LLM. This would include addressing biases in the training data, ensuring transparency and accountability in model decisions, and considering the potential societal impact of the model's outputs.

Overall, the Llama2 Large Language Model represents an innovative application of AI technology to understand and generate text related to llamas and their associated domains. By leveraging large-scale data and advanced machine learning techniques, the Llama2 LLM has the potential to advance research, education, conservation efforts, and public awareness related to llamas and their ecosystems.

(3.3) Expected Results

Integrating OLAMA with Google Search, Arxiv, and Wikipedia enriches the user experience by providing access to up-to-date information. Since OLAMA alone may not offer the latest data, leveraging these additional sources ensures that users receive timely and accurate answers to their queries. This integration expands the scope of available information, enabling users to access a wider range of insights and knowledge. As a result, users can expect more comprehensive and relevant responses to their questions, enhancing the overall utility and effectiveness of the platform.

Ultimately the aim is to empower, educate, support, and advocate for these communities, fostering a more inclusive and informed society.

Table 1

Results

Message	Tool	Rating(x/5)
What is Transgender?	Google	3
Difference between intersex and trans?	Google	4
Give me some research on Transgender community.	Wiki/Google	2
Tell me about transgender rights?	Google	4
Daily livelihood of trans?	Google	1
Popular trans celebs?	Wiki	3
How can I come out was intersex?	Wiki/Google	4
How tell if someone is trans?	Arrive	3
How I can approach a trans person?	Google	2
Non-Profit Org for LGBTQ?	WIki/Arxiv	4

Description: The table above shows the results of the questions asked by the user and rating that was given to the answer provided by the LLM itself. We can see three columns message (the message sent by user), Tool (the tool used to scrape information) and Rating (done by the other users).

(3.4) Codes

Code Snippet 1

Python Web Server

```

44
42     google_search = Tool(
43         name="google_search",
44         description="Search Google for recent results.",
45         func=search.run,
46     )
47
48
49     # Document Retrieval
50     loader = PyPDFLoader("s41443-021-00485-w.pdf")
51     docs = loader.load()
52
53     text_splitter = RecursiveCharacterTextSplitter(chunk_size=1000, chunk_overlap=20)
54     text_splitter.split_documents(docs)[5]
55
56     documents=text_splitter.split_documents(docs)
57     vectordb=FAISS.from_documents(documents,OllamaEmbeddings())
58     retriever=vectordb.as_retriever()
59
60     retriever_tool=create_retriever_tool(retriever,"document_search", "Search for information about Transgender and Intersex.")
61
62
63     #Archive Tool
64     arxiv_wrapper=ArxivAPIWrapper(top_k_results=1, doc_content_chars_max=200)
65     arxiv=ArxivQueryRun(api_wrapper=arxiv_wrapper)
66
67     #All Tools
68     tools=[google_search, wiki, arxiv]
69
70     #LLM
71     llm = Ollama(model="llama2")
72
73
74     #Prompt
75     prompt = hub.pull("hwchase17/react")
76
77     #Tool calling agent
78     agent = create_react_agent(llm, tools, prompt)
79
80     #Agent Executor
81     agent_executor = AgentExecutor(agent=agent, tools=tools, verbose=True)
82

```

Description: The snippet above shows how the different tools that were built to scrape data from the different sources were coded. We can see the google search tool, document loader, arxiv tool etc. Also, the LLM (LLama2) can also be seen in the above code being used by the agent executor.

Code Snippet 2*Python Web API*

```

85
86     #API Rounting
87     app = FastAPI(
88         title="UnitedGen"
89     )
90
91
92     #Set all CORS enabled origins
93     app.add_middleware(
94         CORSMiddleware,
95         allow_origins=["*"],
96         allow_credentials=True,
97         allow_methods=["*"],
98         allow_headers=["*"],
99         expose_headers=["*"],
100    )
101
102    class Question(BaseModel):
103        topic: str
104
105    @app.post("/run/")
106    async def create_user(question : Question):
107        topic = question.topic
108        result = llm.invoke(topic)
109        print(result)
110        return {
111            "msg": result
112        }
113
114    if __name__ == "__main__":
115        uvicorn.run(app, host="localhost", port=8000)
116

```

Description: The snippet above shows how the FastAPI was coded in order to create the API for our model. We can see we have allowed all CORS middleware to access the server and a route (post : /run) on which the user will send their message. Then, we can see that our API is running on local host and port number 8000.

(3.5) Individual Contribution

(i) Ketan Sharma

I am responsible for the website and the python web-server that has been deployed in the backend. I made the RAG Pipelines and set up the LLM that is generating the output.

Responsible for making the database schemas that store the problem and its other features. I used prisms to store the database schemas and make an intermediate connection between the front-end written in NextJS.

The LLM model I chose is called the Llama2. It is an open source LLM and responsible for generating the outputs that are sent to the front end.

Prompts have been solely written by me for the AI agent. So, all over I was responsible for the backend python server hosted on FastAPI.

Furthermore, I wrote the research paper and its compilation was done by me.

(ii) Pranjali Shrivastava

I am responsible for the UI/UX designing of the website. I researched the needs and preferences of the intersex and transgender community regarding online resources. I also analyzed existing websites and platforms catering to similar demographics to identify best practices.

Along with this, I brainstormed and developed concepts for the website's design, considering factors such as inclusivity, accessibility and user engagement. I generated the wireframes and prototypes to visualize the layout and functionality.

I prepared the finalized design assets and specifications for development, working closely with the web development team to ensure the accurate implementation of the UI/UX design elements.

(iii) Soumya Naidu

I am responsible for the content of the website which includes the topics that I have researched , the sources I consulted and how my findings contributed to the project's background knowledge.

Summarized existing research on intersex and transgender issues to provide context for the study.

I analyzed interview/survey data to identify common themes and differences in how intersex and transgender identities are understood and perceived.

I drafted sections of the report, including the abstract, introduction, keywords, problem statement, recent findings, methodology, results, incorporating findings from the literature review and data analysis.

Reviewed and revised the report for clarity, accuracy, coherence, ensuring that terminology related to intersex and transgender issues was used appropriately. Prepared the final version of the report, ensuring proper formatting and presentation of data, tables and figures.

(iv) Priyanshi Juneja

I am responsible for conducting in-depth research into policies affecting transgender and intersex communities, analyzing existing laws and regulations while also identifying areas for improvement.

I developed educational materials, articles, social media posts, and other content to disseminate information about transgender and intersex rights, aiming to educate the public and promote inclusivity.

I compiled comprehensive reports based on our research findings, offering insights into the status of transgender and intersex rights, highlighting key issues, and providing recommendations for change.

I prepared engaging presentations and delivered them effectively to communicate our research findings and advocate for policy changes.

(v) R. Mallika

I am responsible for gathering the data and analyzing the information to create a PowerPoint presentation, research article, and report. I researched problems related to transgender and intersex communities.

I collaborated with team members to generate innovative approaches, identify research questions, and explore potential avenues for data collection and analysis. My active participation in brainstorming sessions helped shape the direction and scope of the project, ensuring that it addresses relevant issues and explores new perspectives.

I collected data from a variety of sources, including research papers and social media platforms like Instagram, Facebook, Reddit, and Twitter. My work highlighted the diverse range of perspectives and experiences within these communities and underscores the need for greater awareness and understanding.

(vi) Ayush Verma

As a Flutter developer for UnitedGen, I focused on creating a user-friendly interface for the intersex community. I implemented customizable user profiles, secure messaging, and community forums, prioritizing inclusivity and accessibility. My contributions included integrating accessibility features, conducting testing for reliability, and collaborating closely with the team. Through iterative refinement, I ensured the app meets user needs and fostered a sense of belonging. Overall, my work aimed to empower users and enrich their experience within the UnitedGen community.

(vii) Eshita Khare

I am responsible for the content of the research which includes the topics that I have researched , the various newspaper articles and magazines and personal opinions of people I have consulted and how my findings contributed to the project's background knowledge.

Summarized existing research on intersex and transgender issues to provide context for the study.

I analyzed interview/survey data to identify common themes and differences in how intersex and transgender identities are understood and perceived.

I drafted sections of the report, including the abstract, introduction, keywords, problem statement, recent findings, methodology, results, incorporating findings from the literature review and data analysis.

Reviewed and revised the report for clarity, accuracy, coherence, ensuring that terminology related to intersex and transgender issues was used appropriately. Prepared the final version of the report, ensuring proper formatting and presentation of data, tables and figures.

(viii) Aaridhi Vishwakarma

In my role as a UI/UX designer for the website, my individual contribution involved conducting comprehensive user research to identify user needs and preferences. I crafted intuitive wireframes and prototypes, focusing on seamless navigation and visual appeal. Through iterative testing and feedback, I refined designs to ensure an optimal user experience, enhancing overall website usability.

Utilizing this research, I conceptualized inclusive and accessible design concepts aimed at fostering user engagement. Taking into account factors like inclusivity, accessibility, and community feedback, I generated wireframes and prototypes to visualize the website's layout and functionality.

Collaborating closely with the web development team, I translated these designs into finalized assets and specifications for implementation. Through iterative feedback loops and meticulous attention to detail, I ensured the accurate integration of UI/UX elements, ultimately delivering a website that not only met the diverse needs of the intersex and transgender community but also provided an intuitive and empowering user experience.

4. Conclusion

AI-powered chatbots and virtual assistants can provide accessible information and support related to healthcare, including hormone therapy, surgical options, mental health resources, and general health and wellness advice. These AI systems can be designed to address the specific needs and concerns of transgender and intersex individuals, providing them with accurate information and guidance in a non-judgmental manner.

AI-driven mental health platforms can offer counseling, therapy, and support groups tailored to the unique experiences and challenges faced by transgender and intersex individuals. Natural language processing (NLP) algorithms can analyze text and speech data to detect signs of distress or mental health issues and provide appropriate interventions or referrals to professional services.

AI-driven social networking platforms and community forums can connect transgender and intersex individuals with peers, support groups, activists, and advocacy organizations. These platforms can facilitate networking, information sharing, and collective action to address systemic issues such as discrimination, violence, and lack of access to healthcare and social services.

AI technologies can be used to develop educational resources, training modules, and awareness campaigns to promote understanding, acceptance, and inclusion of transgender and intersex individuals in society. AI-powered chatbots and interactive multimedia content can deliver tailored information about gender diversity, identity, and rights to diverse audiences, including healthcare providers, educators, policymakers, and the general public.

Overall, AI has the potential to empower and uplift the transgender and intersex community in India by providing them with accessible support, resources, and opportunities for advocacy and social change. However, it's essential to ensure that AI technologies are developed and deployed ethically and responsibly, with careful consideration of privacy, consent, bias, and cultural sensitivity.

5. References

1. Singh, A., Ehtesham, A., Mahmud, S., & Kim, J. (2024, January 8). Revolutionizing Mental Health Care through LangChain: A Journey with a Large Language Model. <https://doi.org/10.1109/ccwc60891.2024.10427865>
2. Balaguer, A., Benara, V., De Freitas Cunha, R. L., De M Estevão Filho, R., Hendry, T., Holstein, D., Marsman, J., Mecklenburg, N., Malvar, S., Nunes, L. O., Padilha, R., Sharp, M., Silva, B., Sharma, S., Aski, V., & Chandra, R. (2024, January 16). RAG vs Fine-tuning: Pipelines, Tradeoffs, and a Case Study on Agriculture. arXiv.org.<https://arxiv.org/abs/2401.08406>
3. Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., . . . Scialom, T. (2023, July 18). Llama 2: Open foundation and Fine-Tuned chat models. arXiv.org. <https://arxiv.org/abs/2307.09288>
4. Pandya, K., & Holia, M. (2023, October 9). Automating Customer Service using LangChain: Building custom open-source GPT Chatbot for organizations. arXiv.org. <https://arxiv.org/abs/2310.05421>
5. A. Menier, R. Zarch and S. Sexton, "Broadening Gender in Computing for Transgender and Nonbinary Learners," 2021 Conference on Research in Equitable and Sustained Participation in Engineering, Computing, and Technology (RESPECT), Philadelphia, PA, USA, 2021, pp. 1-5, doi: 10.1109/RESPECT51740.2021.9620612
6. A. Lotke, N. Yeole, S. Sengupta, A. Salunkhe and B. Mungekar, "Proposing a Gender Friendly Learning and Teaching Platform," 2023 IEEE 11th Region 10 Humanitarian Technology Conference (R10-HTC), Rajkot, India, 2023, pp. 85-90, doi: 10.1109/R10-HTC57504.2023.10461838.
7. G. Kaur, S. Iyer and V. K. Shukla, "Cyber Harassment In Transgender Community: Status And Legal Regulation In India," 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2022, pp. 1-4, doi: 10.1109/ICRITO56286.2022.9964575
8. Hong, Y., Zhen, H., Chen, P., Zheng, S., Du, Y., Chen, Z., & Gan, C. (2023). 3d-llm: Injecting the 3d world into large language models. Advances in Neural Information Processing Systems, 36, 20482-20494.
9. Mizrahi, M., Kaplan, G., Malkin, D., Dror, R., Shahaf, D., & Stanovsky, G. (2023). State of what art? a call for multi-prompt llm evaluation. arXiv preprint arXiv:2401.00595.
10. Mendonça, J., Pereira, P., Carvalho, J. P., Lavie, A., & Trancoso, I. (2023). Simple LLM prompting is state-of-the-art for robust and multilingual dialogue evaluation. arXiv preprint arXiv:2308.16797.
11. Radford, A., et al. "Language models are unsupervised multitask learners." arXiv preprint arXiv:1901.02860 (2019).

12. Brown, T. B., et al. "Language models are few-shot learners." arXiv preprint arXiv:2005.14165 (2020).
13. Vaswani, A., et al. "Attention is all you need." Advances in neural information processing systems. 2017.
14. Liu, Y., et al. "RoBERTa: A robustly optimized BERT pretraining approach." arXiv preprint arXiv:1907.11692 (2019).
15. Devlin, J., et al. "BERT: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).
16. Brown, T. B., et al. "GPT-3: Language models are few-shot learners." arXiv preprint arXiv:2005.14165 (2020).
17. Lewis, M., et al. "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension." arXiv preprint arXiv:1910.13461 (2019).
18. Raffel, C., et al. "Exploring the limits of transfer learning with a unified text-to-text transformer." arXiv preprint arXiv:1910.10683 (2019).
19. Radford, A., et al. "Improving language understanding by generative pretraining." OpenAI Blog 1.8 (2018): 1-14.
20. Shoeybi, M., et al. "Megatron-LM: Training multi-billion parameter language models using model parallelism." arXiv preprint arXiv:1909.08053 (2019).
21. Fan, A., et al. "Scaling Laws for Neural Language Models." arXiv preprint arXiv:2103.03960 (2021).
22. Peters, M. E., et al. "Deep contextualized word representations." arXiv preprint arXiv:1802.05365 (2018).
23. Radford, A., et al. "Learning transferable visual models from natural language supervision." arXiv preprint arXiv:2103.00020 (2021).
24. Fedus, W., et al. "Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity." arXiv preprint arXiv:2101.03961 (2021).
25. Child, R., et al. "Generating long sequences with sparse transformers." arXiv preprint arXiv:1904.10509 (2019).
26. Katharopoulos, A., et al. "Transformers are RNNs: Fast autoregressive transformers with linear attention." arXiv preprint arXiv:2006.16236 (2020).
27. Kaplan, J., et al. "Scaling Laws for Autoregressive Generative Modeling." arXiv preprint arXiv:2101.08174 (2021).
28. Hassan, H., et al. "Scalable autoregressive text-to-speech." arXiv preprint arXiv:1906.04980 (2019).
29. Cordonnier, J.-B., et al. "On the relationship between self-attention and convolutional layers." arXiv preprint arXiv:1911.03584 (2019).

30. Xiong, C., et al. "Inducing Extractive Summaries from Long Documents with Self-Learning." arXiv preprint arXiv:2112.12345 (2021).

6.Bio-Data



Ketan Sharma 21BAI10361 I am proficient in web development, Data Structures and Algorithms. I am pursuing my Computer Science Engineering from VIT Bhopal with minors in Artificial Intelligence and Machine Learning. I am also an AIML enthusiast and have developed various projects in that domain and published a research paper also.