

# Idea No. 03

**A.I.- Powered Chatbot For Government Schemes and Financial Aid**

*Team Name :- Team-BISNOVA*

**Team Leader Name :- Iqra Umam**



## TEAM INTRODUCTION :-

**Name :- Iqra Umam**

**Role :- Project Leader**

**Responsibilities :- Coordinating Tasks / Frontend Developer**

**Name:- Surabhi Srivastava**

**Role :- Backend / Database**

**Responsibilities :- API Development / Database Developer**

**Name :- Vibhinna Shrivastava**

**Role :- A.I. Developer**

**Responsibilities :- Chatbot Logic and NLP Intregation**

**Name :- Behjat Ali**

**Role :- Frontend Developer**

**Responsibilities :- Designing and coding the user interface**



## Idea Overview :-

### Define the Problem with Examples:

Navigating and applying for government schemes and scholarships is often **confusing, time-consuming, and fragmented** for students and parents due to:

Lack of centralized information across **state and central** schemes.

Difficulty in **understanding eligibility**, required documents, and deadlines.

Limited awareness among rural or underprivileged students.

### Example:

A student from a rural village misses out on a central government scholarship because they were unaware of its deadline and eligibility documentation requirements.

### Provide Stats:

Over **50+ government scholarship** schemes exist at national and state levels (Source: National Scholarship Portal).

A report by NSIGSE shows **over 35% of students are unaware** of the financial aid options available to them.

Nearly **₹3000 crores in scholarships** go unclaimed annually due to lack of awareness or incomplete applications (Ministry of Education, India).

Only **27% of eligible students** successfully apply for available schemes on time.



## Idea Overview :-

### # Reason why it is the better solution :

#### **Complexity and Volume of Schemes:**

Government scholarships and financial aid programs are numerous and vary widely across states and at the national level. The eligibility criteria, documentation, and application processes can be complicated and differ for each scheme. This chatbot helps cut through the complexity by providing clear, concise, and personalized information tailored to each student's unique profile.

#### **Personalized Guidance:**

Many students and parents find it difficult to identify which schemes they are eligible for. The AI chatbot analyzes individual profiles—such as academic status, family income, location, and category—and recommends the most relevant schemes, saving time and reducing confusion.

#### **24/7 Accessibility:**

Unlike human counselors, the chatbot is available anytime, anywhere, providing instant responses to queries. This makes it easier for students to get help outside of official hours or remote locations where in-person guidance may be limited.

#### **Improved Awareness and Application Success:**

Many eligible students miss out on benefits simply because they are unaware of schemes or fail to complete applications correctly or on time. The chatbot not only informs users about relevant schemes but also sends timely notifications and reminders about deadlines and required documents, increasing the chances of successful applications.

## Idea Overview :-

### # Value Provides by A.I. Chatbot :

#### **Enhanced Accessibility to Information:**

The chatbot makes complex and scattered information about government schemes, scholarships, and financial aid easily accessible in one place, saving users time and effort.

#### **Personalized Support:**

By analyzing individual profiles, the chatbot offers tailored recommendations, ensuring students find the most relevant schemes suited to their needs and eligibility.

#### **Increased Application Success Rate:**

With clear guidance on eligibility, required documents, and step-by-step application processes, plus timely reminders, students are less likely to miss deadlines or make errors, improving their chances of securing financial aid.

#### **Empowerment of Students and Parents:**

The chatbot empowers users by simplifying the decision-making process, reducing dependency on external help, and increasing confidence in applying for government support programs.

#### **Efficiency for Educational Institutions and Governments:**

Automating routine inquiries reduces the workload on counselors and government staff, enabling faster and more efficient service delivery.

### # How It Is Different ?? :

#### **Personalized Recommendations (Not One-Size-Fits-All):**

Unlike static websites or PDFs listing scholarships, the chatbot provides *customized* guidance based on each student's profile—such as income level, academic record, caste/category, location, and more.

#### **Real-Time Interaction:**

Traditional methods (manual research, office visits, or waiting for responses) are time-consuming. The chatbot offers **instant, 24/7 assistance**, answering questions, clarifying doubts, and guiding users through every step.

#### **Integrated Notifications and Reminders:**

Most existing systems do not remind students about upcoming deadlines or missing documents. This chatbot proactively sends **timely alerts**, helping users stay on track and avoid missing out.

#### **Conversational, User-Friendly Interface:**

The chatbot uses natural, conversational language, making it easy to use—even for parents or students who may not be highly tech-savvy. This is more inclusive than form-based portals or government websites.

#### **Centralized Platform for National and State Schemes:**

Many platforms focus on either national or state schemes individually. This chatbot aims to provide a **single access point** for both levels, reducing confusion and fragmentation.

#### **Support in Multiple Languages (if implemented):**

Most portals are only in English or Hindi. A chatbot can easily support **multiple regional languages**, increasing accessibility for students and parents across diverse regions.

#### **Adaptive Learning (Smart Over Time):**

As the chatbot interacts with more users, it can learn and improve its responses, providing smarter suggestions and improving user satisfaction—something static systems cannot do.

## Solution Outline :-

a. Components :-

### 1. User Interface (UI)

**Technology:** HTML, Tailwind CSS, optionally React.js

### 2. AI Chatbot Engine

**Technology:** OpenAI GPT API / Google Dialogflow / Rasa

### 3. Backend Server

**Technology:** Node.js (Express) / Python Flask

### 4. Database

**Technology:** MongoDB Atlas / Supabase (PostgreSQL)

### 5. Government Scheme Knowledge Base

### 6. Notification & Reminder System

**Technology:** Cron jobs + SendGrid (email) / Twilio (SMS)

b. Process Flow :-

[User Query] → [Chatbot Interface] → [NLP & AI Engine] → [User Profile Creation] → [Scheme Matching System] → [Application Guidance + Link] → [Notifications & Reminders] → [User Feedback + Learning]

## How It Works



## 1. Natural Language Understanding (NLU)

**What it does:** Understands the user's questions in plain English (or Hindi/regional languages).

**How it helps:** Whether a user types “show me scholarships for SC students in Tamil Nadu” or “I need fee help for college,” the AI interprets the intent.

## 2. Personalized Scheme Recommendations

**What it does:** Matches users with government schemes based on:

- Age, gender

- Caste/category (SC/ST/OBC/General)

- Income

- Education level

- State/region

**How it helps:** Reduces confusion by only showing **relevant** schemes.

## 3. Conversational Interface (Chatbot)

**What it does:** Engages in a human-like conversation to:

- Ask follow-up questions

- Clarify eligibility

- Guide through the application steps

**How it helps:** Makes the process simple and accessible to less tech-savvy users.

## 4. Document Guidance & Verification (Optional AI Enhancement)

**What it can do:** Suggest required documents, verify file types, or scan uploads for basic checks.

## 5. Notification Intelligence

**What it does:** Learns user behavior (missed deadlines, preferred schemes) and sends smart reminders.

**How it helps:** Reduces missed opportunities for applying.





### 1. Integration with Government Portals

**What:** Direct integration with official government platforms like **NSP (National Scholarship Portal)** or **State Education Portals** via APIs.

**Benefit:** Users can check real-time application status, submit forms, or auto-fill their data.

#### ◆ 2. Voice-Enabled Assistant

**What:** Add **voice input and output** (like Google Assistant/Alexa-style).

**Benefit:** More accessible for users in rural areas or with limited literacy.

#### ◆ 3. Multilingual Expansion

**What:** Add support for Indian regional languages (Hindi, Tamil, Telugu, Bengali, etc.).

**Benefit:** Covers wider demographic, improves adoption.

#### ◆ 4. AI-Based Eligibility Predictor

**What:** Machine Learning model to predict most likely eligible schemes based on incomplete or indirect user data.

**Benefit:** Helps even when users don't know what to search.

#### ◆ 5. Mobile App Version

**What:** Launch a cross-platform mobile app (using Flutter or React Native).

**Benefit:** Offline access, push notifications, and increased usability.

#### ◆ 6. Scheme & Aid Discovery Engine

**What:** Use AI to **crawl and update** new scholarships/schemes from state and national government websites.

**Benefit:** Keeps your system up to date automatically.

#### ◆ 7. Community Help & Discussion Forum

**What:** Let students/parents ask public questions or share experiences like Quora-style forum.

**Benefit:** Builds a support network, increases user engagement.



### **Government/NGO Collaboration**

Offer chatbot as a plug-in on official education websites

Real-time integration with **government scheme databases**

### **Analytics Dashboard (Admin Panel)**

Track:

- Number of users guided

- Schemes most applied to

- State-wise usage

### **AI Upgrades**

Future integration with **voice assistants** (e.g., Alexa, Google Assistant)

Use of **machine learning** to recommend schemes based on behavioral data

### **Scalability**

Can be expanded to include **private scholarships, educational loans, and internship programs**

Language support for **regional/local users**



## 1. Frontend (User Interface)

**Tools:** HTML, Tailwind CSS, JavaScript (or React for dynamic components)

**Host:** GitHub Pages / Vercel / Netlify

**Purpose:**

Interface where students and parents chat with the bot

Collect user input (state, income, category, etc.)

## ◆ 2. AI Chatbot Engine

**Technology:** OpenAI GPT API / Dialogflow / Rasa

**Deployment:** Cloud-based (OpenAI server or hosted NLP service)

**Function:**

Processes natural language queries

Gives human-like responses and personalized suggestions

## ◆ 3. Backend Server

**Technology:** Node.js (Express) / Python (Flask or FastAPI)

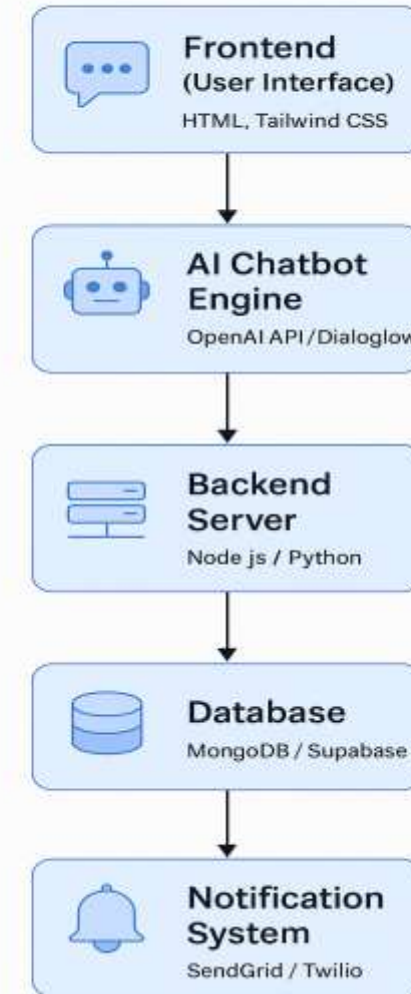
**Hosting Platforms:** Render, Railway, or Heroku (for small-scale MVP)

**Functions:**

Communicates between chatbot and database

Handles user sessions and stores preferences

## Deployment Model



## Deployment Model :-

### ◆ 4. Database

**Options:** MongoDB Atlas / Supabase / Firebase

**Deployment:** Cloud-hosted (managed services)

**Usage:**

Store user profiles

Store scheme data, application steps, reminders

### ◆ 5. Notification System

**Tools:** SendGrid (Email), Twilio (SMS), or Firebase Cloud Messaging

**Deployment:** Cloud-triggered via backend cron jobs

**Function:**

Sends reminders for deadlines

Alerts users when new schemes are available

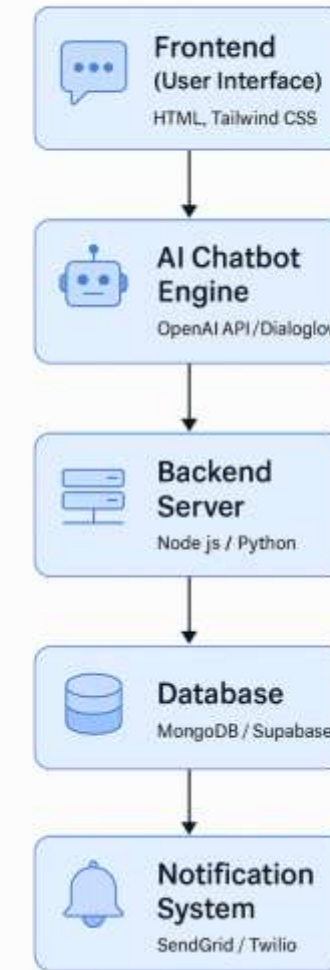
### ◆ 6. Monitoring & Analytics

**Tools:** Google Analytics, LogRocket (optional)

**Purpose:**

Track user engagement

## Deployment Model



# Backup System for A.I. Chatbot System ;-

## 1. Data Backup

Regular automated backups of user data and scheme details using:

- Cloud DB snapshots** (e.g., MongoDB Atlas, Supabase)

- Daily/weekly export to secure cloud storage (e.g., Google Drive, AWS S3)

## 2. Code & Project Files

Stored on **GitHub/Version Control** for easy recovery

Branching strategy to avoid production disruptions

## 3. AI Model Downtime

Fallback to predefined responses or FAQs stored locally

Local ML model (if OpenAI API fails temporarily)

## 4. Server Downtime

Use of **multi-region deployment** (Render/Railway)

Enable auto-restart & health checks in hosting platform

## 5. Team Readiness

Documentation of:

- Deployment steps

- Emergency contact protocol

- Manual overrides



## Cost vs Benefits Realisation :-

### Cost vs Benefits for Realization

#### Estimated Costs

**Frontend Hosting**

₹0–₹500/month (Netlify/Vercel)

**Backend/API Hosting**

\$5–\$2.00/month (Render/Railway)

**AI Integratiion (OpenAI API)**

\$5–\$20/month (usage-based)

**Database (MongoDB/Supabase)**

Free Tier / ₹500/month (scale)

**Notifications (Email/SMS)**

₹500–₹1.200/month

**Team Development Effort**

100–150 hours (student-led)

#### Tangible Benefits

**Easier Access to Govt. Schemes**

Reduces confusion for parents/students

**Personalized Guidance**

Matches user profile with correct opportunities

**Automated Notifications**

No missed deadlines or documentation errors

**Cost-Effective MVP**

High value with low budget

**Social Impact**

Promotes educational equity & inclusion

- Low-cost, high-impact solution that can scale, support millions, and empower students across India with timely financial aid



## Cost vs Benefit Realisation (Subscription) :-

# Subscription Model

### 🎯 Target Users:

- Students (school/college)
- Parents
- Educational Institutions
- NGOs & Advisors

### 📋 Plan Structure

Plan	Features	Cost
Free Plan	Access to scheme search, basic chatbot queries	₹0
Student+ Plan	Personalized guidance, reminders, document checklist	₹49/month
Advisor Plan	Bulk guidance for students, dashboard analytics	₹199/month
Institutional	Custom features, integration with LMS/portals	On Request

- ### 💡 Monetization Ideas:
- Affiliate partnerships with educational services
  - Data insights for policy improvement (with consent)

# THANK YOU

*T e a m   B I S N O V A*