ChatGPT

Requirements Document: AI-Powered Education Platform (MagicSchool.ai Replication)

Introduction and Objectives

The goal is to develop an AI-powered **education platform** similar to MagicSchool.ai, tailored for Indian educators and students. This platform will leverage artificial intelligence to **reduce teacher workload, combat burnout, and improve student learning outcomes** [1] [2] . MagicSchool's success shows that a suite of AI tools can save teachers significant time (self-reported **7–10 hours per week** on routine tasks) [3] , allowing them to focus on teaching and student interaction. In India, where **65% of school teachers are overloaded with work** according to a national survey [4] , such a platform can provide much-needed support. The objectives of the platform are:

- **Save teachers' time** by automating or assisting with lesson planning, content creation, assessments, and administrative paperwork, thereby alleviating overwork and burnout [1] [3] .
- **Improve teaching quality and personalization** by providing tools for differentiated instruction, individualized student support (e.g. special education plans), and creative lesson enhancements.
- **Empower students** with AI-driven learning tools (e.g. study aids, chatbots) that promote engagement and self-learning, all under teacher supervision for responsible AI use.
- **Ensure ease of use** so that even non-technical educators can quickly adopt the platform. This includes a **simple, intuitive interface** and guided workflows, aligning with the idea that teachers (not technical skills) are the "magic" behind education [5] .
- **Incorporate an Indian perspective** by aligning content with Indian curricula (CBSE, ICSE, state boards), supporting local languages, and addressing local classroom challenges. The platform should respect cultural and educational norms in India, providing relevant examples and context in generated content.

By achieving these objectives, the platform will help **reclaim valuable time for educators, combat burnout, and elevate the art of teaching** in India [6] , similar to what MagicSchool has done globally.

Target Users and Use Cases

Primary Users – Teachers: The platform is designed foremost for K-12 school teachers across India (and potentially tutors or college faculty). Teachers will use the AI tools to generate lesson materials, assessments, and communications. For example, a teacher could instantly create a lesson plan on a Class 8 science topic or generate a Hindi worksheet for a grammar lesson. The AI assistant will serve as a *"thought partner"* that jump-starts teachers' work (as MagicSchool users describe) while still allowing teacher input and customization [7] [8] . Key use cases for teachers include: lesson planning, content creation, exam/quiz preparation, grading assistance, student feedback, and parent communications.

Secondary Users – Students: A student-facing side will allow learners to interact with AI in a safe environment, under teacher guidance. Students can use AI tools for studying (e.g. quizzing themselves, getting explanations) and creative projects (like generating essays, asking a tutor-bot questions, etc.). MagicSchool provides 50+ student tools (e.g. a *"Study Bot"* or *"Tutor Me with AI"* using an educational chatbot named Raina) [9] [10] . In our platform, students might access a simplified set of tools for

practice and exploration, building **AI literacy and personalized learning** experiences [11] . Teachers should have oversight of student usage for safety and to integrate these activities into their teaching.

Administrators (optional): School or district admins may use the platform's analytics to monitor usage and effectiveness. They might configure the platform to align with school policies (for example, uploading curriculum documents to tailor the AI outputs). While not the primary users, admins are stakeholders for enterprise features like custom tool deployment and data dashboards (as MagicSchool's enterprise offering demonstrates [12] [13]).

Functional Requirements

The platform will provide a **suite of AI-driven tools (target ~80+ teacher tools, ~50+ student tools)** similar to MagicSchool [1] . These tools fall into several functional categories:

1. Lesson Planning & Content Generation Tools

- **Lesson Plan Generator:** Teachers can generate comprehensive lesson plans for any topic or objective by entering a topic, standard, or few keywords [14] . The AI should produce a structured lesson plan including learning objectives, instructional activities, and assessment ideas, aligned with the chosen curriculum standards [15] . For Indian context, the tool should allow alignment with NCERT or state board standards (e.g. select Class 10 CBSE Science), ensuring the content matches local syllabi.

- **Academic Content & Worksheet Generator:** The system will create original instructional content upon request [16] [17] . For example, a teacher could ask for a **worksheet or practice sheet** on a math chapter or a **set of examples** for a literature lesson. The AI should generate problems, reading passages, or exercises at appropriate difficulty levels. Teachers can specify criteria (e.g. "Grade 5, fraction word problems") and the tool produces a ready-to-use worksheet [18] . Content should be **standards-aligned and age-appropriate** (e.g. incorporate elements of NEP 2020 like competency-based questions if relevant).

- **Presentation Slide Generator:** A tool to automatically create slide decks for a given lesson or topic. Teachers input a topic or even upload a chapter text, and the AI generates a set of slide outlines or bullet points, which can be exported to common formats (Google Slides, PowerPoint) [19] [20] . This helps teachers quickly prepare visual aids for classes.

- **Example and Hook Generator:** To support engaging lessons, the platform can generate **lesson hooks** or real-world examples related to a topic. For instance, given a concept, it can suggest a quick opening activity or a relatable example to introduce the lesson [21] . This encourages student interest and contextual learning ("Make it Relevant" feature as in MagicSchool) [22] .

- **Unit Plan / Syllabus Generator:** For longer-term planning, teachers can generate a draft **unit plan** or **course syllabus** by specifying the topic range and duration (e.g. "Physics Term 1 for Grade 11") [23] . The AI will outline week-by-week topics, key objectives, and suggested activities, which teachers can refine.

2. Assessment & Evaluation Tools

- **Quiz and Exam Generator:** The platform will let teachers create quizzes, tests, or exam papers automatically. Teachers specify the topic or even particular chapters, and the AI generates

multiple-choice questions, short answers, or even descriptive questions as needed [24] . For example, a teacher could generate a 20-question multiple-choice quiz on Chapter 2 of History, or even a full-length practice exam. The tool should support **difficulty level settings** and format (unit test, midterm, competitive exam prep, etc.). MagicSchool's tool can create quizzes based on any topic or standard [24] , and similarly, our platform should allow generation of question banks aligned with Indian exam styles (including board exam style questions or IIT-JEE/NEET style for higher secondary, if in scope).

- **Rubric Generator:** Teachers often need rubrics to grade assignments. The platform will include a rubric generator – given an assignment description, it produces a rubric with criteria and scoring guidelines [25] . This helps standardize evaluations. For instance, if a teacher is assigning an essay, the AI can output a table of criteria (content, language, originality, etc.) with levels of achievement.

- **Automated Grading Feedback:** A **writing feedback** tool will use AI to review student-written text or answers and provide formative feedback [26] . Teachers can input a sample of student work (or multiple answers), and the AI highlights strengths and areas for improvement based on custom criteria [26] . This is useful for English essays or short answers – the AI might say a student's essay has good arguments but needs better grammar, for example. (The final grading decision remains with the teacher, but this saves time in analysis.)

- **Report Card Comment Generator:** To streamline report writing, the system can generate **report card comments** or student feedback summaries given some notes on student performance [27] . The teacher provides key points (e.g. "struggles in math, excellent in art, improving in participation") and the AI drafts professional, constructive comments about the student's progress. This can be especially helpful when a teacher has to write comments for dozens of students.

- **Adaptive Testing & Questioning:** The platform may include tools to generate questions at various cognitive levels (e.g. Bloom's taxonomy or Depth of Knowledge levels). For example, a teacher can get higher-order thinking questions on a topic to challenge advanced students ("DOK level 4 questions on photosynthesis") [28] . This helps in differentiated assessments.

3. Differentiation & Student Support Tools

- **Individualized Education Plan (IEP) and Support Drafting:** For students with special needs or learning plans, the platform can generate draft **IEPs** or accommodation plans based on provided student information and goals [29] . A teacher or special educator inputs the student's needs, and the AI suggests goals, accommodations, and strategies (aligned with best practices). This speeds up the paperwork for special education while ensuring thorough plans [30] .

- **Differentiation Strategies:** The platform will help teachers adapt content for varied learners. For example, a **Text Scaffolder** can take any given passage and simplify or annotate it for students who read below grade level [31] . A **Text Leveler** can adjust reading materials to different difficulty levels (e.g. converting a Grade 8 passage into a Grade 5 level text for weaker readers) [32] . Similarly, a **Vocabulary List Generator** can produce a list of key terms with definitions for any topic to pre-teach vocabulary [33] .

- **Remediation and Enrichment:** Tools like **Common Misconceptions** (to list typical misunderstandings in a topic) and **Multiple Explanations** (to re-explain a concept in different

ways) will be available [34] [35]. If a teacher finds students are struggling with a concept (e.g. fractions), they can ask the AI for alternative explanations or real-life analogies to help understanding. Conversely, for advanced students, the teacher could use an **Advanced Learning Plan** generator to suggest enrichment activities or deeper challenges [36].

- **Behavior and Counseling Aids:** The platform might include a **Behavior Intervention Suggestions** tool for classroom management support. Teachers describe a classroom behavior challenge, and the AI suggests positive intervention strategies or restorative practices (for instance, how to handle a student who frequently disrupts class) [30]. It could also generate **social stories** for younger children to help them understand routines or events (like a story about why we have fire drills) [32].

- **"AI-Resistant" Assignment Ideas:** Notably, to address the concern of students misusing AI, an interesting feature from MagicSchool is to get suggestions on making assignments more *AI-proof* [37]. Our platform can include a similar function: a teacher enters an assignment idea and the system advises how to tweak it so it's less likely to be solved entirely by AI (encouraging originality or hands-on work). This will be important for maintaining academic integrity as AI becomes ubiquitous.

4. Communication & Administrative Tools

- **Email and Letter Drafting:** The platform will assist in drafting professional communications. Teachers can use an **Email Generator** to compose emails to parents or colleagues by providing the basic information, and the AI outputs a clear, respectful message [38]. For example, a teacher can input points for a parent-teacher email about a student's progress, and get a polished email draft in English (and even translate it into Hindi or other languages as needed). Another feature is an **Email Responder** – given the text of an email received, the AI can draft a context-appropriate reply [38], saving teachers time in writing routine emails.

- **Newsletter and Announcement Writer:** Schools often send weekly updates. A **Class Newsletter Generator** will allow teachers or principals to generate a newsletter for families [39]. By inputting the week's highlights or a few bullet points (upcoming events, class achievements), the AI produces a friendly newsletter ready to send. This ensures consistent communication with parents in a time-efficient way.

- **Professional Documents:** Additional tools will handle other writing tasks. For instance, a **Letter of Recommendation Generator** helps draft recommendation letters for students (for scholarships, college admissions) using details about the student's accomplishments [40]. A **Meeting Agenda or Report Generator** could also be included for administrative tasks (e.g. creating an agenda for a staff meeting or summarizing student data in a report format). These reduce the burden of bureaucratic writing.

- **Social Media & Creative Outputs:** Though secondary to core teaching tasks, the platform can offer fun or creative generators (MagicSchool even has a *Teacher Jokes* generator for light-hearted classroom moments [41]). Similarly, a **Social Media Post Generator** can help teachers or schools draft social media content to share educational achievements [42]. These tools make the platform engaging and help build community, though they are supplementary.

5. AI Chatbot Assistants

- **Teacher's AI Assistant (Chatbot):** A central feature will be an AI-powered chatbot (similar to MagicSchool's **"Raina"** assistant) built specifically for education [43] . This chatbot will function as a **24/7 assistant** for teachers. They can ask it open-ended questions or requests, such as *"Give me ideas to introduce the concept of photosynthesis in a fun way"* or *"How can I accommodate a dyslexic student in this reading assignment?"*. The chatbot will respond with helpful suggestions, resources, or even moral support, drawing from a large knowledge base of pedagogy and subject matter. Importantly, the chatbot should maintain a **professional tone and safe content filters**, since it might be asked about sensitive educational scenarios.

- **Student Chatbots:** The platform will also allow creation of AI chatbots for student use in a controlled manner. Possible features:

- **Character Chatbot:** Students can "chat" with a historical figure or book character for learning purposes [44] . For example, a student could ask "Mahatma Gandhi" about his life philosophy, and the bot would respond in that persona (based on historical data). This makes learning interactive and engaging.
- **Custom Resource Bots:** Teachers can create a **custom chatbot trained on specific content** (by uploading a document or text) for their class [45] [46] . For instance, after uploading a chapter or a syllabus, a teacher could enable a chatbot that answers student questions only from that material. This is useful for revision: students ask the bot questions and it responds based on class notes.

- **Tutor/Study Buddy:** A **Study Bot or Tutor Bot** will help students practice any topic [9] [10] . Students might say, "I'm struggling with quadratic equations," and the bot can explain step-by-step or quiz the student with problems, acting as a personal tutor. The tutor bot ("Raina for Students") should follow the teacher's guidelines and be restricted to supportive roles (e.g., not just giving away answers but guiding the student).

- **No Complex Prompting Required:** A key design requirement is that teachers and students **do not need to craft complex AI prompts** to use these tools. The interface will offer **guided inputs and templates** for each tool. MagicSchool's approach is to eliminate the need for expert prompt engineering – their tools deliver results in seconds with minimal user input [47] . For our platform, this means each feature should have a simple form or question interface (e.g., "Enter topic" or checkboxes for tone, length) rather than requiring users to type long instructions. This lowers the barrier to entry for educators who are not tech-savvy, ensuring **simplicity and user-friendliness**.

6. Integration and Collaboration Features

- **Learning Management System (LMS) Integration:** The platform should integrate with popular LMS platforms and classroom tools used in India. Given many schools use Google Classroom, Microsoft Teams or similar, the platform will support **one-click export or sharing** of generated content to these systems [48] . For example, after creating a quiz, a teacher can directly export it to Google Forms or as a PDF/Word doc for printing. MagicSchool Plus offers one-click exports to Google Docs/Slides and LMS like Canvas and Schoology [49] [50] – our platform should aim for similar convenience with Google Classroom, Canvas, and possibly Indian e-learning portals like DIKSHA if feasible.

- **Single Sign-On (SSO):** To simplify access, the system will support SSO via Google, Microsoft, or other education SSO providers (like **Classlink or Clever** which MagicSchool supports) [50] . This ensures teachers and students can log in with existing school credentials. It's essential for district-level deployment and for ease of onboarding.

- **Content Upload and Retrieval:** Teachers should be able to **upload documents (PDFs, DOCs)** such as curriculum guides, lesson materials, or student essays into the platform. The AI can then use that content for specific tools (e.g., *"Chat with Docs"* which enables a chatbot to answer questions from an uploaded document [45]). This is crucial for an Indian context too – e.g., a teacher might upload a chapter from an NCERT textbook and generate questions or summaries from it, or a school might upload their policy documents to create a custom policy Q&A bot. The platform may use Retrieval-Augmented Generation (RAG) in the backend to incorporate such custom data [51] [52] .

- **Collaboration & Sharing:** The platform should allow teachers to share the AI-generated resources with others. For example, a teacher could share a lesson plan or worksheet with colleagues or to a community library. This fosters a community of practice. Also, if multiple teachers co-teach, they might collaborate on the same AI-generated content, editing and refining it together. Basic collaboration features (commenting, version history for content) would be beneficial.

- **Analytics Dashboard:** For administrators or even teachers themselves, include an **analytics view** to track usage and impact. This might show metrics like how many hours saved (based on tool usage), which tools are used most, student engagement stats (MagicSchool's *"Student Room Insights"* provides real-time data on student AI tool usage) [53] . In an Indian school setting, such data can help identify training needs or success stories (e.g., one teacher using the AI extensively and improving student results could inspire others).

Non-Functional Requirements

1. Ease of Use and UI/UX Design

The platform must be extremely **user-friendly and intuitive**, keeping in mind that teachers are busy and may not be tech experts. Key UI/UX requirements:

- **Clean Dashboard:** Provide a dashboard of tools categorized logically (Lesson Planning, Assessment, Communication, etc.) so that teachers can find what they need quickly. MagicSchool's interface highlights an exemplar for each tool to show how it works [54] . Our platform should similarly provide examples or templates for every feature (e.g., a sample input and output) to guide new users.

- **Guided Prompts:** As noted, **no complex prompting** should be needed from the user's side [47] . The system will handle the heavy lifting of prompt engineering. For instance, instead of a blank chat where a teacher has to guess what to type, the tool interfaces will have labeled fields (like "Enter topic" or "Student name") and options (dropdowns for tone, length, language). This ensures using the AI is as simple as filling a form or answering a few questions. Outputs should be generated in **seconds** to maintain a smooth experience [47] .

- **Multilingual Support:** The UI should support at least English (primary) and possibly other Indian languages in the future. More critically, the **content generation** should handle multiple

languages. For example, a teacher at a Hindi-medium school should be able to get a lesson plan or worksheet output in Hindi. MagicSchool already allows translating outputs into multiple languages with one click [55] . Our platform should include a **"translate output"** option for major languages (Hindi, Bengali, Tamil, etc.), leveraging AI translation models, so that teachers can easily produce materials in their preferred language.

- **Responsive Design:** Many Indian educators might access the platform on mobile devices or tablets (if computers are not available). The interface must be mobile-responsive or offered as a mobile app for Android/iOS. All features should be usable on a smartphone – e.g., a teacher in a rural area might only have a phone; they should still be able to generate a lesson plan on it.

- **Accessibility:** Follow accessibility standards (like proper contrast, screen-reader compatibility, etc.) so that differently-abled educators or students can also use the platform. This includes clear fonts, the ability to zoom interface, and avoiding clutter.

- **Performance:** The system should be optimized to return AI-generated results quickly (ideally within a few seconds for most requests) to keep the workflow efficient. Caching or pre-loading of common requests might be used to speed up responses. As usage scales, the backend should autoscale to maintain responsiveness for potentially thousands of concurrent users.

2. Scalability and Reliability

- The platform should be built on scalable cloud infrastructure to accommodate potentially **millions of users** (MagicSchool serves nearly 6 million educators worldwide [56] , so our target in India could be huge given the size of the education sector). It must handle peak usage times (e.g., evenings when teachers plan lessons, or exam seasons) without slowdowns.

- Use a robust AI model backend (likely via integration with a large language model API or a fine-tuned local model). The architecture should allow swapping/improving the AI model as technology advances, without major changes to the user experience. Initially, existing APIs (OpenAI, etc.) could be used for quick development, with the option to move to dedicated or open-source models for cost-efficiency long-term.

- Ensure **high availability** with minimal downtime. Since teachers might plan late at night or early morning, the service should be operational 24/7. Aim for >99% uptime, and provide a status page (like MagicSchool's app status monitoring [57]).

- Implement proper error handling and fallback. If the AI fails to generate a good answer (which can happen), the system should either retry with adjusted parameters or return a polite message with suggestions (rather than just an error). Users should never feel stranded; always guide them on next steps (e.g., "If the result isn't what you need, try adding more details or contact support").

3. Security and Privacy

- **Data Privacy Compliance:** The platform must handle sensitive data (student information, potentially) with care. It should comply with international standards like **FERPA, COPPA, GDPR** for student data protection [58] . While India's own data protection law is evolving, we should be prepared to meet any local regulations (and certainly follow consent principles for minors' data).

MagicSchool boasts a 93% privacy rating by an independent evaluation [59] – we should strive for similar trust by design.

- **No Personal Data Leakage:** Built-in safeguards should prevent AI outputs from unintentionally revealing personal identifiable information (PII) about students or teachers [60]. For example, if a teacher uploads a class list or student work, that data should not be used to train the AI model beyond that session, and the AI should not include such names or details in unrelated outputs. Privacy agreements (Data Privacy Agreements) can be established with institutions to assure them of data handling practices [60].

- **Secure Architecture:** All user interactions and data transfers must be encrypted (HTTPS, secure APIs). Implement strong authentication (especially if SSO fails over to manual login). Store data (like saved lesson plans or student room interactions) securely, with access controls so that one teacher cannot accidentally access another teacher's private content. For enterprise/school clients, provide options for data isolation as needed.

- **AI Usage Safeguards:** The platform should include **content filtering** on the AI's outputs to avoid inappropriate or harmful content. Since students might use it, it needs a moderation system to prevent profanity, biased or culturally insensitive content, etc. MagicSchool highlights user-friendly alerts to avoid bias and ensure factual accuracy [61]; similarly, our AI should warn the user if an output might contain a factual error or if it's only a draft that needs review. Regular audits of AI outputs will be done to improve the safety and quality of the model's responses.

- **User Access Control:** Especially for student-facing aspects, ensure that students can only access the tools appropriate for them (likely through a class join code or teacher-managed accounts). Teachers should control whether a student can chat with the AI directly or only use certain study tools. This prevents misuse and keeps the AI as a guided learning tool rather than an answer vending machine.

4. Localization and Indian Context Adaptation

This is a **crucial requirement**: the platform must be adapted to India's educational context to maximize its relevance and effectiveness.

- **Curriculum Alignment:** All content generation tools should allow alignment with Indian curricula. For example, when generating a lesson or questions, the teacher could select the board (CBSE, ICSE, or State Board) and grade level, so the AI knows the context. The platform's knowledge base should include **NCERT syllabus and state syllabus frameworks** to ensure generated content is on-point. E.g., if a teacher asks for a lesson on "Trigonometry identities for Class 10 CBSE," the AI should be aware of what Class 10 CBSE students are expected to learn (perhaps by training on or referencing NCERT textbooks). This alignment ensures **standards-based outputs** that teachers can trust [15].

- **Language Support:** India's multilingual classrooms mean the platform should support content generation in **multiple languages**. Initially, **English and Hindi** are a must (as many teachers in government schools teach in Hindi or bilingual). The **Text Translator** tool will instantly translate any generated text into a target language [62], which is helpful for bilingual classrooms or for sending notes to parents in their local language. Over time, adding support for regional languages (like Marathi, Tamil, Telugu, etc.) will broaden the platform's utility. Even the AI

chatbot should ideally handle queries in these languages or mix of English and local language (since teachers might phrase questions in Hinglish, for example).

- **Cultural Relevance:** The AI should produce examples and contexts relatable to Indian students. For instance, a word problem might mention local names or settings (e.g., cricket analogies, Indian festivals) when appropriate, rather than exclusively Western contexts. Ensuring the training data or prompt design includes Indian cultural references will make outputs feel more relevant. Teachers should also be able to specify context (like "make it relevant to rural Indian context" or "use an example from Indian history"), and the AI should comply. This ensures that the tool isn't just global but *speaks to the Indian classroom experience*.

- **Content Moderation for Local Sensitivities:** The platform must be aware of social and cultural sensitivities in India. Avoid content that might inadvertently touch on political, religious, or social issues in a way that could be problematic in a classroom. For example, when generating history content, ensure factual accuracy and neutrality (to align with approved curricula). If a teacher asks something outside the scope (like a contentious historical interpretation), the AI should politely defer or stick to curriculum facts. This may involve tuning the model or having a list of sensitive topics with guidelines.

- **Offline or Low-Bandwidth Use:** Recognizing that not all Indian schools have high-speed internet, the platform should be optimized for low bandwidth. This might mean a **lightweight mode** (minimal graphics, mostly text) and possibly an option to pre-download certain AI modules or have an offline cache for frequently used features (though full offline AI may not be feasible initially). At the very least, ensure the web app loads quickly even on 3G connections and handle intermittent connectivity gracefully (e.g., queue requests if connection drops).

- **Training and Change Management:** Introducing AI in classrooms requires supporting teachers in understanding and trusting the tool. The platform should include **professional development resources** tailored to the Indian education system. For example, short tutorial videos, step-by-step guides, and community forums (possibly in multiple languages) to help teachers learn how to use each tool and share best practices. MagicSchool provides free certification courses and guides for educators to get started [63]; likewise, we should offer webinars or workshops aligned with Indian teacher training norms (perhaps in collaboration with CBSE or DIETs for reach). Emphasize that the AI is **meant to assist, not replace** teachers – aligning with MagicSchool's ethos that "educators are the magic, not the AI" [5]. This messaging will help alleviate concerns (noted by some surveys where ~44% of Indian teachers were concerned about AI's impact on teaching [64]) and position the platform as a teacher's ally.

Conclusion and Future Scope

In summary, the platform will replicate the core functionalities of MagicSchool.ai – providing an **extensive suite of AI tools for educators** – while customizing them for simplicity and the **Indian educational context**. By including lesson generators, assessment creators, communication aids, differentiation tools, and AI assistants, the platform addresses the full spectrum of a teacher's needs from planning to personalization [1] [3]. The **Indian perspective** is woven in through curriculum alignment, multilingual support, and culturally relevant content, ensuring that the tool is immediately useful to Indian teachers overwhelmed by large classes and heavy workloads [4].

The outcome will be a secure, scalable system that **saves teachers time every week**, improves the quality of lesson delivery, and fosters better learning outcomes for students. As educators use these

tools, they can redirect their energy towards interactive teaching and mentoring, which no AI can replace. Over time, as the platform gathers feedback, it can evolve – for example, adding more regional curricula, supporting vocational education content, or integrating with government education portals. The vision is to empower teachers with AI so they can be more creative and effective, ultimately benefiting students across India.

By meeting the requirements outlined above, we aim to deliver a platform that is **"most loved, trusted, and secure" by educators** [65], much like MagicSchool, and truly make teachers' and students' lives easier through the *magic* of AI.

Sources:

- MagicSchool.ai – *Website content and features overview* [14] [27] [40] [30]
- MagicSchool.ai – *MagicSchool for Educators page* (AI tools, benefits) [66] [67] [3]
- MagicSchool.ai – *Pricing & Features (Free vs Plus vs Enterprise)* [58] [50]
- Reuters Press Release – *MagicSchool "Teachers Are Magic" Campaign* (stats on usage and time saved) [1] [65]
- EdTechReview – *MagicSchool.ai funding news* (AI tools for lesson planning, time saved) [68]
- The Economic Times – *Survey of Indian teachers* (workload statistics) [4]

---

[1] [2] [6] [56] [65] MagicSchool AI Launches 'Teachers Are Magic' Nationwide Campaign Featuring Real Teachers | Reuters

https://www.reuters.com/press-releases/magicschool-ai-teachers-are-magic-campaign-2025-08-07/

[3] [5] [7] [8] [11] [47] [54] [57] [63] [66] [67] Teacher AI | MagicSchool

https://www.magicschool.ai/magicschool

[4] NEP: Only 58% school teachers took part in discussions on NEP, 65% overloaded with work: MoE survey - The Economic Times

https://economictimes.indiatimes.com/news/india/only-58-school-teachers-took-part-in-discussions-on-nep-65-overloaded-with-work-moe-survey/articleshow/91813203.cms?from=mdr

[9] [10] [14] [16] [17] [18] [21] [22] [23] [24] [25] [26] [27] [28] [29] [30] [31] [32] [33] [34] [35] [36] [37] [38] [39] [40] [41] [42] [44] [45] [46] [55] [62] Teacher Tools | MagicSchool

https://www.magicschool.ai/magic-tools

[12] [13] [19] [20] [43] [48] [49] [50] [51] [52] [53] [58] [61] Pricing | MagicSchool

https://www.magicschool.ai/pricing

[15] [59] [60] AI for Educators | MagicSchool

https://www.magicschool.ai/

[64] Around 44% of school teachers in India are concerned about the …

https://indiaai.gov.in/news/around-44-of-school-teachers-in-india-are-concerned-about-the-impact-of-ai-on-teaching-and-assessment-survey

[68] MagicSchool.ai Raises $2.4M to Tackle Teacher Burnout With Artificial Intelligence – EdTechReview

https://www.edtechreview.in/news/magicschool-ai-raises-2-4m-to-tackle-teacher-burnout-with-artificial-intelligence/