Project Report

**EduTutorAI:PersonalizedLearningwithGenerativeAIandLMS Integration**

# INTRODUCTION

## 1.1 ProjectOverview

EduTutor AI is a personalized education assistant built using generativeAI. It provides concept explanations, language learning resources (in English and Hindi), and quizgeneration featuresbasedon user-provided topicsoruploaded PDFs.The system is developed using Python, integrated with the IBM Granite 3.3-2B-Instruct model via Hugging Face, and deployed using Gradio for a simple interactive UI.

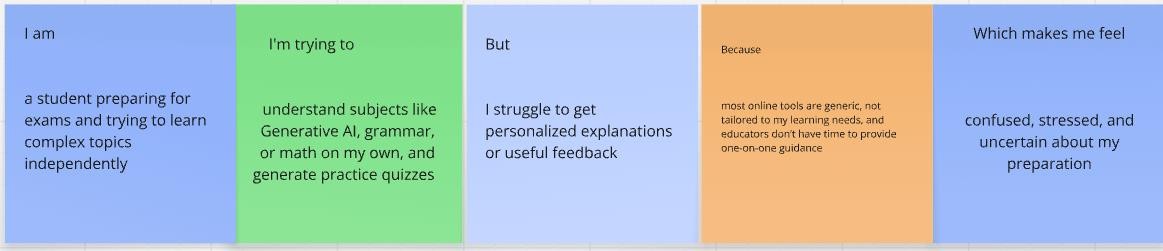
## 1.2 Purpose

The purpose of EduTutorAI is to bridge the gap in personalized learning by deliveringinstantAI-powerededucationalsupporttolearners.Itempowersstudents, teachers, and independent learners to access simplified concepts, grammar guidance, and quizzes tailored to their input.

# IDEATION PHASE

## 2.1 ProblemStatement

Students often lack access to immediate, tailored explanations of academic concepts.Teachersspendsignificanttimepreparingtestsandassessments.Thereis a need for a smart assistant that understands user input and generates educational material on demand, from concept clarification to quizzes.



## EmpathyMap Canvas

**Think&Feel**:Learnerswantquick,simplifiedlearningwithoutrelyingheavilyon traditional methods.

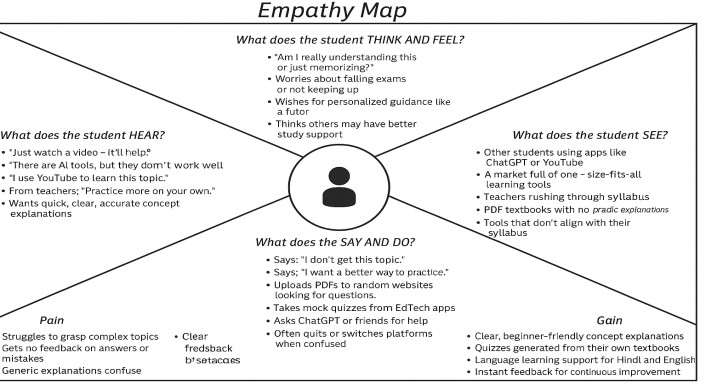
**See**:Overwhelmingresources,confusingtextbooks,toomany tools.

**Say&Do**:Ask forhelp,searchonline,preferinteractiveformats.

**Hear**:"Thistopicishard","TryYouTubeor ChatGPT".

**Pain**:Time-consumingsearch,inconsistentcontent,lackofquizzes.

**Gain**:Singletool offeringexplanations,languagelearning,andMCQs.



## Brainstorming

Key ideas:

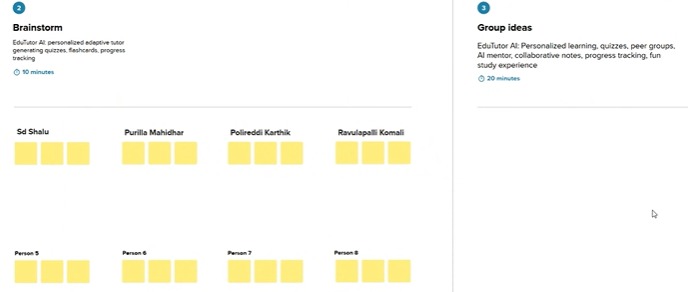
ProvideAI-basedexplanationsinsimplelanguage

Supportmultiplelanguages

Accept books/PDFs togeneratequestions

Providelogin and usersessionmanagement

Avoidneed fordeeptechknowledge to usethe app



3.REQUIREMENTANALYSIS : EduTutor AI: personalized learning, quizzes, collaboration, progress tracking, secure, scalable.

## 3.1 CustomerJourney Map

Stages:

Awareness→SeesAItoolon search/social

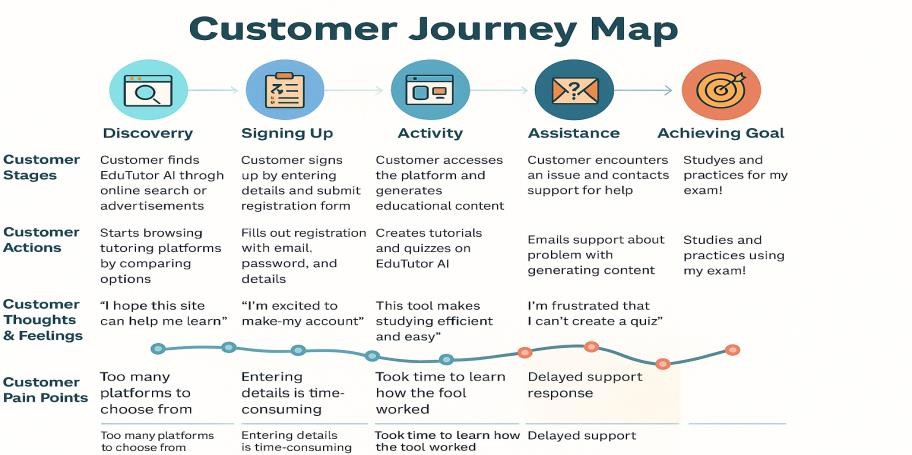
Consideration→TestsconceptandPDFquizfeature

Onboarding→Registersandinputsconcepts

Engagement→Usesregularlyfor varioussubjects

Retention→ Returnsoftenforpractice

Referral→Recommendstopeersafteruseful experience



## SolutionRequirement

The systemmustsupport:

Conceptinput andprompt processing

Languageselection(English/Hindi)

PDFreadingandquestion generation

Topic-basedquizgeneration

Sessiontracking

Basicloginandregistration

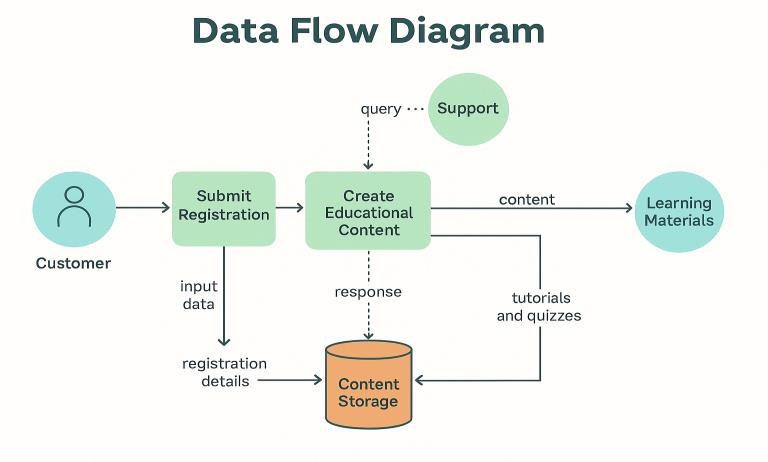
## DataFlowDiagram

User→GradioUI→Backend processing

→ Prompt sent toHugging Face model

→ Responsedisplayedbackto user

→ Sessionstrackedin Pythondictionary



## TechnologyStack

Python

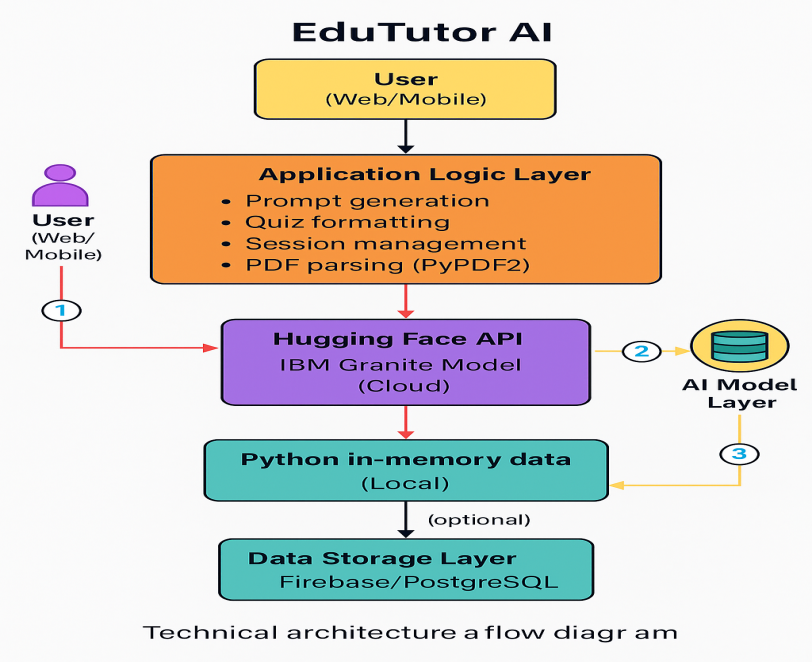
Gradio(UI)

HuggingFaceTransformers(API)

PyPDF2(PDFparsing)

IBMGranite3.3-2B-Instruct(LLM)

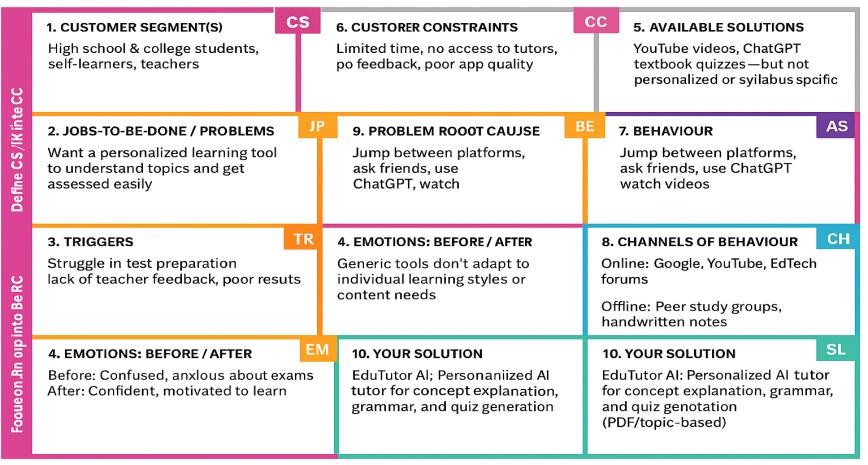
GoogleColab/Jupyter (execution environment)



# PROJECT DESIGN

## ProblemSolution Fit

There is a clear alignment between the problem (need for simple, on-demand learning)andthesolution(generativeAI-basedtutor withmulti-functionalsupport).



## ProposedSolution

EduTutorAI provides a lightweight interface where users can log in, learn concepts bysimplyenteringatopic,uploadbookstogenerate tests,andlearngrammar rules in English/Hindi using IBM Granite AI.

## SolutionArchitecture

**Frontend**: Gradio Blocks UI

**Backend**:Pythonprocessinglogic

**ModelAPI**:HuggingFace(Granite 3.3-2B)

**FileHandler**:PyPDF2

**SessionTracker**:Pythondictionary

**Optional**:ExtendabletoFirebaseorLMS integration

# PROJECTPLANNING&SCHEDULING

## ProjectPlanning

**Sprint1(5Days)**

Userloginsystem

ConceptexplanationusingAI

Languageselectionlogic

Basic sessionmanagement

## Sprint2(5Days)

PDFupload&quizgeneration

Topic-basedquizcreation

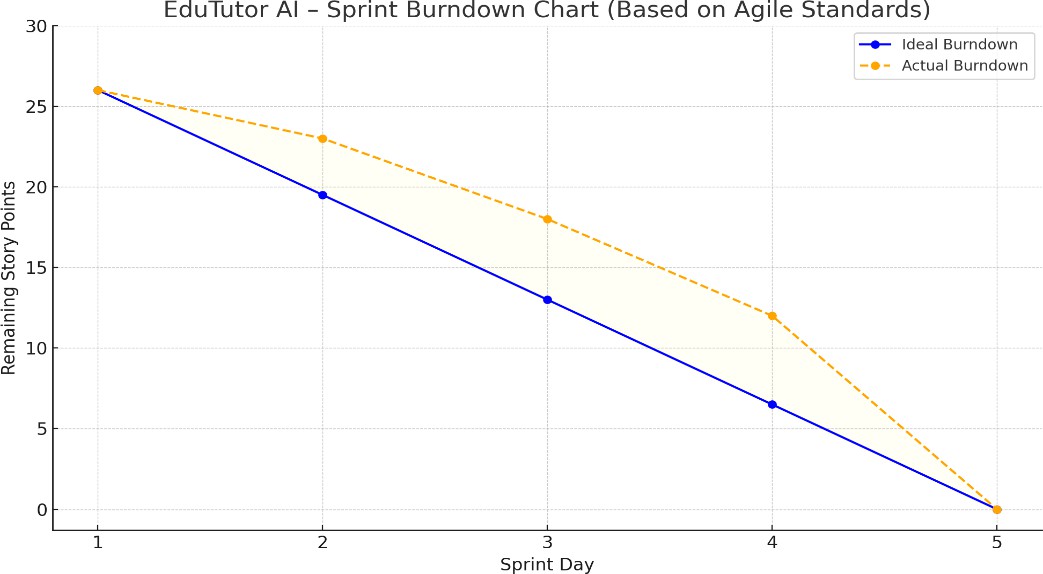
GradioUIintegration

Final testinganddemosetup

## TotalStoryPoints:26

**TeamVelocity**:13pointspersprint

**BurndownChart**: Demonstratesconsistenttaskcompletion acrosssprints



# FUNCTIONALANDPERFORMANCETESTING

## PerformanceTesting

Responsetime forquizgeneration <4 seconds

MultiplePDFuploadshandledwithoutcrash

Model respondswithinacceptabletimeunderload

Loginandregistrationsystembehaves asexpected

# RESULTS

## OutputScreenshots

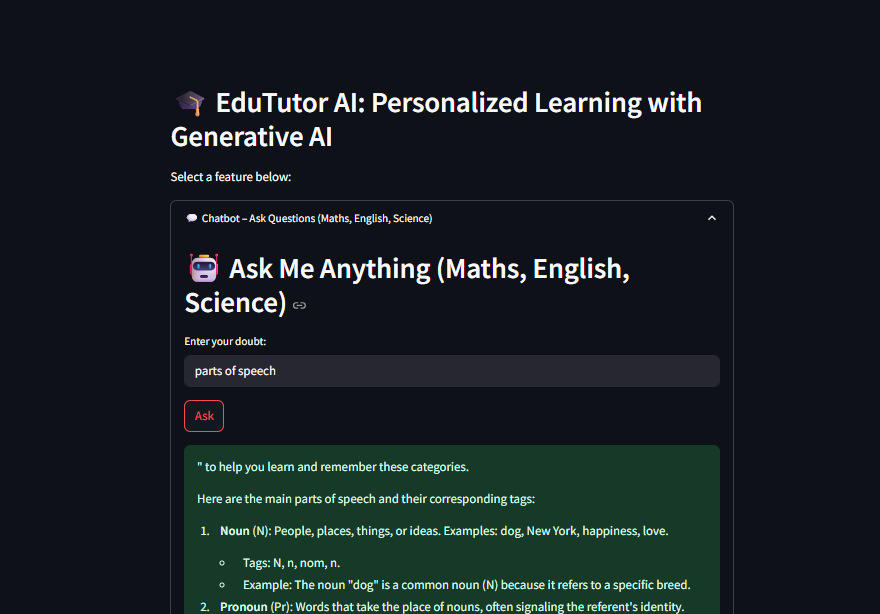
**ConceptOutput**:Clearexplanationforentered topic

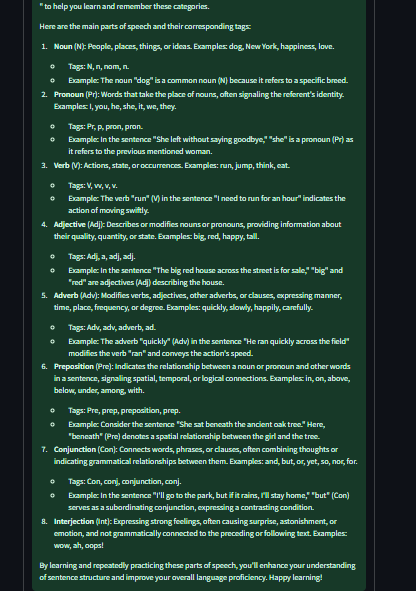
**LanguageOutput**:Grammarpoints, parts ofspeech

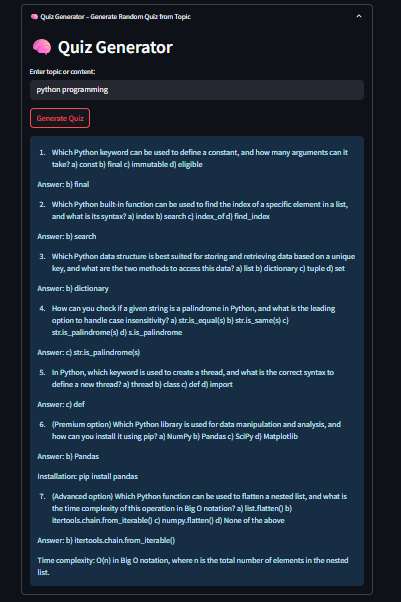
**QuizOutput**:MCQs frombothtopicandPDF content

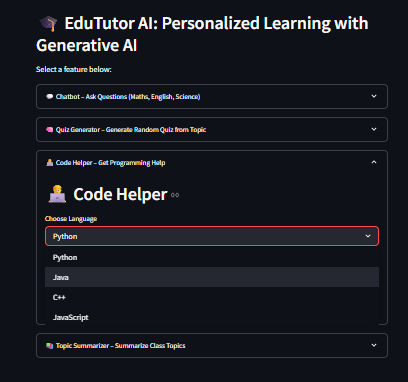
Interfaceisclean,responsive,anduser-friendly

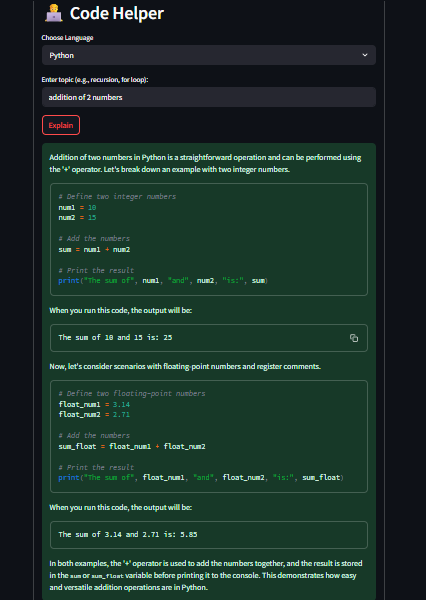


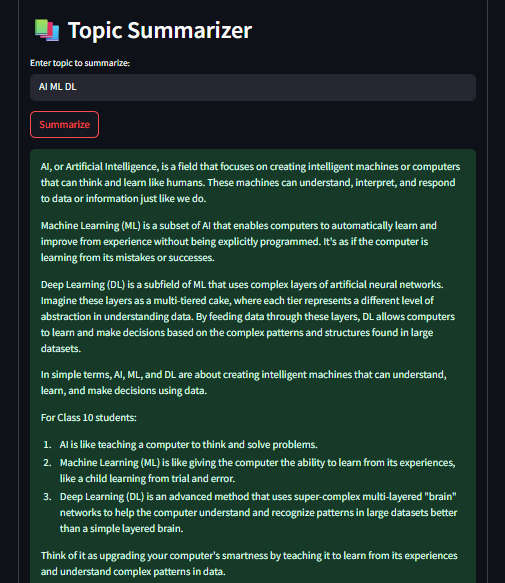












# ADVANTAGES&DISADVANTAGES

## Advantages:

* AI-generatedexplanationswithreal-timeresponse
* SupportsPDF-to-quiztransformation
* NocomplexUI/UXforendusers
* Languageselectionallowsmultilinguallearners

## Disadvantages:

* Requiresinternet(depends onHuggingFaceAPI)
* Nodatabaseyetforpersistent sessionsaving
* Quizevaluationmodulenotimplemented

# 9. CONCLUSION

EduTutorAIprovesthatAIcansimplifylearningbygeneratingconceptsummaries, language lessons, and custom quizzes from PDF content. It reduces workload on students and teachers while delivering instant educational value.

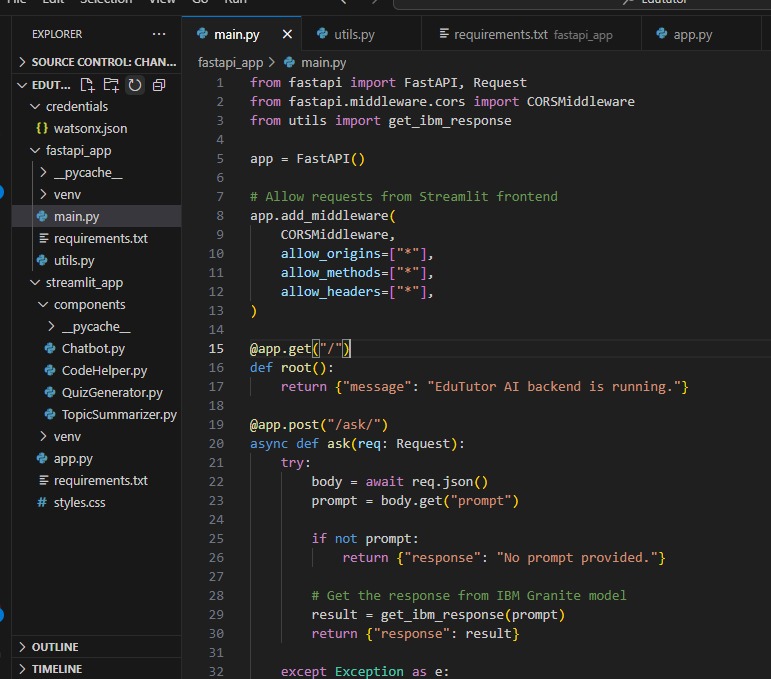
# FUTURESCOPE

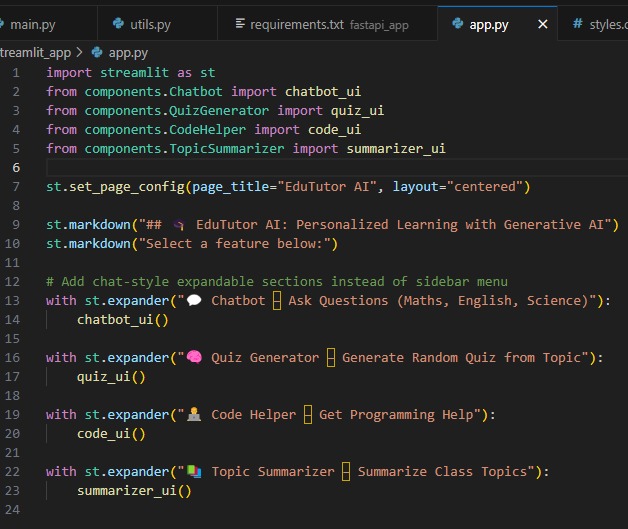
ConnecttoLMSplatforms(like Moodle,GoogleClassroom).

* Addanswerevaluationand quizscoring
* PersistdatausingFirebase/PostgreSQL
* Supportvoiceinputs usingSTTmodels
* Add analyticsandprogresstracking for learners

# APPENDIX

* **SourceCode**:Python script/GoogleColabNotebook





* **GitHub&ProjectDemoLink**: GitHub link:

<https://github.com/Mahidhar27/EdututorAI>

Demo video link: <https://drive.google.com/file/d/1x_MSNYBD9BwSRuH3z8BazQMSpmiTQmzC/view?usp=sharing>