ProjectTitle

ProjectDocumentation

1. Introduction

- Projecttitle:
  Edu Tutor
  AI:Personalised
  Learning
- Teammember:
  S.Mohanapriya
- Teammember:
  M.Kavipriya
- Teammember:
  M.Sujatha

2. projectoverview

- Purpose:

  The purpose of a Sustainable Smart City Assistant is to empowercitiesandtheirresidentstothriveinamoreeco-consciousand connected urban environment. By leveraging AI and real-time data, the assistanthelpsoptimizeessentialresourceslikeenergy,water,andwaste, while also guiding sustainable behaviors among citizens through personalized tips and services. For city officials, it serves as a decision-making partner—offering clear insights, forecasting tools, and summarizations of complex policies to support strategic planning. Ultimately,thisassistantbridgestechnology,governance,andcommunity engagementtofostergreenercitiesthataremoreefficient,inclusive,and resilient.
- Features:

ConversationalInterface

*KeyPoint:*Naturallanguageinteraction

*Functionality:*Allowscitizensandofficialstoaskquestions,getupdates,and receive guidance in plain language

PolicySummarization

*KeyPoint:*Simplifiedpolicyunderstanding

*Functionality:*Convertslengthygovernmentdocumentsintoconcise, actionablesummaries.

ResourceForecasting
*KeyPoint:*Predictiveanalytics
*Functionality:*Estimatesfutureenergy,water,andwasteusageusinghistorical and real-time data.

Eco-TipGenerator
*KeyPoint:*Personalizedsustainabilityadvice
*Functionality:*Recommendsdailyactionstoreduceenvironmentalimpact based on user behavior.

CitizenFeedbackLoop
*KeyPoint:*Communityengagement
*Functionality:*Collectsandanalyzespublicinputtoinformcityplanningand service improvements.

KPI Forecasting
*KeyPoint:*Strategicplanningsupport
*Functionality:*Projectskeyperformanceindicatorstohelpofficialstrack progress and plan ahead.

AnomalyDetection
*KeyPoint:*Earlywarningsystem
*Functionality:*Identifiesunusualpatternsinsensororusagedatatoflag potential issues.

MultimodalInputSupport
*KeyPoint:*Flexibledatahandling
*Functionality:*Acceptstext,PDFs,andCSVsfordocumentanalysisand forecasting.

StreamlitorGradioUI
*KeyPoint:*User-friendlyinterface
*Functionality:*Providesanintuitivedashboardforbothcitizensandcityofficials to interact with the assistant.

3. Architecture

Frontend(Streamlit):

The frontend is built with Stream lit, offering an interactive web UI with multiple pages including dashboards, file uploads, chat interface, feedback forms,andreportviewers.Navigationishandledthroughasidebarusingthe stream lit-option-menu library. Each page is modularized for scalability.

Backend(FastAPI):

FastAPIserves as thebackendRESTframeworkthatpowers APIendpoints for document processing, chat interactions, eco tip generation, report creation, andvectorembedding.Itisoptimizedforasynchronousperformanceandeasy Swagger integration.

LLMIntegration(IBMWatsonxGranite):

Granite LLM models from IBM Watsonx are used for natural language understandingandgeneration.Promptsarecarefullydesignedtogenerate summaries, sustainability tips, and reports.

VectorSearch(Pinecone):

UploadedpolicydocumentsareembeddedusingSentenceTransformersand storedinPinecone.Semanticsearchisimplementedusingcosinesimilarityto allow users to search documents using natural language queries.

MLModules(ForecastingandAnomalyDetection):

LightweightMLmodelsareusedforforecastingandanomalydetectionusing Scikit-learn.Time-seriesdataisparsed,modeled,andvisualizedusingpandas and matplotlib.

4. SetupInstructions

Prerequisites:

- Python3.9orlater
- pipandvirtualenvironmenttools
- APIkeysforIBMWatsonxandPinecone
- Internetaccesstoaccesscloudservices

InstallationProcess:

- Clonetherepository
- Installdependenciesfromrequirements.txt
- Createa.envfileandconfigurecredentials
- RunthebackendserverusingFastAPI
- LaunchthefrontendviaStreamlit
- Uploaddataandinteractwiththemodules

5. FolderStructure

app/–ContainsallFastAPIbackendlogicincludingrouters,models,and integration modules.

app/api/–SubdirectoryformodularAPIrouteslikechat,feedback,report,and document vectorization.

ui/–ContainsfrontendcomponentsforStreamlitpages,cardlayouts,and form UIs.

smart_dashboard.py–EntryscriptforlaunchingthemainStreamlit dashboard.

granite_llm.py–HandlesallcommunicationwithIBMWatsonxGranitemodel including summarization and chat.

document_embedder.py–Convertsdocumentstoembeddingsandstoresin Pinecone.

kpi_file_forecaster.py–Forecastsfutureenergy/watertrendsusingregression.

anomaly_file_checker.py – Flags unusual values in uploaded KPI data.

report_generator.py– ConstructsAI-generatedsustainabilityreports.

6. RunningtheApplication

Tostarttheproject:

- LaunchtheFastAPIservertoexposebackendendpoints.
- RuntheStreamlitdashboardtoaccessthewebinterface.
- Navigatethroughpagesviathesidebar.
- UploaddocumentsorCSVs,interactwiththechatassistant,andview outputs like reports, summaries, and predictions.

- Allinteractionsarereal-timeandusebackendAPIstodynamically update the frontend.

Frontend(Streamlit):

The frontend is built with Stream lit, offering an interactive web UI with multiple pages including dashboards, file uploads, chat interface, feedback forms,andreportviewers.Navigationishandledthroughasidebarusingthe stream lit-option-menu library. Each page is modularized for scalability.

Backend(FastAPI):

FastAPIserves as thebackendRESTframeworkthatpowers APIendpoints for document processing, chat interactions, eco tip generation, report creation, andvectorembedding.Itisoptimizedforasynchronousperformanceandeasy Swagger integration.

7. APIDocumentation

BackendAPIsavailableinclude:

POST/chat/ask–AcceptsauserqueryandrespondswithanAI-generated message

POST/upload-doc–UploadsandembedsdocumentsinPinecone

GET/search-docs–Returnssemanticallysimilarpoliciestotheinputquery

GET/get-eco-tips–Providessustainabilitytipsforselectedtopicslikeenergy, water, or waste

POST/submit-feedback–StorescitizenfeedbackforlaterrevieworanalyticsEachendpointistestedanddocumentedinSwaggerUIforquickinspection and trial during development.

8. Authentication

eachendpointistestedanddocumentedinSwaggerUIforquickinspection and trial during development.

Thisversionoftheprojectrunsinanopenenvironmentfordemonstration. However, secure deployments can integrate:

- Token-basedauthentication(JWTor API keys)
- OAuth2withIBMCloudcredentials
- Role-basedaccess(admin,citizen,researcher)
- Plannedenhancementsincludeusersessionsandhistorytracking.8. Authentication

9. UserInterface

Theinterfaceisminimalistandfunctional,focusingonaccessibilityfornon- technical users. It includes:

Sidebarwithnavigation

KPIvisualizationswithsummarycards

Tabbedlayoutsforchat,ecotips,andforecasting

Real-time form handling

PDFreportdownloadcapability

Thedesignprioritizesclarity,speed,anduserguidancewithhelptextsand intuitive flows.

10. Testing

Testingwasdoneinmultiplephases:

UnitTesting:Forpromptengineeringfunctionsandutilityscripts API

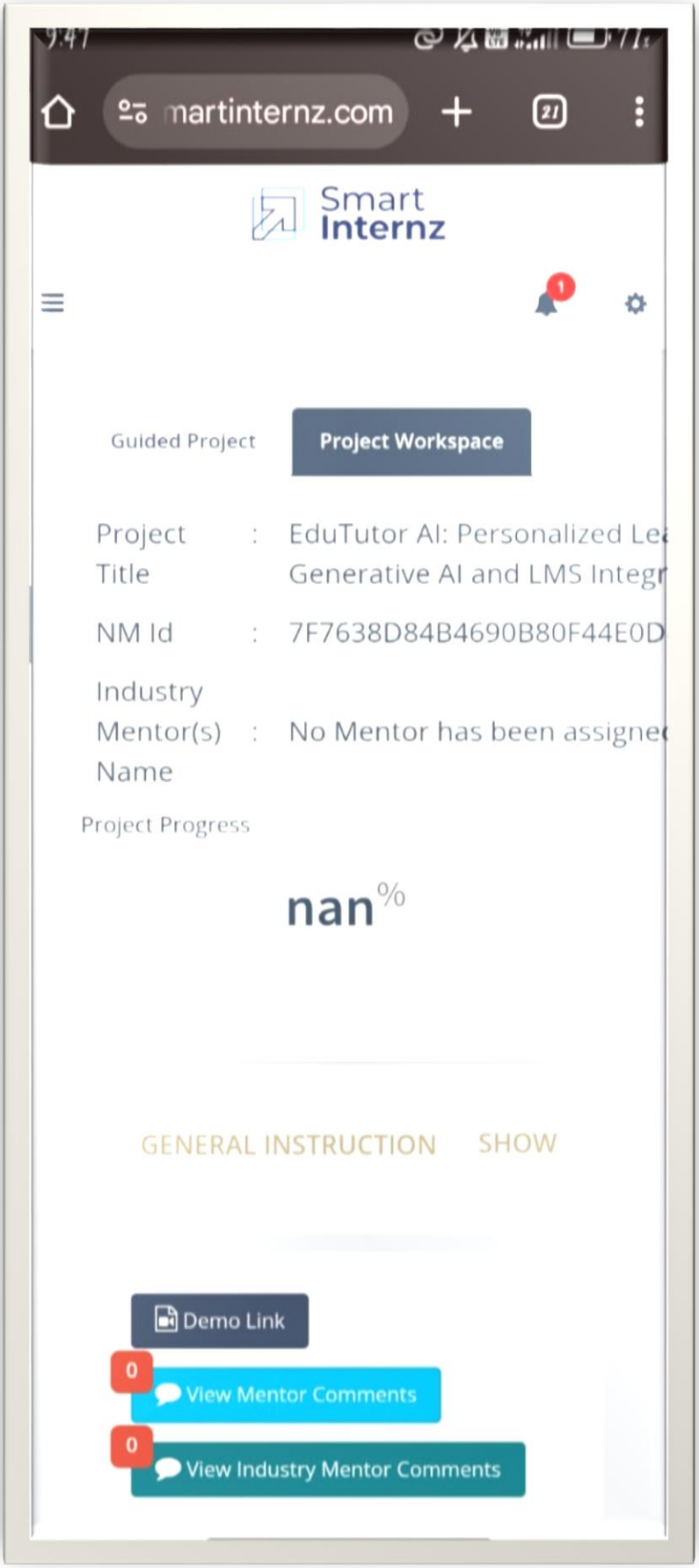
Testing: Via Swagger UI, Postman, and test scripts

ManualTesting:Forfileuploads,chatresponses,andoutputconsistency Edge

Case Handling: Malformed inputs, large files, invalid API keys

Eachfunction was validatedto ensure reliability in both ofline andAPI-connected modes.

11. **Screenshots** :



12. **KnownIssues: many issues…**

13. **Futureenhancement| Enhancement Area**

| What It Could Do
| Why It Helps / Trend Backing
|

14. | -- | --- | -- |

15. | **Personalisation & Adaptive Learning Paths** | • Dynamically adapt difficulty, pace, style (e.g. more visuals vs. text) based on student performance.
 • Diagnose weak areas and provide targeted practice.
 • Allow learners to choose learning modes (e.g. quizzes, simulations, video, reading) depending on their preferences. | Helps engagement, retention. Teachers free up time; students feel learning is relevant. Forecasts for EdTech show personalised learning as key. ([Momen][1]) |

16. | **Intelligent Tutoring & Feedback** | • Use generative AI to generate explanations, hints, solutions.
 • Provide timely, meaningful feedback on essays or open‑ended responses.
 • Use dialogue‑based tutoring (conversational agents) for 1‑on‑1 help. | Learners benefit from scaffolding; can scale individual tutoring. Research on generative AI + ITS shows promise. ([arXiv][2]) |

17. | **Immersive & Multimodal Learning** | • Include AR / VR simulations (labs, historical reenactments, experiments) to let students experience concepts.
 • Use multimodal content — audio, video, text, interactive visuals.
 • Use speech recognition / pronunciation tools (especially for language learning). | Multimodal, immersive content enhances understanding & retention. Trend shows growing use of VR/AR in EdTech. ([ED-EX][3]) |

18. | **Predictive Analytics & Early Warning** | • Predict students at risk of falling behind (e.g. via performance trends, engagement metrics) and alert teachers or suggest interventions.
 • Tailor future content based on predicted gaps. | Helps reduce dropout / failure; gives proactive support rather than reactive. ([Momen][1]) |

19. | **Accessibility & Inclusion** | • Support for students with disabilities (text to speech, speech to text, visual aids, customizable interfaces).
 • Multilingual support, real time translation, adapt to local contexts.
 • Low bandwidth / offline modes for areas with poor connectivity. | Essential for equity; necessary in many regions. Inclusive tools broaden reach. ([Enterprise League][4]) |

20. | **Automating Routine / Administrative Tasks** | • Auto-grading of objective and some subjective assignments.
 • Automatically generate quizzes, lesson plans, flashcards.
 • Generate summaries of readings, concept maps.
 • Allow teachers to spend less time on prep & feedback and more on mentoring. | Improves teacher efficiency; more time for high-value tasks. Widely seen in EdTech trends. ([ED-EX][3]) |

21. | **Gamification & Motivation** | • Reward systems, points, badges, leaderboards.
 • Game-like challenges, quests, interactive simulations.
 • Social / peer challenges / collaborative learning. | Keeps engagement high; modern learners often respond well to interactive / game-like features. ([ED-EX][3]) |

22. | **Ethics, Transparency, Explainability** | • Make AI decisions (e.g. why certain recommendation / feedback) explainable to students & teachers.
 • Ensure fairness, low bias in assessments &

recommendations.
 • Data privacy, consent, secure handling of personal/student data.
| Critical for trust & wide adoption; important in research and regulation. ([arXiv][5])
|

23. | **Offline & Edge Functionality** | • Capabilities that work without needing constant internet (e.g. cached lessons, downloadable modules).
 • Lightweight client side models for low resource devices.
| Ensures reach to areas with connectivity issues; reduces latency; more accessible.
|

24. | **Lifelong Learning / Skills / Career Alignment** | • Include modules to build soft skills (critical thinking, collaboration, communication).
 • Offer career pathways, guidance, credentials / micro-credentials.
 • Align content to current job market / future skills.
| Helps with employability; EdTech trend toward microcredentials. ([Amazon Web Services, Inc.][6])
|

25. [1]: https://momen.app/blogs/ai-ed-tech-market-forecast-growth-opportunities-2025-2034/?utm_source=chatgpt.com "AI EdTech Market Forecasts and Emerging Opportunities in 2025 and Beyond"

26. [2]: https://arxiv.org/abs/2410.10650?utm_source=chatgpt.com "Generative AI and Its Impact on Personalized Intelligent Tutoring Systems"

27. [3]: https://ed-ex.com/en/blog/global-edtech-trends-2025-how-ai-is-reshaping-learning?utm_source=chatgpt.com "Global EdTech Trends 2025: How AI Is Reshaping Learning"

28. [4]: https://enterpriseleague.com/blog/top-trends-of-ai-in-edtech/?utm_source=chatgpt.com "Top 10 trends of AI in edtech in 2025"

29. [5]: https://arxiv.org/abs/2303.13379?utm_source=chatgpt.com "Practical and Ethical Challenges of Large Language Models in Education: A Systematic Scoping Review"
30. [6]: https://aws.amazon.com/blogs/publicsector/6-edtech-ai-trends-how-artificial-intelligence-is-reshaping-education/?utm_source=chatgpt.com "6 EdTech AI trends: How artificial intelligence is reshaping education | AWS Public Sector Blog"