**TECH TUTOR PERSONALIZED AI LEARNING ASSISTANCE FOR TECH ENTHUSIASTS**

**INTRODUCTION:**

Today's technology has become an unavoidable part of the passage of time. Technology has not only changed people's lifestyles but has also changed how we work, learn, and interact. Various kinds of innovations appear all the time, making our activities and work more practical and effective. A more recent technological development is the emergence of the term artificial intelligence which is abbreviated as AI (artificial intelligence) which is currently starting to steal attention as a tool to act like humans. In its development, artificial intelligence has also penetrated the world of education. AI systems allow people to learn with the help of education assistants such as bots. The development of the times requires the world of education to adapt to technological developments to improve the quality of education, especially the adjustment of information and communication technology. Digital learning content that is developing today can be presented thanks to the application of AI. Thick textbooks can now be presented into content that is more concise, easier to read and understand by students, such as study guides, material summaries, or short notes. AI as a pillar of the industrial revolution 4.0 plays a central role in facilitating the learning process mediated by technology. Artificial Intelligence (AI) is the process of modelling human thinking and designing a machine so that it can behave like humans or other terms called cognitive tasks, namely how machines can learn automatically from programmed data and information. Artificial intelligence can also be interpreted Artificial intelligence or AI is one part of computer science that makes machines (computers) able to do work as and as well as humans do. The use of Artificial Intelligence consciously or not we have applied it in everyday life. Many applications have implemented artificial intelligence as an advantage of these applications. Artificial intelligence technology referred to here refers to machines that can think, weigh the actions to be taken, and can make decisions as humans do. Artificial intelligence (AI) is currently being developed on a large scale so that this technology will imitate and even take over the work normally done by humans. Based on the definition of AI, it was created to be able to act like humans in the form of programs and robots. Of course, to facilitate human work. Even various digital platforms have used AI as a part of it. AI is used to make things easier for humans to do. Several technology companies have implemented AI including Amazon, Facebook, Microsoft, and Google. Artificial intelligence or (AI) is a technology where machines can learn and understand logic like humans. This technology is said to be able to help simplify human life which is very complex (Fitria, 2021a). AI itself works by combining the presence of several data, iterative processing, and intelligent algorithms. This allows the software to learn automatically from patterns or features in the data. AI can also be said to be a very broad field of study. The scope of theories, methods, technologies, and subfields that exist in AI is very wide, including machine learning, neural networks, cognitive computing, computer vision, and scientific language processing. The role of Artificial Intelligence (AI) technology is increasingly evident in various sectors, including the education sector. The presence of AI technology has transformed the educational curriculum, especially in the fields of technology, science, mathematics, and engineering. But AI will also change the face of the world of education as a whole. One technology that has recently received attention is Artificial Intelligence (AI). This technology has an important role in facilitating various job functions, including in the field of education. AI can also be implemented in the world of education. Teachers/lecturers can understand student needs more easily and more deeply (Fitria, 2021b). The students also can learn according to their needs without encountering difficulties. Artificial Intelligence (AI) is believed to be able to help humans learn better and achieve educational goals more effectively. So, it's not surprising that currently many AI-based innovations and breakthroughs are being and will be applied to support the learning process to make it more practical and effective. So, when AI is present in the education sector, which then raises concerns by teachers, it is a challenge that must be faced so that the existence of education continues. Some of the strong beliefs that teachers cannot be replaced by AI are teacher collaboration with AI in the implementation of learning. Teachers need to have skills in utilizing science and technology (Science and Technology). So that teachers can take advantage of AI in terms of completing school administration such as making lesson plans, student attendance lists, reporting student learning outcomes, making learning media and learning resources. The researcher is interested to investigate Artificial Intelligence (AI). Therefore, the objective f this study is to investigate Artificial Intelligence (AI) in education, especially in the teaching and learning process.

**1.1 Objective of the project:**

An AI-based Question Answer Generation and Evaluation System uses advanced NLP and machine learning to create and assess questions and answers automatically. It generates a variety of relevant questions from text, supporting effective testing and personalized learning. The system evaluates answers with accurate algorithms, offering detailed feedback and scores. This technology ensures consistent, efficient, and scalable assessments. It improves educational and training environments, enhancing learning experiences and outcomes.

**2. LITERATURE SURVEY:**

**“An Intelligent Tutoring System for Teaching English Grammar”,**

**Abu Ghali, M. J., Abu Ayyad, A., Abu-Naser, S. S., & Abu Laban, M. (2018)**

Education sector in the world takes the largest part from the other sectors, because of this; all countries are interested in the field of education. If we look at learning English language is the third most common languages in the world. Also, IS the internationally dominant in the telecommunications, science and radio, aviation, entertainment, read and diplomatic language as most of the areas of work now taught in English. In this paper, we describe an intelligent tutoring system to help students to help students learn English language grammar easily and smoothly. Therefore, AI experts developed tools for improve learning ways under the name Intelligent Tutoring System. The Intelligent Tutoring System (ITS) is a computer system that offers an instant, adapted instruction and customized feedback to students without human teacher interference. System adapts with all the individual differences of students and begins gradually with students from easier to harder level. The intelligent tutoring system was given to a group of students all age groups to try it and to see the impact of the system on students.

**“Efficient Artificial Intelligence-Teaching Assistant Based on ChatGPT”,**

**Yiqian Chen,** **Hanxi Deng, (2020)**

Due to the real problem of “insufficient basic knowledge in interdisciplinary fields” that students may face when studying interdisciplinary courses and learning interdisciplinary knowledge, this study proposes an “Artificial Intelligence (AI)-Teaching Assistant (TA) App” system. The proposed system combines a Question-and-Answer (QA) system, an expansion mechanism, a cache mechanism, and an error correction mechanism to assist students in their learning process by AI-based QA system. Students can ask “AI-TA App” questions through voice and the App can quickly provide corresponding answers to help them learn. For providing an efficient “AI-TA App,” this study proposes a cache mechanism. A neural network-based QA system is built in the “AI-TA App” server and database to enable the QA system to respond to students' questions quickly; experimental results show that the average response time using the cache mechanism is 0.001 seconds, while the average response time using ChatGPT is 15.652 seconds, indicating that the proposed cache mechanism in this study can effectively improve system efficiency. Furthermore, this study also proposes an expansion mechanism. By calling the ChatGPT APIs, the expansion mechanism expands the question bank and improves the accuracy and robustness of the QA system; experimental results show that the accuracy without using the expansion mechanism is 25.45%, while the accuracy can be improved to 99.09% by using the expansion mechanism.

**“Application of virtual reality teaching method and artificial intelligence technology in digital media art creation”**

**Y. Gong,2021**

With the progress and innovation of science and technology, virtual reality technology emerges as the times require, and its application in digital media art creation is more and more extensive. The application of virtual reality technology in digital media art creation helps to enrich artistic creation means and improve the quality of artistic creation process. This paper first gives the application of virtual reality teaching method in digital media art creation. Virtual reality education has obvious characteristics of vividness and autonomy. In the teaching mode and teaching design, it emphasizes the comprehensive improvement of teachers and students' teaching experience, so as to drive teachers and students to separate from the traditional teaching concept and mode, and achieve twice the teaching effect with half the effort. Then, it gives the technical advantages of artificial intelligence technology for digital media art creation when using virtual reality teaching method.Finally, we give the application of virtual reality and artificial intelligence technology in the teaching of digital media art creation. The experimental results show that the virtual reality teaching method, which combines the advantages of artificial intelligence algorithm, has potential application value and good practical significance in the teaching of digital media art creation.

**“A proposed collaborative framework by using artificial intelligence-internet of things (AI-loT) in COVID-19 pandemic situation for healthcare workers,”**

**S. Kumar, R. D. Raut and B. E. Narkhede,2020**

The COVID-19 epidemic has created a global fear for humanity. Despite technological advancement, the use of face masks, hand gloves, and sanitizers are the only available preventives measures to stop the spread and transmission of coronavirus. Healthcare providers and medical staffs are always at risk of infection as they deal with the infected person at the frontline. The safety of health care workers is essential as the number of infected health care works is increasing day by day. A detailed literature review is done on a previous epidemic outbreak and COVID-19 by using suitable keywords on SCOPUS. During the analysis of literature, several challenges were found that restrict the operation of the health care workers and cause safety issues for them. This research paper aims to identify the challenges faced by healthcare sectors in dealing with this epidemic outbreak and to propose the potential solutions in terms of popular technologies like AI and IoT by doing a critical analysis of literature. The study contributes by identifying the issues and categorizing them into physical, operational, resource-based, organizational, technological, and external health care challenges. Further, suggestions and future research direction concerning epidemic control are explained for practitioners and academicians.

**“Medical students need artificial intelligence and machine learning training”,**

**A. Pucchio, E. A. Eisenhauer and F. Y. Moraes, 2021**

As the information age wanes, enabling the prevalence of the artificial intelligence age; expectations, responsibilities, and job definitions need to be redefined for those who provide services in healthcare. This study examined the perceptions of future physicians on the possible influences of artificial intelligence on medicine, and to determine the needs that might be helpful for curriculum restructuring.

**“Learning design to support student-AI collaboration: perspectives of leading teachers for AI in education,”**

**J. Kim, H. Lee and Y. H. Cho,2022**

Preparing students to collaborate with AI remains a challenging goal. As AI technologies are new to K-12 schools, there is a lack of studies that inform how to design learning when AI is introduced as a collaborative learning agent to classrooms. The present study, therefore, aimed to explore teachers’ perspectives on what (1) curriculum design, (2) student-AI interaction, and (3) learning environments are required to design student-AI collaboration (SAC) in learning and (4) how SAC would evolve. Through in-depth interviews with 10 Korean leading teachers in AI in Education (AIED), the study found that teachers perceived capacity and subject-matter knowledge building as the optimal learning goals for SAC. SAC can be facilitated through interdisciplinary learning, authentic problem solving, and creative tasks in tandem with process-oriented assessment and collaboration performance assessment. While teachers expressed instruction on AI principles, data literacy, error analysis, AI ethics, and AI experiences in daily life were crucial support, AI needs to offer an instructional scaffolding and possess attributes as a learning mate to enhance student-AI interaction. In addition, teachers highlighted systematic AIED policy, flexible school system, the culture of collaborative learning, and a safe to fail environment are significant. Teachers further anticipated students would develop collaboration with AI through three stages: (1) learn about AI, (2) learn from AI, and (3) learn together. These findings can provide a more holistic understanding of the AIED and implications for the educational policies, educational AI design as well as instructional design that are aimed at enhancing SAC in learning.

**“How I use ChatGPT responsibly in my teaching,”**

**H. Yang,2023**

By now, you’ve probably heard all the hubbub about ChatGPT, the artificial intelligence chatbot. “Students will never write their own papers again!” or “ChatGPT is going to replace teachers!” But what if we told you that by embracing this tech tool, you could make your own life as a teacher a little easier? It’s true. Like any form of technology, you and your students need to learn the right way to use it. But once you do, AI tech like ChatGPT really can work for teachers. Read on to learn important dos and don’ts of using ChatGPT, plus our favourite ways teachers can use it as a teaching tool in the classroom. (Oh, and by the way, ChatGPT didn’t write this post. We used it to generate the queries you see in the images, but all the text was written by a real person and represents our real opinions. Plus, we came up with a lot more ideas than the bot!)

**“An augmented reality question answering system based on ensemble neural networks,”**

**C. -H. Chen, C. -L. Wu, C. -C. Lo and F. -J. Hwang,2017**

This paper proposes a classification algorithm based on ensemble neural networks. In the training phase, the proposed algorithm uses a random number of training data to develop multiple random artificial neural network (ANN) models until those ANN models converge. Those models with lower accuracy than the threshold are filtered out. The remaining highly accurate models will be used to predict the output in the testing phase. Meanwhile, the accuracy of ANN models is presented as a weighting value in the testing phase. In the testing phase, the testing data are loaded into the selected ANN models to predict the output class. The output values are multiplied by the corresponding weighting values of ANN models. Then the weighted average of the outputs can be obtained. Finally, the predicted output is converted into the predicted class. We design an augmented reality question answering system (AR-QAS) applying and implementing the proposed algorithm on mobile devices. AR-QAS offers an interactive user interface and automatically replies according to user's queries. By comparing with the logistic regression method and the ANN method, the experiment results demonstrate that the proposed algorithm offers the highest accuracy.

**“Adam: A method for stochastic optimization,”**

**D. P. Kingma and J. Ba,2014**

We introduce Adam, an algorithm for first-order gradient-based optimization of stochastic objective functions. The method is straightforward to implement and is based an adaptive estimate of lower-order moments of the gradients. The method is computationally efficient, has little memory requirements and is well suited for problems that are large in terms of data and/or parameters. The method is also ap- propriate for non-stationary objectives and problems with very noisy and/or sparse gradients. The method exhibits invariance to diagonal rescaling of the gradients by adapting to the geometry of the objective function. The hyper-parameters have intuitive interpretations and typically require little tuning. Some connections to related algorithms, on which Adam was inspired, are discussed. We also analyse the theoretical convergence properties of the algorithm and provide a regret bound on the convergence rate that is comparable to the best-known results under the online convex optimization framework. We demonstrate that Adam works well in practice when experimentally compared to other stochastic optimization methods.

**“Comparative Study of Artificial Intelligence-Based Teaching With Human Interactive Teaching,”**

**Jain, S., & Alam, M. A. (2020)**

Artificial intelligence applications in higher education have become popular, which promotes various learning programs with skill-based curriculum. These artificial intelligence-enabled systems bring a perspective of global classroom. The fundamental system of training helps the students to associate self-learning habit where they can have selection of activities, which can be utilized for different other courses. The human interactive learning prepares the course material, design curriculum, opportunity of articulation, and contribute to the mind-boggling capabilities of the future. The chapter shall study the impact of artificial intelligence learning and human interactive teaching on the students of a university and analyse the results. The chapter focuses to analyse features of traditional system of learning, and artificial intelligence-based robotic learning associates are winding up increasingly accessible are explored in depth using illustrations and case studies. The chapter portrays the effect of artificial intelligence on the education, which can help the student to overcome troubles and see how to push them, to improve the creative mind of a collectivity, and to structure another educational experience. The chapter also discusses the features of human interactive learning method, which is incorporated in artificial intelligence educational systems but can considered for research in future.

**3. SYSTEM ANALYSIS**

**3.1 Existing System**

In the existing system, early approaches relied on predefined templates and rule-based algorithms for tasks such as generating questions from text and analysing network data. While these methods were straightforward, they often lacked flexibility and adaptability to dynamic changes. Similarly, automated grading systems employed NLP techniques like keyword matching, syntactic analysis, and semantic similarity to evaluate answers against predefined correct responses. Though effective, these methods were limited by their reliance on static rules, making them less efficient in handling complex or evolving patterns.

**Disadvantages**

1. Lack of Flexibility

2. Time-Consuming

3. Limited Accuracy

4. Scalability Issues

5. Inability to Evolve

6. Dependence on Static Rules

**3.2 PROPOSED SYSTEM**

The proposed AI-based question-answer generation and evaluation system leverages advanced AI technologies to enhance efficiency, accuracy, and scalability in education and training. It comprises two main components: Question Generation and Answer Evaluation, developed through stages like data collection, preprocessing, model training, and system integration. An RNN model is employed for question generation, effectively handling sequential data to understand context and create relevant questions. Techniques such as teacher forcing and beam search are used to improve the quality and diversity of generated questions. For answer evaluation, a separate RNN model is trained to analyse responses, compare them with ideal answers using semantic similarity, assign scores based on predefined rubrics, and provide detailed feedback to enhance learning outcomes.

**Advantages**

1. Efficiency
2. Scalability
3. Accuracy
4. Personalization
5. Diversity
6. Consistency

**Modules:**

In this project we have created modules for faculties and students to sign up and login to application. After login faculty will perform below operations

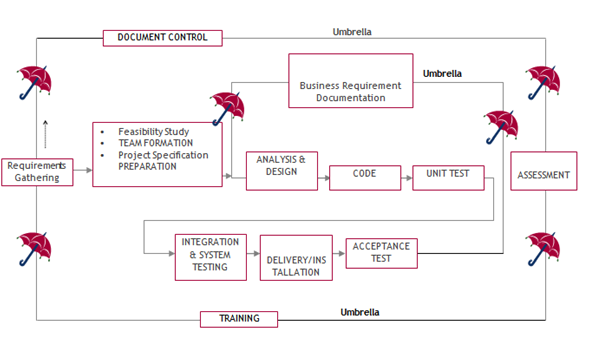
1. Text Summary: Faculty can enter text or upload pdf or word file and then system will generate summary from input file or text
2. Subjective Question: using this module faculty can upload or enter text and then system will generated subjective question and answer from given text
3. Multiple Choice Question: using this module faculty can upload or enter text and then system will generated multiple choice question and answer from given text
4. View Marks: system will apply AI algorithm to evaluate students answers and then assigned marks and those marks can be view by faculty
5. Model Analysis: will display accuracy of model in terms of assigning marks to students answers evaluation which will compare students answers with correct answers generated by AI models

Students can login and perform below options

1. Generate Summary: can generate summary from word, pdf or text
2. Write Exam: can write subjective or multiple choice questions exam
3. View Marks: can view their marks obtained for each assignment.

**3.3. PROCESS MODEL USED WITH JUSTIFICATION**

**SDLC (Umbrella Model):**

****

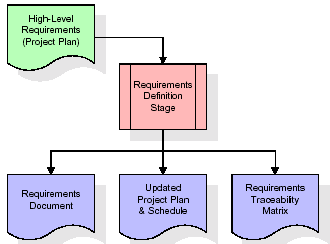
SDLC is nothing but Software Development Life Cycle. It is a standard which is used by software industry to develop good software.

**Stages in SDLC:**

* Requirement Gathering
* Analysis
* Designing
* Coding
* Testing
* Maintenance

**Requirements Gathering stage:**

The requirements gathering process takes as its input the goals identified in the high-level requirements section of the project plan. Each goal will be refined into a set of one or more requirements. These requirements define the major functions of the intended application, define operational data areas and reference data areas, and define the initial data entities. Major functions include critical processes to be managed, as well as mission critical inputs, outputs and reports. A user class hierarchy is developed and associated with these major functions, data areas, and data entities. Each of these definitions is termed a Requirement. Requirements are identified by unique requirement identifiers and, at minimum, contain a requirement title and textual description.



These requirements are fully described in the primary deliverables for this stage: the Requirements Document and the Requirements Traceability Matrix (RTM). The requirements document contains complete descriptions of each requirement, including diagrams and references to external documents as necessary. Note that detailed listings of database tables and fields are *not* included in the requirements document.

The title of each requirement is also placed into the first version of the RTM, along with the title of each goal from the project plan. The purpose of the RTM is to show that the product components developed during each stage of the software development lifecycle are formally connected to the components developed in prior stages.

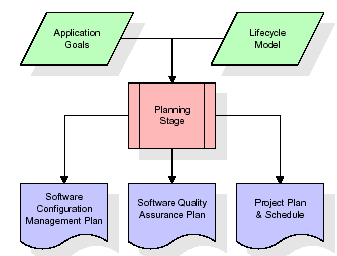
In the requirements stage, the RTM consists of a list of high-level requirements, or goals, by title, with a listing of associated requirements for each goal, listed by requirement title. In this hierarchical listing, the RTM shows that each requirement developed during this stage is formally linked to a specific product goal. In this format, each requirement can be traced to a specific product goal, hence the term requirements traceability.

The outputs of the requirements definition stage include the requirements document, the RTM, and an updated project plan.

* Feasibility study is all about identification of problems in a project.
* No. of staff required to handle a project is represented as Team Formation, in this case only modules are individual tasks will be assigned to employees who are working for that project.
* Project Specifications are all about representing of various possible inputs submitting to the server and corresponding outputs along with reports maintained by administrator.

**Analysis Stage:**

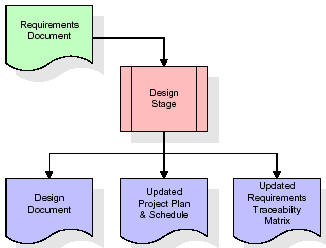
The planning stage establishes a bird's eye view of the intended software product, and uses this to establish the basic project structure, evaluate feasibility and risks associated with the project, and describe appropriate management and technical approaches.



The most critical section of the project plan is a listing of high-level product requirements, also referred to as goals. All of the software product requirements to be developed during the requirements definition stage flow from one or more of these goals. The minimum information for each goal consists of a title and textual description, although additional information and references to external documents may be included. The outputs of the project planning stage are the configuration management plan, the quality assurance plan, and the project plan and schedule, with a detailed listing of scheduled activities for the upcoming Requirements stage, and high level estimates of effort for the out stages.

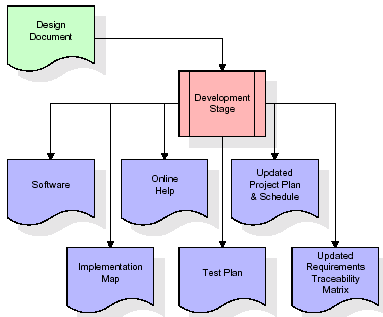
**Designing Stage:**

The design stage takes as its initial input the requirements identified in the approved requirements document. For each requirement, a set of one or more design elements will be produced as a result of interviews, workshops, and/or prototype efforts. Design elements describe the desired software features in detail, and generally include functional hierarchy diagrams, screen layout diagrams, tables of business rules, business process diagrams, pseudo code, and a complete entity-relationship diagram with a full data dictionary. These design elements are intended to describe the software in sufficient detail that skilled programmers may develop the software with minimal additional input.

  
When the design document is finalized and accepted, the RTM is updated to show that each design element is formally associated with a specific requirement. The outputs of the design stage are the design document, an updated RTM, and an updated project plan.

**Development (Coding) Stage:**

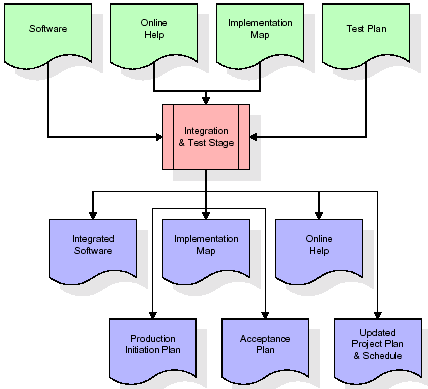
The development stage takes as its primary input the design elements described in the approved design document. For each design element, a set of one or more software artifacts will be produced. Software artifacts include but are not limited to menus, dialogs, and data management forms, data reporting formats, and specialized procedures and functions. Appropriate test cases will be developed for each set of functionally related software artifacts, and an online help system will be developed to guide users in their interactions with the software.



The RTM will be updated to show that each developed artifact is linked to a specific design element, and that each developed artifact has one or more corresponding test case items. At this point, the RTM is in its final configuration. The outputs of the development stage include a fully functional set of software that satisfies the requirements and design elements previously documented, an online help system that describes the operation of the software, an implementation map that identifies the primary code entry points for all major system functions, a test plan that describes the test cases to be used to validate the correctness and completeness of the software, an updated RTM, and an updated project plan.

**Integration & Test Stage:**

During the integration and test stage, the software artifacts, online help, and test data are migrated from the development environment to a separate test environment. At this point, all test cases are run to verify the correctness and completeness of the software. Successful execution of the test suite confirms a robust and complete migration capability. During this stage, reference data is finalized for production use and production users are identified and linked to their appropriate roles. The final reference data (or links to reference data source files) and production user list are compiled into the Production Initiation Plan.

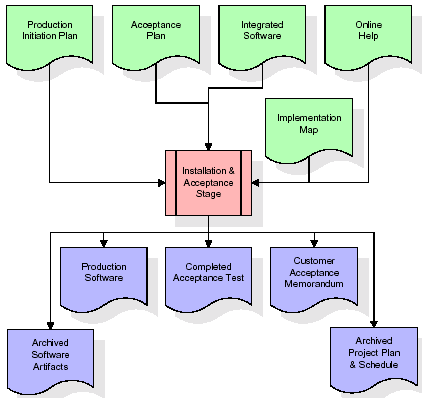


The outputs of the integration and test stage include an integrated set of software, an online help system, an implementation map, a production initiation plan that describes reference data and production users, an acceptance plan which contains the final suite of test cases, and an updated project plan.

* **Installation & Acceptance Test:**

During the installation and acceptance stage, the software artifacts, online help, and initial production data are loaded onto the production server. At this point, all test cases are run to verify the correctness and completeness of the software. Successful execution of the test suite is a prerequisite to acceptance of the software by the customer.

After customer personnel have verified that the initial production data load is correct and the test suite has been executed with satisfactory results, the customer formally accepts the delivery of the software.



The primary outputs of the installation and acceptance stage include a production application, a completed acceptance test suite, and a memorandum of customer acceptance of the software. Finally, the PDR enters the last of the actual labor data into the project schedule and locks the project as a permanent project record. At this point the PDR "locks" the project by archiving all software items, the implementation map, the source code, and the documentation for future reference.

**Maintenance:**

Outer rectangle represents maintenance of a project, Maintenance team will start with requirement study, understanding of documentation later employees will be assigned work and they will undergo training on that particular assigned category. For this life cycle there is no end, it will be continued so on like an umbrella (no ending point to umbrella sticks).

**3.4. Software Requirement Specification**

**3.4.1. Overall Description**

A Software Requirements Specification (SRS) – a requirements specification for a software system is a complete description of the behaviour of a system to be developed. It includes a set of use cases that describe all the interactions the users will have with the software. In addition to use cases, the SRS also contains non-functional requirements. Non-functional requirements are requirements which impose constraints on the design or implementation (such as performance engineering requirements, quality standards, or design constraints).

System requirements specification: A structured collection of information that embodies the requirements of a system. A business analyst, sometimes titled system analyst, is responsible for analysing the business needs of their clients and stakeholders to help identify business problems and propose solutions. Within the systems development lifecycle domain, the BA typically performs a liaison function between the business side of an enterprise and the information technology department or external service providers. Projects are subject to three sorts of requirements:

* Business requirements describe in business terms what must be delivered or accomplished to provide value.
* Product requirements describe properties of a system or product (which could be one of several ways to accomplish a set of business requirements.)
* Process requirements describe activities performed by the developing organization. For instance, process requirements could specify. Preliminary investigation examines project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All system is feasible if they are unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:
* **ECONOMIC FEASIBILITY**

A system can be developed technically and that will be used if installed must still be a good investment for the organization. In the economic feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs. The system is economically feasible. It does not require any addition hardware or software. Since the interface for this system is developed using the existing resources and technologies available at NIC, there is nominal expenditure and economical feasibility for certain.

* **Operational Feasibility**

Proposed projects are beneficial only if they can be turned out into information system. That will meet the organization’s operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration. So there is no question of resistance from the users that can undermine the possible application benefits. The well-planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

* **TECHNICAL FEASIBILITY**

Earlier no system existed to cater to the needs of ‘Secure Infrastructure Implementation System’. The current system developed is technically feasible. It is a web-based user interface for audit workflow at NIC-CSD. Thus, it provides an easy access to. the users. The database’s purpose is to create, establish and maintain a workflow among various entities in order to facilitate all concerned users in their various capacities or roles. Permission to the users would be granted based on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security.

**3.4.2. External Interface Requirements**

**User Interface**

The user interface of this system is a user-friendly python Graphical User Interface.

**Hardware Interfaces**

The interaction between the user and the console is achieved through python capabilities.

**Software Interfaces**

The required software is python.

**SYSTEM REQUIREMENT:**

**HARDWARE REQUIREMENTS:**

# Processor - Intel i3(min)

* Speed - 1.1 GHz
* RAM - 4GB (min)
* Hard Disk - 500 GB

**SOFTWARE REQUIREMENTS:**

* Operating System - Windows10(min)
* Programming Language - Python

**4. SYSTEM DESIGN**

**UML Diagram:**

The Unified Modelling Language allows the software engineer to express an analysis model using the modelling notation that is governed by a set of syntactic semantic and pragmatic rules.

A UML system is represented using five different views that describe the system from distinctly different perspective. Each view is defined by a set of diagrams, which is as follows.

* + **User Model View**
    1. This view represents the system from the user’s perspective.
    2. The analysis representation describes a usage scenario from the end-user’s perspective.
  + **Structural Model view**
    1. In this model the data and functionality are arrived from inside the system.
    2. This model view models the static structures.
* **Behavioural Model View**

It represents the dynamic of behavioural as parts of the system, depicting the interactions of collection between various structural elements described in the user model and structural model view.

* **Implementation Model View**

In this the structural and behavioural as parts of the system are represented as they are to be built.

* **Environmental Model View**

In these the structural and behavioural aspects of the environment in which the system is to be implemented are represented.

**Class Diagram:**

The class diagram is the main building block of object-oriented modeling. It is used both for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main objects, interactions in the application and the classes to be programmed. In the diagram, classes are represented with boxes which contain three parts:

* The upper part holds the name of the class
* The middle part contains the attributes of the class
* The bottom part gives the methods or operations the class can take or undertake.



**Use case Diagram:**

A **use case diagram** at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system. This type of diagram is typically used in conjunction with the textual use case and will often be accompanied by other types of diagrams as well.

****

****

**Sequence diagram:**

A sequence diagram is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

****

****

**Collaboration diagram:**

A collaboration diagram describes interactions among objects in terms of sequenced messages. Collaboration diagrams represent a combination of information taken from class, sequence, and use case diagrams describing both the static structure and dynamic behaviour of a system.





**Component Diagram:**

In the Unified Modelling Language, a component diagram depicts how components are wired together to form larger components and or software systems. They are used to illustrate the structure of arbitrarily complex systems.

Components are wired together by using an assembly connector to connect the required interface of one component with the provided interface of another component. This illustrates the service consumer - service provider relationship between the two components.

****

****

**Deployment Diagram:**

A **deployment diagram** in the Unified Modeling Language models the *physical* deployment of artifacts on nodes. To describe a web site, for example, a deployment diagram would show what hardware components ("nodes") exist (e.g., a web server, an application server, and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI).

The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have sub nodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.

****

****

**Activity Diagram:**

Activity diagram is another important diagram in UML to describe dynamic aspects of the system. It is basically a flow chart to represent the flow form one activity to another

activity. The activity can be described as an operation of the system. So the control flow is drawn from one operation to another. This flow can be sequential, branched or concurrent

**Faculty Login**

**Generated Summary**

**Subjective Question**

**Multiple Choice Question**

**View Marks**

**Model Analysis**

**Students Login**

**Generate Summary**

**Write Exams**

**View Marks**

**Data Flow Diagram:**

Data flow diagrams illustrate how data is processed by a system in terms of inputs and outputs. Data flow diagrams can be used to provide a clear representation of any business function. The technique starts with an overall picture of the business and continues by analyzing each of the functional areas of interest. This analysis can be carried out in precisely the level of detail required. The technique exploits a method called top-down expansion to conduct the analysis in a targeted way.

As the name suggests, Data Flow Diagram (DFD) is an illustration that explicates the passage of information in a process. A DFD can be easily drawn using simple symbols. Additionally, complicated processes can be easily automated by creating DFDs using easy-to-use, free downloadable diagramming tools. A DFD is a model for constructing and analyzing information processes. DFD illustrates the flow of information in a process depending upon the inputs and outputs. A DFD can also be referred to as a Process Model. A DFD demonstrates business or technical process with the support of the outside data saved, plus the data flowing from the process to another and the end results.

**User**

1. Faculty Login 2. Successfully Faculty Login

3. Generated Summary 4. Successfully Generated Summary

5. Subjective Question 6. Successfully Subjective Question

7. Multiple Choice Question 8. Successfully Multiple Choice Question

9. View Marks 10. Successfully View Marks

11. Model Analysis 12. Successfully Model Analysis

**User**

1. Students Login 2. Successfully Students Login

3. Generate Summary 4. Successfully Generate Summary

5. Write Exam 6. Successfully Write Exam

7. View Marks 8. Successfully View Marks

**5. IMPLEMETATION**

**5.1 PYTHON**

\* One of the most popular languages is Python. Guido van Rossum released this language in 1991. Python is available on the Mac, Windows, and Raspberry Pi operating systems. The syntax of Python is simple and identical to that of English. When compared to Python, it was seen that the other language requires a few extra lines.

\*It is an interpreter-based language because code may be run line by line after it has been written. This implies that rapid prototyping is possible across all platforms. Python is a big language with a free, binary-distributed interpreter standard library.

\* It is inferior to maintenance that is conducted and is straightforward to learn. It is an object-oriented, interpreted programming language. It supports several different programming paradigms in addition to object-oriented programming, including functional and procedural programming.

\* It supports several different programming paradigms in addition to object-oriented programming, including practical and procedural programming. Python is mighty while maintaining a relatively straightforward syntax. Classes, highly dynamic data types, modules, and exceptions are covered. Python can also be utilised by programmes that require programmable interfaces as an external language.

Here are some key features and characteristics of Python:

* Readability: Python emphasizes code readability with its clean and intuitive syntax. It uses indentation and whitespace to structure code blocks, making it easy to understand and maintain.
* Easy to Learn: Python's simplicity and readability make it an excellent choice for beginners. Its straightforward syntax and extensive documentation make it accessible for newcomers to programming.
* Interpreted Language: Python is an interpreted language, meaning that it doesn't need to be compiled before running. The Python interpreter reads and executes the code directly, making the development process faster and more interactive.
* Cross-platform Compatibility: Python is available for major operating systems like Windows, macOS, and Linux. This cross-platform compatibility allows developers to write code once and run it on different platforms without modifications.
* Large Standard Library: Python comes with a vast standard library that provides ready-to-use modules and functions for various tasks. It covers areas such as file I/O, networking, regular expressions, databases, and more, saving developers time and effort.
* Extensible and Modular: Python supports modular programming, enabling developers to organize code into reusable modules and packages. Additionally, Python allows integrating modules written in other languages, such as C or C++, providing flexibility and performance optimizations.
* Wide Range of Libraries and Frameworks: Python has a vibrant ecosystem with numerous third-party libraries and frameworks. These libraries, such as NumPy, pandas, TensorFlow, and Django, extend Python's capabilities for specific domains, making it a powerful tool for diverse applications.
* Object-Oriented: Python supports object-oriented programming (OOP) principles, allowing developers to create and work with classes and objects. OOP provides a structured approach to code organization, promoting code reuse and modularity.
* Dynamic Typing: Python is dynamically typed, meaning variable types are determined at runtime. Developers do not need to declare variable types explicitly, which enhances flexibility and simplifies code writing.

**5.2 Installation**

To install Python on your computer, follow these basic steps:

* Step 1: Visit the Python website Go to the official Python website at <https://www.python.org/>.
* Step 2: Select the operating system Choose the appropriate installer for your operating system. Python supports Windows, macOS, and various Linux distributions. Make sure to select the correct version that matches your operating system.
* Step 3: Check which version of Python is installed; if the 3.7.0 version is not there, uninstall it through the control panel and
* Step 4: Install Python 3.7.0 using Cmd.
* Step 5: Install the all libraries that required to run the project
* Step 6: Run

**5.3 Python Features:**

1. **Easy:** Because Python is a more accessible and straightforward language, Python programming is easier to learn.
2. **Interpreted language:** Python is an interpreted language, therefore it can be used to examine the code line by line and provide results.
3. **Open Source:** Python is a free online programming language since it is open-source.
4. **Portable:** Python is portable because the same code may be used on several computer standard
5. **libraries:** Python offers a sizable library that we may utilize to create applications quickly.
6. **GUI:** It stands for GUI (Graphical User Interface)
7. **Dynamical typed:** Python is a dynamically typed language, therefore the type of the value will be determined at runtime.

**5.4 Python GUI (Tkinter)**

* Python provides a wide range of options for GUI development (Graphical User Interfaces).
* Tkinter, the most widely used GUI technique, is used for all of them.
* The Tk GUI toolkit offered by Python is used with the conventional Python interface.
* Tkinter is the easiest and quickest way to write Python GUI programs.
* Using Tkinter, creating a GUI is simple.
* A part of Python's built-in library is Tkinter. The GUI programs were created.
* Python and Tkinter together give a straightforward and quick way. The Tk GUI toolkit's object-oriented user interface is called Tkinter.

Making a GUI application is easy using Tkinter. Following are the steps:

1) Install the Tkinter module in place.

2) The GUI applicatioMakeske the primary window

3) Include one or more of the widgets mentioned above in the GUI application.

4) Set up the main event loop such that it reacts to each user-initiated event.

Although Tkinter is the only GUI framework included in the Python standard library, Python includes a GUI framework. The default library for Python is called Tkinter. Tk is a scripting language often used in designing, testing, and developing GUIs. Tk is a free, open-source widget toolkit that may be used to build GUI applications in a wide range of computer languages.

**5.5 Python IDLE**

* Python IDLE offers a full-fledged file editor, which gives you the ability to write and execute Python programs from within this program. The built-in file editor also includes several features, like code completion and automatic indentation, that will speed up your coding workflow.
* Guido Van Rossum named Python after the British comedy group Monty Python while the name IDLE was chosen to pay tribute to Eric Idle, who was one of the Monty Python's founding members. IDLE comes bundled with the default implementation of the Python language since the 01.5. 2b1 release
* IDLE is used to execute statements similar to Python Shell. IDLE is used to create, modify, and execute Python code. IDLE provides a fully-featured text editor to write Python scripts and provides features like syntax highlighting, auto-completion, and smart indent.
* IDLE has two modes: interactive and script. We wrote our first program, “Hello, World!” in interactive mode. Interactive mode immediately returns the results of commands you enter into the shell. In script mode, you will write a script and then run it.
* The IDE Python IDLE is a good place to start as it helps you become familiar with the way Python works and understand its syntax. This IDE is good to start programming in Python due to its great debugger, but once you are fluent and start developing projects it is necessary to jump to another, more complete IDE.
* Python IDLE (Integrated Development and Learning Environment) is an interactive development environment included with the Python programming language. It provides a convenient way to write, execute, and debug Python code.

When you install Python, IDLE is typically installed along with it. To open IDLE, you can follow these steps:

* Open the command prompt (Windows) or terminal (macOS/Linux).
* Type "idle" and press Enter. Alternatively, you can specify the version with "idle3" or "idle2" for Python 3 or Python 2, respectively.
* Once IDLE is launched, you will see the Python shell, which is an interactive environment where you can type and execute Python code directly.

Here are some features and functionalities provided by Python IDLE:

* Editor: IDLE includes a text editor where you can write your Python code. It offers syntax highlighting, automatic indentation, and code completion to enhance your coding experience.
* Interactive Shell: The Python shell in IDLE allows you to execute Python code interactively. You can type commands, statements, or function calls directly in the shell, and Python will execute them immediately.
* Debugging: IDLE provides basic debugging capabilities to help you find and fix errors in your code. You can set breakpoints, step through code, inspect variables, and track the program's execution.
* Python Help: IDLE provides access to the Python documentation and built-in help. You can access the help menu to find information about Python modules, functions, classes, and more.
* Script Execution: In addition to the interactive shell, IDLE allows you to run Python scripts stored in files. You can write your code in the editor and execute it as a script to see the output or interact with the program.
* Customization: IDLE can be customized to suit your preferences. You can modify settings related to syntax highlighting, indentation, fonts, and more.
* Python IDLE serves as a beginner-friendly development environment and learning tool. It is suitable for writing small scripts, testing code snippets, experimenting with Python features, and learning the language's basics. However, for more advanced development projects, you may consider using other code editors or integrated development environments (IDEs) that provide additional features and better project management capabilities.

**5.6 Libraries**

In Python, libraries (also referred to as modules or packages) are collections of pre-written code that provide additional functionality and tools to extend the capabilities of the Python language. Libraries contain reusable code that developers can leverage to perform specific tasks without having to write everything from scratch.

Python libraries are designed to solve common problems, such as handling data, performing mathematical operations, interacting with databases, working with files, implementing networking protocols, creating graphical user interfaces (GUIs), and much more. They provide ready-to-use functions, classes, and methods that simplify complex operations and save development time.

**Libraries in Python offer various advantages:**

* Code Reusability:
* Efficiency:
* Collaboration
* Domain-Specific Functionality
* To use a Python library, you need to install it first.

There are some libraries following:

* **Pandas:**

Pandas are a Python computer language library for data analysis and manipulation. It offers a specific operation and data format for handling time series and numerical tables. It differs significantly from the release3-clause of the BSD license. It is a well-liked open-source of opinion that is utilized in machine learning and data analysis.

Pandas are a Python package providing fast, flexible, and expressive data structures designed to make working with “relational” or “labelled” data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real-world data analysis in Python. Pandas are a Python library used for working with data sets.

* It has functions for analysing, cleaning, exploring, and manipulating data.
* The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis" and was created by Wes McKinney in 2008.
* Pandas allow us to analyse big data and make conclusions based on statistical theories.
* Pandas can clean messy data sets, and make them readable and relevant.

Relevant data is very important in data science. Pandas are a Python library for data analysis. Started by Wes McKinney in 2008 out of a need for a powerful and flexible quantitative analysis tool, pandas have grown into one of the most popular Python libraries. It has an extremely active community of contributors. The name is derived from the term "panel data", an econometrics term for data sets that include observations over multiple time periods for the same individuals. Its name is a play on the phrase "Python data analysis" itself.

* **NumPy:**

The NumPy Python library for multi-dimensional, big-scale matrices adds a huge number of high-level mathematical functions. It is possible to modify NumPy by utilizing a Python library. Along with line, algebra, and the Fourier transform operations, it also contains several matrices-related functions.

NumPy can be used to perform a wide variety of mathematical operations on arrays. It adds powerful data structures to Python that guarantee efficient calculations with arrays and matrices and it supplies an enormous library of high-level mathematical functions that operate on these arrays and matrices.

* NumPy is a Python library used for working with arrays.
* It also has functions for working in domain of linear algebra, Fourier transform, and matrices.
* NumPy was created in 2005 by Travis Oliphant. It is an open-source project and you can use it freely.
* NumPy stands for Numerical Python.
* In Python we have lists that serve the purpose of arrays, but they are slow to process.
* NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.
* The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy.
* Arrays are very frequently used in data science, where speed and resources are very important.
* **Matplotlib:**

It is a multi-platform, array-based data visualization framework built to interact with the whole SciPy stack. MATLAB is proposed as an open-source alternative. Matplotlib is a Python extension and a cross-platform toolkit for graphical plotting and visualization.

Matplotlib is a popular Python library for creating static, animated, and interactive visualizations. It provides a flexible and comprehensive set of tools for generating plots, charts, histograms, scatter plots, and more. Matplotlib is widely used in various fields, including data analysis, scientific research, and data visualization.

Here are some key features and functionalities of the Matplotlib library:

* Plotting Functions
* Customization Options
* Multiple Interfaces
* Integration with NumPy and pandas
* Subplots and Figures:
* Saving and Exporting
* **Scikit-learn:**

The most stable and practical machine learning library for Python is scikit-learn. Regression, dimensionality reduction, classification, and clustering are just a few of the helpful tools it provides through the Python interface for statistical modeling and machine learning. It is an essential part of the Python machine learning toolbox used by JP Morgan. It is frequently used in various machine learning applications, including classification and predictive analysis.

Scikit-learn (also referred to as sklearn) is a widely used open-source machine learning library for Python. It provides a comprehensive set of tools and algorithms for various machine learning tasks, including classification, regression, clustering, dimensionality reduction, model selection, and pre-processing.

Here are some key features and functionalities of the Scikit-learn library:

* Easy-to-Use Interface:
* Broad Range of Algorithms:
* Data Pre-processing and Feature Engineering:
* Model Evaluation and Validation:
* Integration with NumPy and pandas:
* Robust Documentation and Community Support:
* **Keras:**

\* Google's Keras is a cutting-edge deep learning API for creating neural networks. It is created in Python and is designed to simplify the development of neural networks. Additionally, it enables the use of various neural networks for computation. Deep learning models are developed and tested using the free and open-source Python software known as Keras.

Keras is a high-level deep learning library for Python. It is designed to provide a user-friendly and intuitive interface for building and training deep learning models. Keras acts as a front-end API, allowing developers to define and configure neural networks while leveraging the computational backend engines, such as Tensor Flow or Theano.

Here are some key features and functionalities of the Keras library:

* User-Friendly API
* Multi-backend Support
* Wide Range of Neural Network Architectures
* Pre-trained Models and Transfer Learning:
* Easy Model Training and Evaluation:
* GPU Support:
* **h5py:**

\* The h5py Python module offers an interface for the binary HDF5 data format. Thanks to p5py, the top can quickly halt the vast amount of numerical data and alter it using the NumPy library. It employs common syntax for Python, NumPy, and dictionary arrays.

h5py is a Python library that provides a simple and efficient interface for working with datasets and files in the Hierarchical Data Format 5 (HDF5) format. HDF5 is a versatile data format commonly used for storing and managing large volumes of numerical data.

Here are some key features and functionalities of the h5py library:

* + HDF5 File Access
  + Dataset Handling:
  + Group Organization:
  + Attributes:
  + Compatibility with NumPy
  + Performance
* **Tensor flow**

TensorFlow is a Python library for fast numerical computing created and released by Google. It is a foundation library that can be used to create Deep Learning models directly or by using wrapper libraries that simplify the process built on top of TensorFlow. TensorFlow is an end-to-end open-source platform for machine learning. TensorFlow is a rich system for managing all aspects of a machine learning system; however, this class focuses on using a particular TensorFlow API to develop and train machine learning models.

TensorFlow is a popular open-source library for machine learning and deep learning. It provides a comprehensive set of tools, APIs, and computational resources for building and training various types of machine learning models, especially neural networks.

Here are some key features and functionalities of TensorFlow:

* Neural Network Framework:
* Computational Graphs
* Automatic Differentiation
* GPU and TPU Support
* Distributed Computing
* Deployment Capabilities
* **Tkinter**

Tkinter is an acronym for "Tk interface". Tk was developed as a GUI extension for the Tcl scripting language by John Ousterhout. The first release was in 1991. Tkinter is the de facto way in Python to create Graphical User interfaces (GUIs) and is included in all standard Python Distributions. In fact, it's the only framework built into the Python standard library.

Tkinter is a standard Python library used for creating graphical user interfaces (GUIs). It provides a set of modules and classes that allow you to develop interactive and visually appealing desktop applications.

Here are some key features and functionalities of Tkinter:

* Cross-Platform Compatibility
* Simple and Easy-to-Use
* Widgets and Layout Management
* Event-Driven Programming
* Customization and Styling
* Integration with Other Libraries
* **NLTK**

NLTK is a toolkit build for working with NLP in Python. It provides us various text processing libraries with a lot of test datasets. A variety of tasks can be performed using NLTK such as tokenizing, parse tree visualization, etc NLTK (Natural Language Toolkit) is the go-to API for NLP (Natural Language Processing) with Python. It is a really powerful tool to pre-process text data for further analysis like with ML models for instance. It helps convert text into numbers, which the model can then easily work with.

NLTK (Natural Language Toolkit) is a Python library widely used for working with human language data and implementing natural language processing (NLP) tasks. It provides a set of tools, corpora, and resources for tasks such as tokenization, stemming, tagging, parsing, sentiment analysis, and more.

Here are some key features and functionalities of NLTK:

* Text Processing
* Part-of-Speech Tagging
* Named Entity Recognition
* Chunking and Parsing
* Sentiment Analysis:
* WordNet Integration:
* **Scipy**

SciPy is a collection of mathematical algorithms and convenience functions built on the NumPy extension of Python. It adds significant power to the interactive Python session by providing the user with high-level commands and classes for manipulating and visualizing data.

SciPy is a powerful scientific computing library for Python that provides a wide range of mathematical algorithms and functions. It builds upon NumPy, another fundamental library for numerical computing, and extends its capabilities by adding additional tools for scientific and technical computing tasks.

Here are some key features and functionalities of SciPy:

* Numerical Integration:
* Optimization and Root Finding
* Linear Algebra
* Signal and Image Processing
* Statistics

**5.2 Sample Code:**

**Main.py**

from django.shortcuts import render

from django.template import RequestContext

from django.contrib import messages

from django.http import HttpResponse

import os

import numpy as np

import pymysql

from pyresparser import utils

from django.core.files.storage import FileSystemStorage

import torch

from transformers import T5Tokenizer, T5ForConditionalGeneration, T5Config

from transformers import AutoTokenizer

from questiongenerator import QuestionGenerator

from questiongenerator import print\_qa

from sklearn.feature\_extraction.text import TfidfVectorizer

from numpy import dot

from numpy.linalg import norm

from datetime import date

global uname, accuracy, question\_count

global model, tokenizer, device

model = T5ForConditionalGeneration.from\_pretrained('t5-small')

tokenizer = T5Tokenizer.from\_pretrained('t5-small',model\_max\_length=512)

device = torch.device('cpu')

qg = QuestionGenerator()

def ViewAnswersAction(request):

if request.method == 'POST':

global uname, question\_count

aid = request.POST.get('t1', False)

cols = ['username', 'Assignment ID', 'Question', 'Correct Answer', 'Your Answer']

output = '<table border="1" align="center" width="100%"><tr>'

font = '<font size="" color="black">'

for i in range(len(cols)):

output += "<td>"+font+cols[i]+"</font></td>"

output += "</tr>"

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select \* from student\_answers where username='"+uname+"' and assignment\_id='"+aid+"'")

rows = cur.fetchall()

for row in rows:

output += "<tr><td>"+font+str(row[0])+"</font></td>"

output += "<td>"+font+str(row[1])+"</font></td>"

output += "<td>"+font+str(row[2])+"</font></td>"

output += "<td>"+font+str(row[3])+"</font></td>"

output += "<td>"+font+str(row[4])+"</font></td></tr>"

output += "</table><br/><br/><br/><br/>"

context= {'data':output}

return render(request, "StudentScreen.html", context)

def ViewAnswers(request):

if request.method == 'GET':

global username

aid = []

output = '<tr><td><font size="3" color="black">Choose&nbsp;Assignment&nbsp;Id</td><td><select name="t1">'

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select assignment\_id FROM subjective")

rows = cur.fetchall()

for row in rows:

assignment = str(row[0])

if assignment not in aid:

aid.append(assignment)

output += '<option value="'+assignment+'">'+assignment+"</option>"

with con:

cur = con.cursor()

cur.execute("select assignment\_id FROM multiplechoice")

rows = cur.fetchall()

for row in rows:

assignment = str(row[0])

if assignment not in aid:

aid.append(assignment)

output += '<option value="'+assignment+'">'+assignment+"</option>"

output += "</select></td></tr>"

context= {'data1':output}

return render(request,'ViewAnswers.html', context)

def StudentMarks(request):

if request.method == 'GET':

global uname

cols = ['Assignment ID', 'Student Name', 'Marks', 'Attempt Date']

output = '<table border="1" align="center" width="100%"><tr>'

font = '<font size="" color="black">'

for i in range(len(cols)):

output += "<td>"+font+cols[i]+"</font></td>"

output += "</tr>"

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select \* FROM marks where student\_name='"+uname+"'")

rows = cur.fetchall()

for row in rows:

output += "<tr><td>"+font+str(row[0])+"</font></td>"

output += "<td>"+font+str(row[1])+"</font></td>"

output += "<td>"+font+str(row[2])+"</font></td>"

output += "<td>"+font+str(row[3])+"</font></td></tr>"

output += "</table><br/><br/><br/><br/>"

context= {'data':output}

return render(request, "StudentScreen.html", context)

def calculateMarks(question, user\_answer, assignment\_id):

correct\_answer = ""

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select answer FROM subjective where assignment\_id='"+assignment\_id+"' and question='"+question+"'")

rows = cur.fetchall()

for row in rows:

correct\_answer = row[0]

vectorizer = TfidfVectorizer(use\_idf=True, smooth\_idf=False, norm=None, decode\_error='replace')

X = vectorizer.fit\_transform([correct\_answer]).toarray()

test = vectorizer.transform([user\_answer]).toarray()

score = dot(X[0], test[0]) / (norm(X[0])\* norm(test[0]))

return score

def getCorrectOption(question, assignment\_id):

correct\_answer = ""

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select correct FROM multiplechoice where assignment\_id='"+assignment\_id+"' and question='"+question+"'")

rows = cur.fetchall()

for row in rows:

correct\_answer = row[0]

break

return correct\_answer

def saveStudentAnswer(question, answer, exam\_type, assignment\_id):

global uname

query = ""

if exam\_type == "subjective":

query = "select answer FROM subjective where question='"+question+"'"

else:

query = "select correct FROM multiplechoice where question='"+question+"'"

correct\_answer = ""

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute(query)

rows = cur.fetchall()

for row in rows:

correct\_answer = row[0]

break

db\_connection = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

db\_cursor = db\_connection.cursor()

student\_sql\_query = "INSERT INTO student\_answers VALUES('"+str(uname)+"','"+str(assignment\_id)+"','"+str(question)+"','"+correct\_answer+"','"+answer+"')"

db\_cursor.execute(student\_sql\_query)

db\_connection.commit()

def WriteExamAction(request):

if request.method == 'POST':

global uname, question\_count

assignment\_id = request.POST.get('t1', False)

assignment\_id = assignment\_id.split("-")

assignment\_type = assignment\_id[1]

assignment\_id = assignment\_id[0]

score = 0

count = 0

print(str(assignment\_id)+" "+assignment\_type)

if assignment\_type == 'subjective':

for i in range(2, question\_count):

question = request.POST.get('tq'+str(i), False)

answer = request.POST.get('ta'+str(i), False)

saveStudentAnswer(question, answer, "subjective", assignment\_id)

print(str(question)+" "+str(answer))

score += calculateMarks(question, answer, assignment\_id)

count += 1

if score > 0:

score = score / count

else:

for i in range(2, question\_count):

question = request.POST.get('tq'+str(i), False)

answer = request.POST.get('ta'+str(i), False)

choosen\_option = getCorrectOption(question, assignment\_id)

saveStudentAnswer(question, choosen\_option, "multiple", assignment\_id)

if answer == choosen\_option:

score += 1

count += 1

if score > 0:

score = score / count

dd = str(date.today())

db\_connection = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

db\_cursor = db\_connection.cursor()

student\_sql\_query = "INSERT INTO marks VALUES('"+str(assignment\_id)+"','"+uname+"','"+str(score)+"','"+str(dd)+"')"

db\_cursor.execute(student\_sql\_query)

db\_connection.commit()

context= {'data':"Your score = "+str(score)}

return render(request,'StudentScreen.html', context)

def getMultiple(assignment\_id):

global question\_count

question\_count = 2

output = ""

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select question,option\_a,option\_b,option\_c,option\_d FROM multiplechoice where assignment\_id='"+assignment\_id+"'")

rows = cur.fetchall()

for row in rows:

output += '<tr><td><font size="3" color="black">Question</td><td><input type="text" name="tq'+str(question\_count)+'" size="60" value="'+row[0]+'" readonly/></td></tr>'

output += '<tr><td><font size="3" color="black">Option A: </td><td>'+row[1]+'<input type="radio" name="ta'+str(question\_count)+'" value="'+row[1]+'"/></td></tr>'

output += '<tr><td><font size="3" color="black">Option B: </td><td>'+row[2]+'<input type="radio" name="ta'+str(question\_count)+'" value="'+row[2]+'"/></td></tr>'

output += '<tr><td><font size="3" color="black">Option C: </td><td>'+row[3]+'<input type="radio" name="ta'+str(question\_count)+'" value="'+row[3]+'"/></td></tr>'

output += '<tr><td><font size="3" color="black">Option D: </td><td>'+row[4]+'<input type="radio" name="ta'+str(question\_count)+'" value="'+row[4]+'"/></td></tr>'

output += "<br/>"

question\_count += 1

return output

def getSubjective(assignment\_id):

global question\_count

question\_count = 2

output = ""

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select question FROM subjective where assignment\_id='"+assignment\_id+"'")

rows = cur.fetchall()

for row in rows:

output += '<tr><td><font size="3" color="black">Question</td><td><input type="text" name="tq'+str(question\_count)+'" size="60" value="'+str(row[0])+'" readonly/></td></tr>'

output += '<tr><td><font size="3" color="black">Your&nbsp;Answer</td><td><input type="text" name="ta'+str(question\_count)+'" size="60" /></td></tr>'

question\_count += 1

return output

def ShowQuestions(request):

if request.method == 'GET':

global username

assignment\_id = request.GET['aid']

assignment\_type = request.GET['type']

output = '<tr><td><font size="3" color="black">Assignment&nbsp;Type</td><td><input type="text" name="t1" size="15" value="'+assignment\_id+"-"+assignment\_type+'" readonly/></td></tr>'

if assignment\_type == 'subjective':

output += getSubjective(assignment\_id)

else:

output += getMultiple(assignment\_id)

print(output)

context= {'data1':output}

return render(request,'ExamScreen.html', context)

def WriteExam(request):

if request.method == 'GET':

global uname, accuracy

accuracy = 0

count = 0

cols = ['Assignment ID', 'Subject Name', 'Faculty Name', 'Assignment Type', 'Click Here to Write']

output = '<table border="1" align="center" width="100%"><tr>'

font = '<font size="" color="black">'

for i in range(len(cols)):

output += "<td>"+font+cols[i]+"</font></td>"

output += "</tr>"

aid = []

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select assignment\_id,subject\_name, faculty\_name FROM subjective")

rows = cur.fetchall()

for row in rows:

if row[0] not in aid:

aid.append(row[0])

output += "<tr><td>"+font+str(row[0])+"</font></td>"

output += "<td>"+font+str(row[1])+"</font></td>"

output += "<td>"+font+str(row[2])+"</font></td>"

output += "<td>"+font+"Subjective</font></td>"

output+='<td><a href=\'ShowQuestions?aid='+str(row[0])+'&type=subjective\'><font size=3 color=red>Click Here to write</font></a></td></tr>'

aid = []

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select assignment\_id,subject\_name, faculty\_name FROM multiplechoice")

rows = cur.fetchall()

for row in rows:

if row[0] not in aid:

aid.append(row[0])

output += "<tr><td>"+font+str(row[0])+"</font></td>"

output += "<td>"+font+str(row[1])+"</font></td>"

output += "<td>"+font+str(row[2])+"</font></td>"

output += "<td>"+font+"Multiple Choice</font></td>"

output+='<td><a href=\'ShowQuestions?aid='+str(row[0])+'&type=multiple\'><font size=3 color=red>Click Here to write</font></a></td></tr>'

output += "</table><br/><br/><br/><br/>"

context= {'data':output}

return render(request, "StudentScreen.html", context)

def StudentSummaryAction(request):

if request.method == 'POST':

global uname, model, tokenizer, device

text = request.POST.get('t1', False)

data = None

summary = "Unable to extract summary from given text"

if len(text.strip()) > 0:

data = text

else:

filename = request.FILES['t2'].name

ext = filename.split(".")[1]

data = request.FILES['t2'].read() #reading uploaded file from user

if os.path.exists("LearningApp/static/"+filename):

os.remove("LearningApp/static/"+filename)

with open("LearningApp/static/"+filename, "wb") as file:

file.write(data)

file.close()

text\_raw = utils.extract\_text("LearningApp/static/"+filename, '.'+ext)

data = ' '.join(text\_raw.split())

if data is not None:

data = data.strip().replace('\n',' ')

data = 'summarize: ' + data

tokenizedText = tokenizer.encode(data, return\_tensors='pt', max\_length=512, truncation=True).to(device)

summaryIds = model.generate(tokenizedText, min\_length=30, max\_length=120)

summary = tokenizer.decode(summaryIds[0], skip\_special\_tokens=True)

context= {'data': "<font size=3 color=black>Generated Summary = "+summary+"</font>"}

return render(request, 'StudentScreen.html', context)

def StudentSummary(request):

if request.method == 'GET':

return render(request, 'StudentSummary.html', {})

def getId(table\_name):

assignment\_id = 0

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select max(assignment\_id) FROM "+table\_name)

rows = cur.fetchall()

for row in rows:

assignment\_id = row[0]

break

if assignment\_id is not None:

assignment\_id += 1

else:

assignment\_id = 1

return assignment\_id

def ModelAnalysis(request):

if request.method == 'GET':

global uname, accuracy

output = "Student Answer Evaluation Model Accuracy = "+str(accuracy - 0.012)

context= {'data':output}

return render(request, "FacultyScreen.html", context)

def ViewMarks(request):

if request.method == 'GET':

global uname, accuracy

accuracy = 0

count = 0

cols = ['Assignment ID', 'Student Name', 'Marks', 'Attempt Date']

output = '<table border="1" align="center" width="100%"><tr>'

font = '<font size="" color="black">'

for i in range(len(cols)):

output += "<td>"+font+cols[i]+"</font></td>"

output += "</tr>"

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select \* FROM marks")

rows = cur.fetchall()

for row in rows:

output += "<tr><td>"+font+str(row[0])+"</font></td>"

output += "<td>"+font+str(row[1])+"</font></td>"

output += "<td>"+font+str(row[2])+"</font></td>"

output += "<td>"+font+str(row[3])+"</font></td></tr>"

accuracy += float(row[2])

count += 1

if count > 0:

accuracy = accuracy / count

output += "</table><br/><br/><br/><br/>"

context= {'data':output}

return render(request, "FacultyScreen.html", context)

def ChoiceQuestionAction(request):

if request.method == 'POST':

global uname, qg

questions = ""

subject = request.POST.get('t1', False)

text = request.POST.get('t2', False)

data = None

assignment\_id = getId('multiplechoice')

if len(text.strip()) > 0:

data = text

else:

filename = request.FILES['t3'].name

ext = filename.split(".")[1]

data = request.FILES['t3'].read() #reading uploaded file from user

if os.path.exists("LearningApp/static/"+filename):

os.remove("LearningApp/static/"+filename)

with open("LearningApp/static/"+filename, "wb") as file:

file.write(data)

file.close()

text\_raw = utils.extract\_text("LearningApp/static/"+filename, '.'+ext)

data = ' '.join(text\_raw.split())

if data is not None:

qa\_list = qg.generate(data, num\_questions=10, answer\_style='all')

for k in range(len(qa\_list)):

data = qa\_list[k]

question = data['question']

answer = data['answer']

if len(answer) > 1:

option\_a = "No option"

option\_b = "No option"

option\_c = "No option"

option\_d = "No option"

correct\_answer = "No answer"

print(question+" == "+str(answer))

if type(answer) is list:

for i in range(len(answer)):

correct = answer[i]['correct']

if correct == True:

correct\_answer = answer[i]['answer']

if i == 0:

option\_a = answer[i]['answer']

if i == 1:

option\_b = answer[i]['answer']

if i == 2:

option\_c = answer[i]['answer']

if i == 3:

option\_d = answer[i]['answer']

if option\_a != correct\_answer and option\_b != correct\_answer and option\_c != correct\_answer and option\_d != correct\_answer:

option\_a = correct\_answer

question = question.replace("'","")

question = question.replace(",","")

correct\_answer = correct\_answer.replace("'","")

correct\_answer = correct\_answer.replace(",","")

option\_a = option\_a.replace("'","")

option\_a = option\_a.replace(",","")

option\_b = option\_b.replace("'","")

option\_b = option\_b.replace(",","")

option\_c = option\_c.replace("'","")

option\_c = option\_c.replace(",","")

option\_d = option\_d.replace("'","")

option\_d = option\_d.replace(",","")

db\_connection = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

db\_cursor = db\_connection.cursor()

student\_sql\_query = "INSERT INTO multiplechoice VALUES('"+str(assignment\_id)+"','"+subject+"','"+uname+"','"+question+"','"+option\_a+"','"+option\_b+"','"+option\_c+"','"+option\_d+"','"+correct\_answer+"')"

db\_cursor.execute(student\_sql\_query)

db\_connection.commit()

questions += question+" Options = "+option\_a+", "+option\_b+", "+option\_c+", "+option\_d+"<br/><br/>"

if len(questions) == 0:

questions = "Unable to identify question. Please try some other text"

context= {'data': questions}

return render(request, 'FacultyScreen.html', context)

def ChoiceQuestion(request):

if request.method == 'GET':

return render(request, 'ChoiceQuestion.html', {})

def SubjectiveAction(request):

if request.method == 'POST':

global uname, qg

questions = ""

subject = request.POST.get('t1', False)

text = request.POST.get('t2', False)

data = None

assignment\_id = getId('subjective')

if len(text.strip()) > 0:

data = text

else:

filename = request.FILES['t3'].name

ext = filename.split(".")[1]

data = request.FILES['t3'].read() #reading uploaded file from user

if os.path.exists("LearningApp/static/"+filename):

os.remove("LearningApp/static/"+filename)

with open("LearningApp/static/"+filename, "wb") as file:

file.write(data)

file.close()

text\_raw = utils.extract\_text("LearningApp/static/"+filename, '.'+ext)

data = ' '.join(text\_raw.split())

if data is not None:

qa\_list = qg.generate(data, num\_questions=10, answer\_style='all')

for k in range(len(qa\_list)):

data = qa\_list[k]

question = data['question']

answer = data['answer']

correct\_answer = ""

print(question+" == "+str(answer))

if type(answer) is list:

for i in range(len(answer)):

correct = answer[i]['correct']

if correct == True:

correct\_answer = answer[i]['answer']

else:

correct\_answer = answer

question = question.replace("'","")

question = question.replace(",","")

correct\_answer = correct\_answer.replace("'","")

correct\_answer = correct\_answer.replace(",","")

db\_connection = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

db\_cursor = db\_connection.cursor()

student\_sql\_query = "INSERT INTO subjective VALUES('"+str(assignment\_id)+"','"+subject+"','"+uname+"','"+question+"','"+correct\_answer+"')"

db\_cursor.execute(student\_sql\_query)

db\_connection.commit()

questions += question+"<br/><br/>"

if len(questions) == 0:

questions = "Unable to identify question. Please try some other text"

context= {'data': questions}

return render(request, 'FacultyScreen.html', context)

def Subjective(request):

if request.method == 'GET':

return render(request, 'Subjective.html', {})

def SummaryAction(request):

if request.method == 'POST':

global uname, model, tokenizer, device

text = request.POST.get('t1', False)

text = text.strip()

data = None

summary = "Unable to extract summary from given text"

if len(text) > 0:

data = text

else:

filename = request.FILES['t2']

filename = request.FILES['t2'].name

ext = filename.split(".")[1]

data = request.FILES['t2'].read() #reading uploaded file from user

if os.path.exists("LearningApp/static/"+filename):

os.remove("LearningApp/static/"+filename)

with open("LearningApp/static/"+filename, "wb") as file:

file.write(data)

file.close()

text\_raw = utils.extract\_text("LearningApp/static/"+filename, '.'+ext)

data = ' '.join(text\_raw.split())

if data is not None:

data = data.strip().replace('\n',' ')

data = 'summarize: ' + data

tokenizedText = tokenizer.encode(data, return\_tensors='pt', max\_length=512, truncation=True).to(device)

summaryIds = model.generate(tokenizedText, min\_length=30, max\_length=120)

summary = tokenizer.decode(summaryIds[0], skip\_special\_tokens=True)

context= {'data': "<font size=3 color=black>Generated Summary = "+summary+"</font>"}

return render(request, 'FacultyScreen.html', context)

def Summary(request):

if request.method == 'GET':

return render(request, 'Summary.html', {})

def Signup(request):

if request.method == 'GET':

return render(request, 'Signup.html', {})

def index(request):

if request.method == 'GET':

return render(request, 'index.html', {})

def FacultyLogin(request):

if request.method == 'GET':

return render(request, 'FacultyLogin.html', {})

def StudentLogin(request):

if request.method == 'GET':

return render(request, 'StudentLogin.html', {})

def StudentLoginAction(request):

if request.method == 'POST':

global uname

username = request.POST.get('t1', False)

password = request.POST.get('t2', False)

index = 0

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select \* FROM register where usertype='Student'")

rows = cur.fetchall()

for row in rows:

if row[0] == username and password == row[1]:

uname = username

index = 1

break

if index == 1:

context= {'data':'welcome '+username}

return render(request, 'StudentScreen.html', context)

else:

context= {'data':'login failed'}

return render(request, 'StudentLogin.html', context)

def FacultyLoginAction(request):

if request.method == 'POST':

global uname

username = request.POST.get('t1', False)

password = request.POST.get('t2', False)

index = 0

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select \* FROM register where usertype='Faculty'")

rows = cur.fetchall()

for row in rows:

if row[0] == username and password == row[1]:

uname = username

index = 1

break

if index == 1:

context= {'data':'welcome '+username}

return render(request, 'FacultyScreen.html', context)

else:

context= {'data':'login failed'}

return render(request, 'FacultyLogin.html', context)

def SignupAction(request):

if request.method == 'POST':

username = request.POST.get('t1', False)

password = request.POST.get('t2', False)

contact = request.POST.get('t3', False)

email = request.POST.get('t4', False)

address = request.POST.get('t5', False)

utype = request.POST.get('t6', False)

status = "none"

con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

with con:

cur = con.cursor()

cur.execute("select \* FROM register")

rows = cur.fetchall()

for row in rows:

if row[0] == username:

status = "Username already exists"

break

if status == "none":

db\_connection = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root', database = 'learning',charset='utf8')

db\_cursor = db\_connection.cursor()

student\_sql\_query = "INSERT INTO register(username,password,contact,email,address,usertype) VALUES('"+username+"','"+password+"','"+contact+"','"+email+"','"+address+"','"+utype+"')"

db\_cursor.execute(student\_sql\_query)

db\_connection.commit()

print(db\_cursor.rowcount, "Record Inserted")

if db\_cursor.rowcount == 1:

status = "Signup Process Completed. You can Login now"

context= {'data': status}

return render(request, 'Signup.html', context)

**6. TESTING**

**Implementation and Testing:**

Implementation is one of the most important tasks in project is the phase in which one has to be cautions because all the efforts undertaken during the project will be very interactive. Implementation is the most crucial stage in achieving successful system and giving the users confidence that the new system is workable and effective. Each program is tested individually at the time of development using the sample data and has verified that these programs link together in the way specified in the program specification. The computer system and its environment are tested to the satisfaction of the user.

**Implementation**

The implementation phase is less creative than system design. It is primarily concerned with user training, and file conversion. The system may be requiring extensive user training. The initial parameters of the system should be modifies as a result of a programming. A simple operating procedure is provided so that the user can understand the different functions clearly and quickly. The different reports can be obtained either on the inkjet or dot matrix printer, which is available at the disposal of the user. The proposed system is very easy to implement. In general implementation is used to mean the process of converting a new or revised system design into an operational one.

## **Testing**

Testing is the process where the test data is prepared and is used for testing the modules individually and later the validation given for the fields. Then the system testing takes place which makes sure that all components of the system property function as a unit. The test data should be chosen such that it passed through all possible condition. Actually, testing is the state of implementation which aimed at ensuring that the system works accurately and efficiently before the actual operation commence. The following is the description of the testing strategies, which were carried out during the testing period.

### **System Testing**

Testing has become an integral part of any system or project especially in the field of information technology. The importance of testing is a method of justifying, if one is ready to move further, be it to be check if one is capable to with stand the rigors of a particular situation cannot be underplayed and that is why testing before development is so critical. When the software is developed before it is given to user to use the software must be tested whether it is solving the purpose for which it is developed. This testing involves various types through which one can ensure the software is reliable. The program was tested logically and pattern of execution of the program for a set of data are repeated. Thus, the code was exhaustively checked for all possible correct data and the outcomes were also checked.

**Module Testing**

To locate errors, each module is tested individually. This enables us to detect error and correct it without affecting any other modules. Whenever the program is not satisfying the required function, it must be corrected to get the required result. Thus, all the modules are individually tested from bottom up starting with the smallest and lowest modules and proceeding to the next level. Each module in the system is tested separately. For example, the job classification module is tested separately. This module is tested with different job and its approximate execution time and the result of the test is compared with the results that are prepared manually. The comparison shows that the results proposed system works efficiently than the existing system. Each module in the system is tested separately. In this system the resource classification and job scheduling modules are tested separately and their corresponding results are obtained which reduces the process waiting time.

**Integration Testing**

After the module testing, the integration testing is applied. When linking the modules there may be chance for errors to occur, these errors are corrected by using this testing. In this system all modules are connected and tested. The testing results are very correct. Thus the mapping of jobs with resources is done correctly by the system.

**Acceptance Testing**

When that user fined no major problems with its accuracy, the system passers through a final acceptance test. This test confirms that the system needs the original goals, objectives and requirements established during analysis without actual execution which elimination wastage of time and money acceptance tests on the shoulders of users and management, it is finally acceptable and ready for the operation

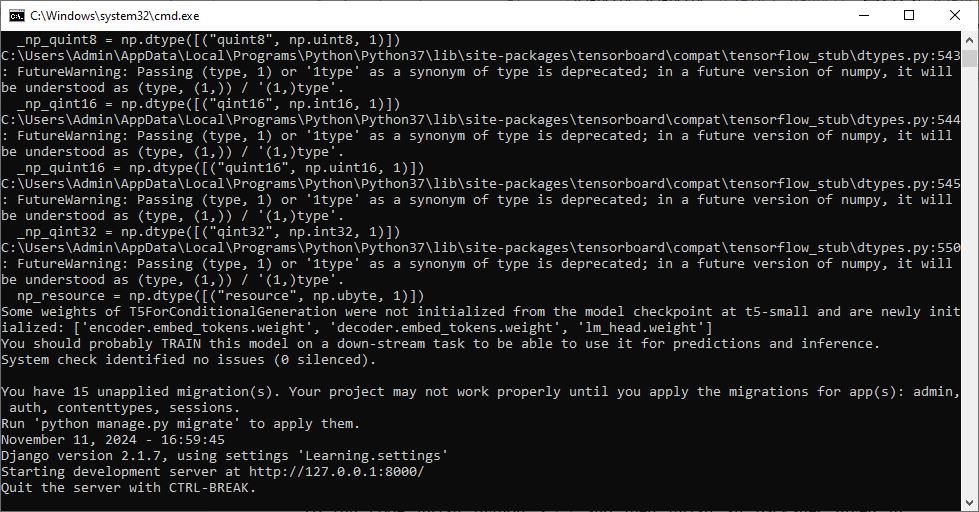
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case Id** | **Test Case Name** | **Test Case Desc.** | **Test Steps** | | | **Test Case Status** | **Test Priority** |
| **Step** | **Expected** | **Actual** |
| 01 | Faculty Login | Test whether Faculty Login or not into the system | If the Faculty Login may not uploaded | We cannot do further operations | Faculty Login we will do further operations | High | High |
| 02 | Generated Summary | Test whether the Generated Summary Successfully or not | If the Generated Summary may not Run Successfully | We cannot do further operations | we will do further  operations | High | High |
| 03 | Subjective Question | Test whether Subjective Question Successfully or not | If the  Subjective Question may not Run Successfully | We cannot do further operations | we will do further  operations | High | High |
| 04 | Multiple Choice Question | Test whether Multiple Choice Question Successfully or not | If the  Multiple Choice Question may not Run Successfully | We cannot do further operations | we will do further  operations | High | High |
| 05 | View Marks | Test whether View Marks Successfully or not | If the  View Marks may not Run Successfully | We cannot do further operations | we will do further  operations | High | High |
| 06 | Model Analysis | Test Model Analysis whether Successfully or not | If the  Model Analysis may not Successfully | We cannot do further operations | we will do further  operations | High | High |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case Id** | **Test Case Name** | **Test Case Desc.** | **Test Steps** | | | **Test Case Status** | **Test Priority** |
| **Step** | **Expected** | **Actual** |
| 01 | Students Login | Test whether Students Login or not into the system | If the Students Login may not uploaded | We cannot do further operations | Students Login we will do further operations | High | High |
| 02 | Generate Summary | Test whether the Generate Summary Successfully or not | If the Generate Summary may not Run Successfully | We cannot do further operations | we will do further  operations | High | High |
| 03 | Write Exam | Test whether Write Exam Successfully or not | If the  Write Exam may not Run Successfully | We cannot do further operations | we will do further  operations | High | High |
| 04 | View Marks | Test whether View Marks Successfully or not | If the  View Marks may not Run Successfully | We cannot do further operations | we will do further  operations | High | High |

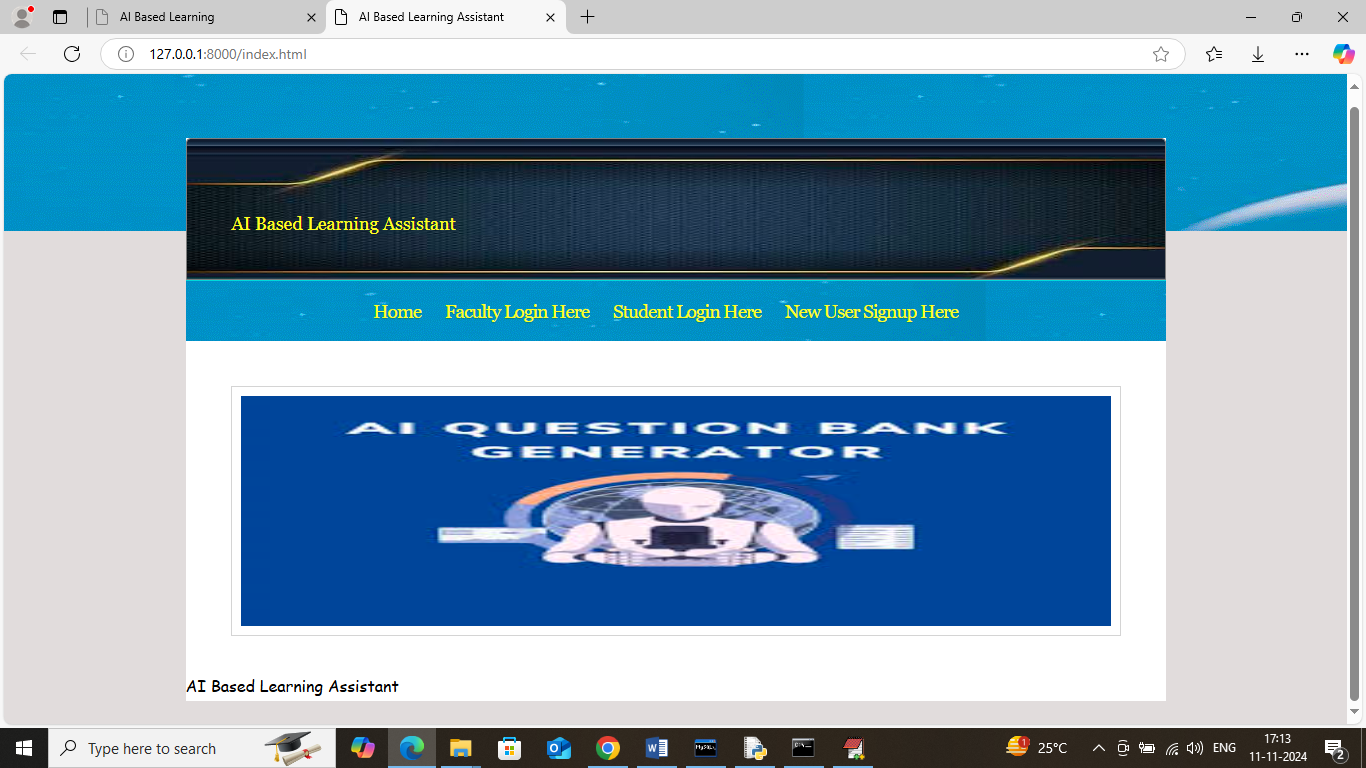
**7.SCREENSHOTS**:

To run code install python 3.7.2 and then install all packages given in requirements.txt file and then double click on ‘run.bat’ file to start server and get below page.

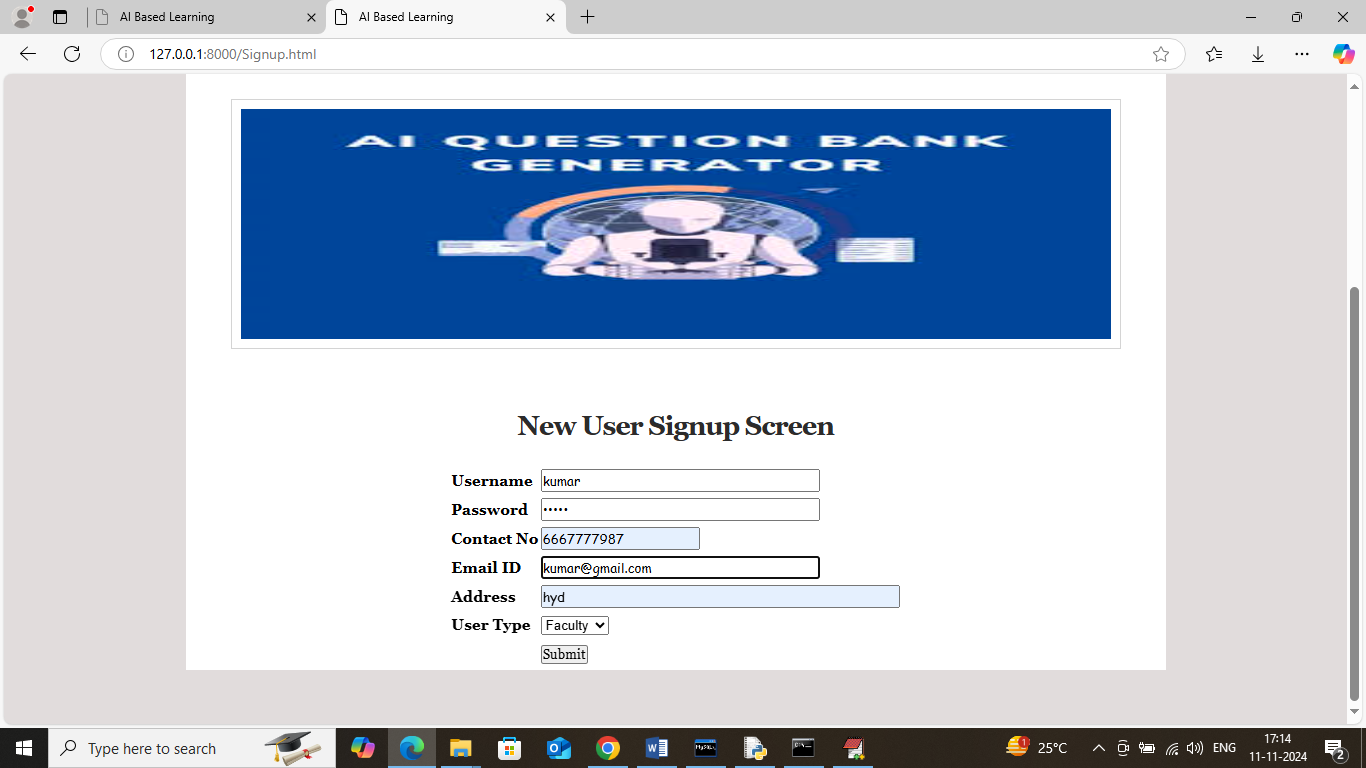
Note: summary and question generations models are very heavy and it will take 3 minutes to start server and will take 6 minutes to generate question and answers in normal CPU system. In GPU it will run faster.



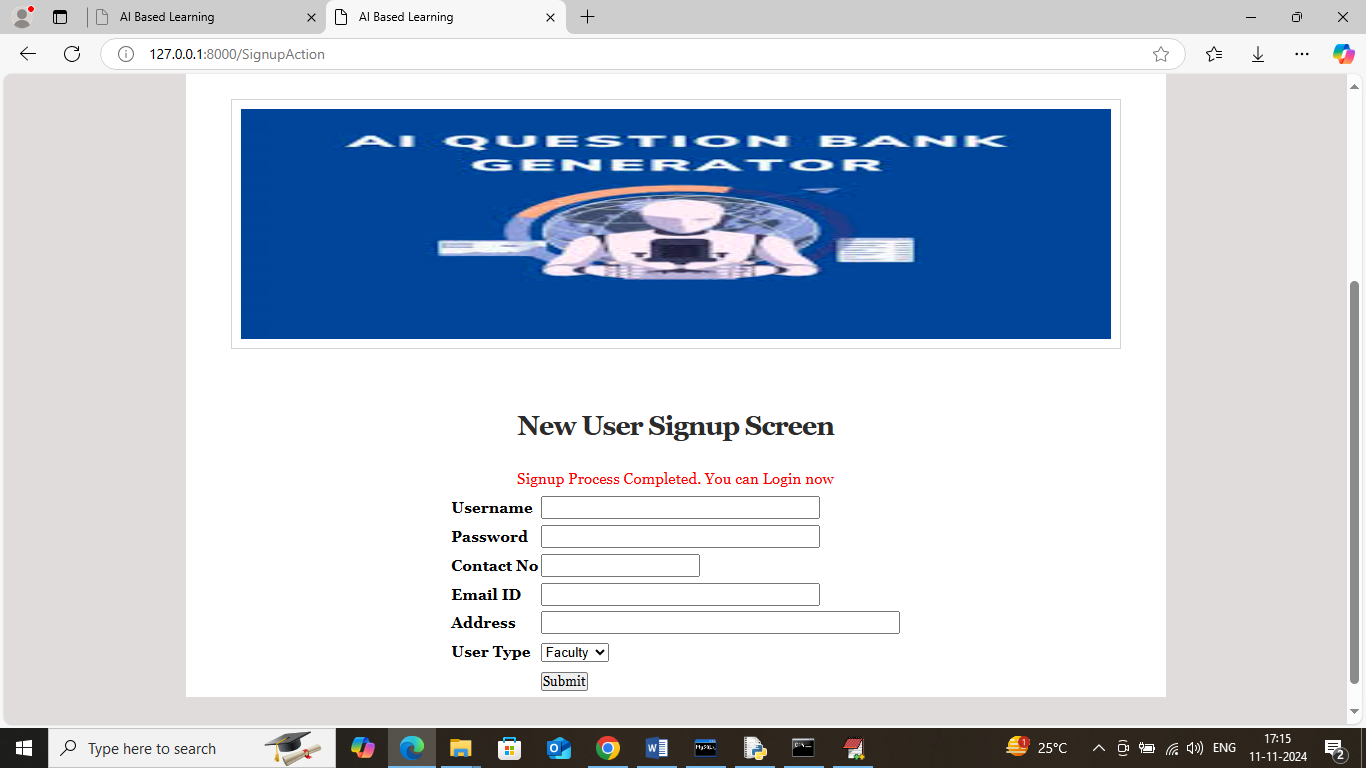
In above screen python server started and now open browser and enter URL as <http://127.0.0.1:8000/index.html> and then press enter button to get below page



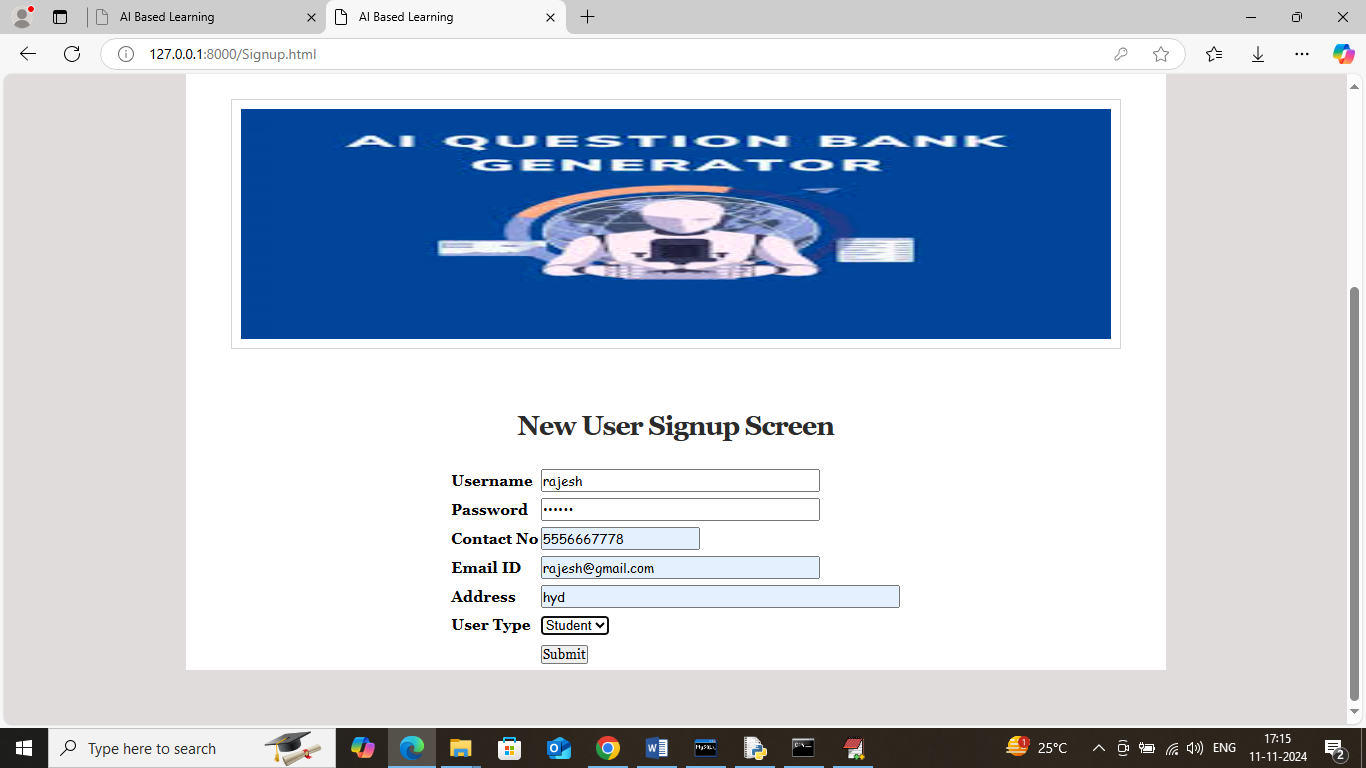
In above screen click on ‘New User Sign up Here’ link to get below page



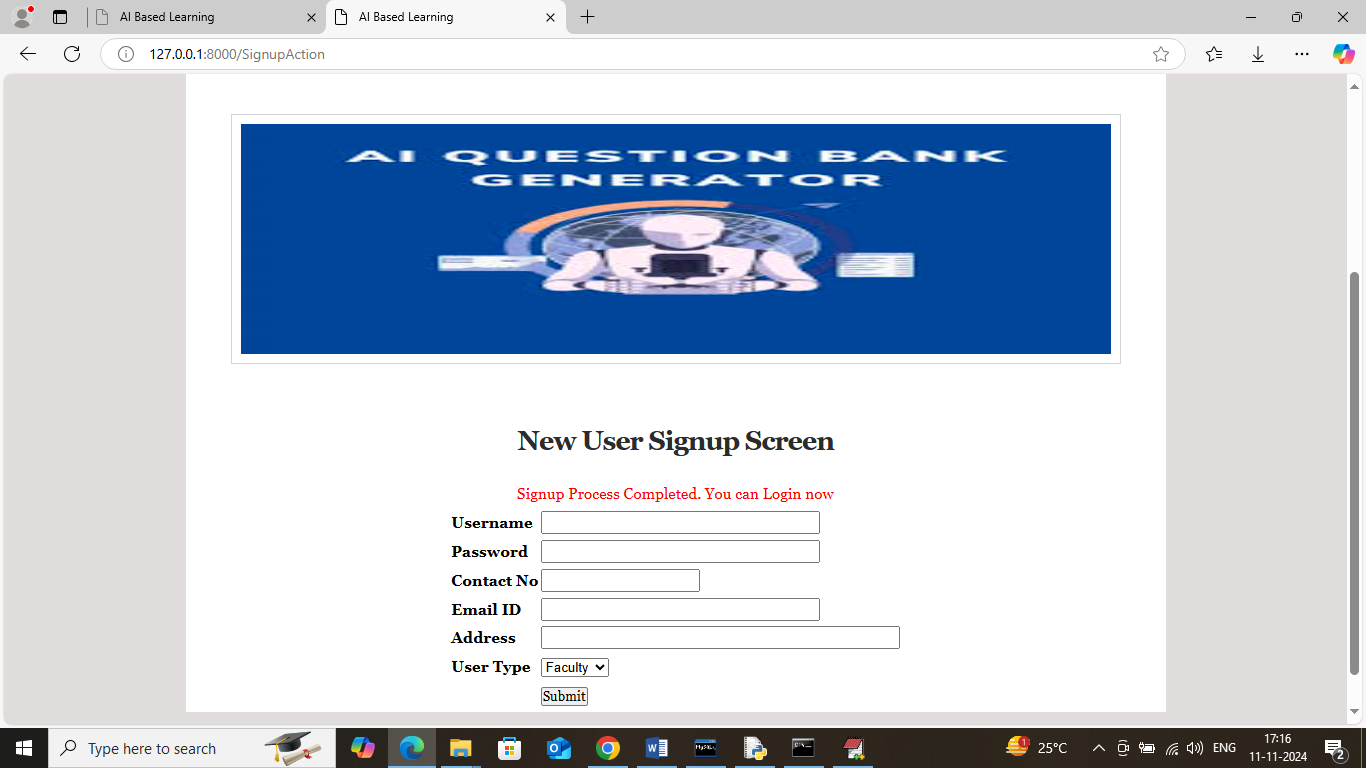
In above screen entering faculty sign up details and then press button to get below page



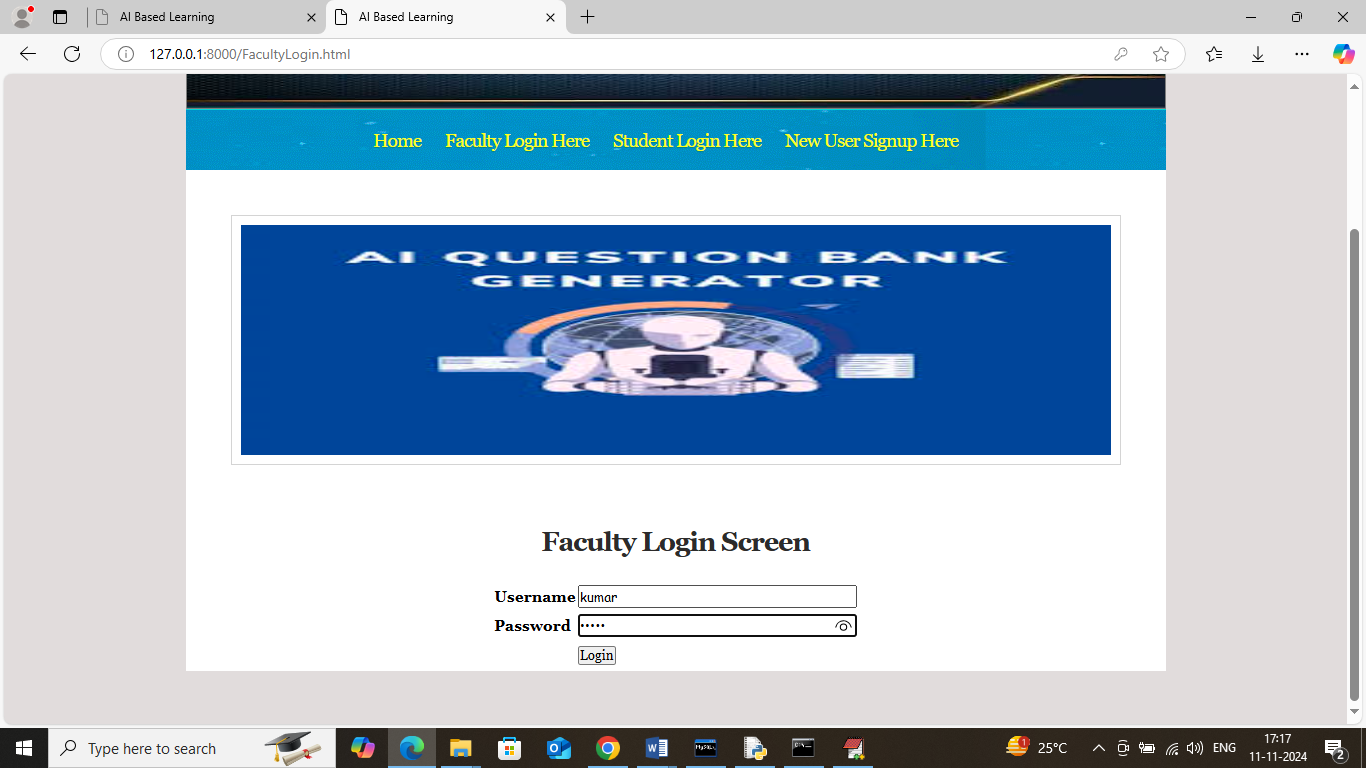
In above screen faculty sign up completed and similarly add student details



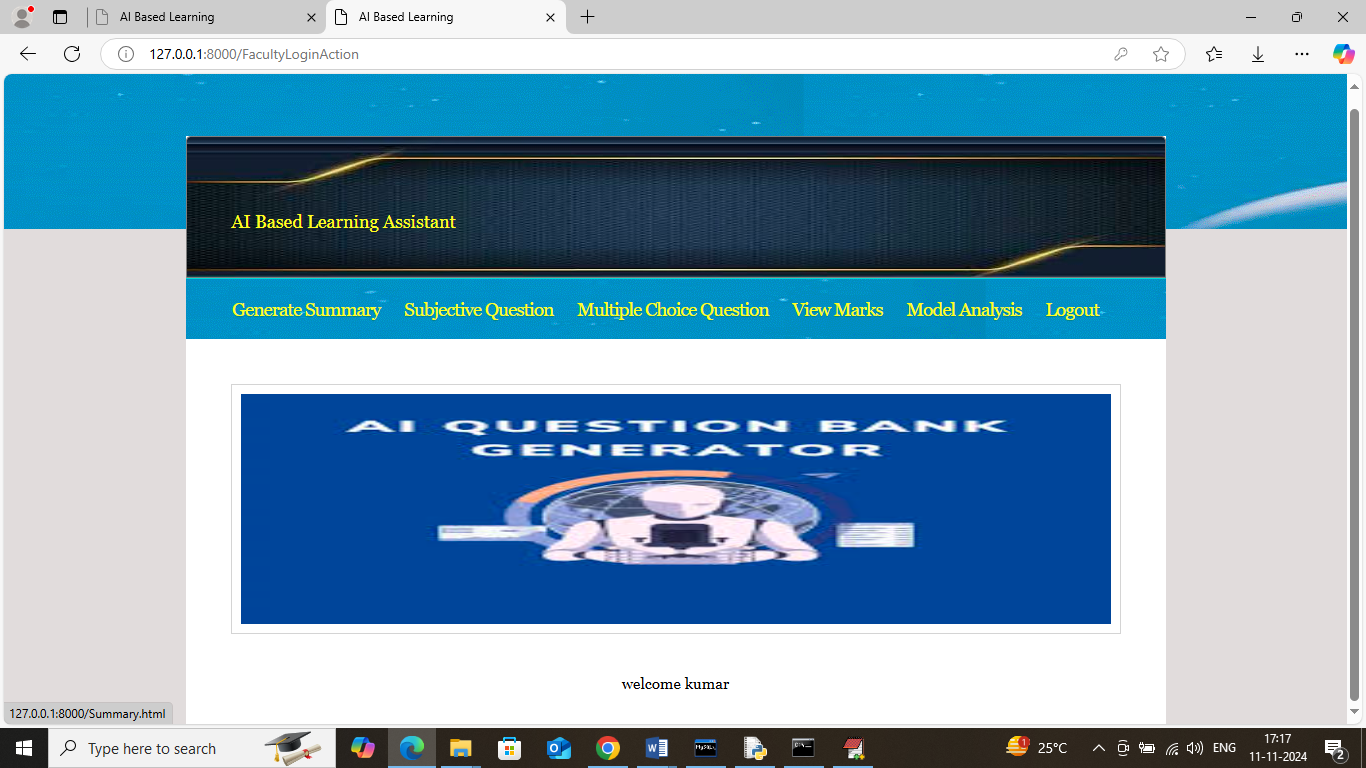
In above screen adding sign up details and then press button to get below page



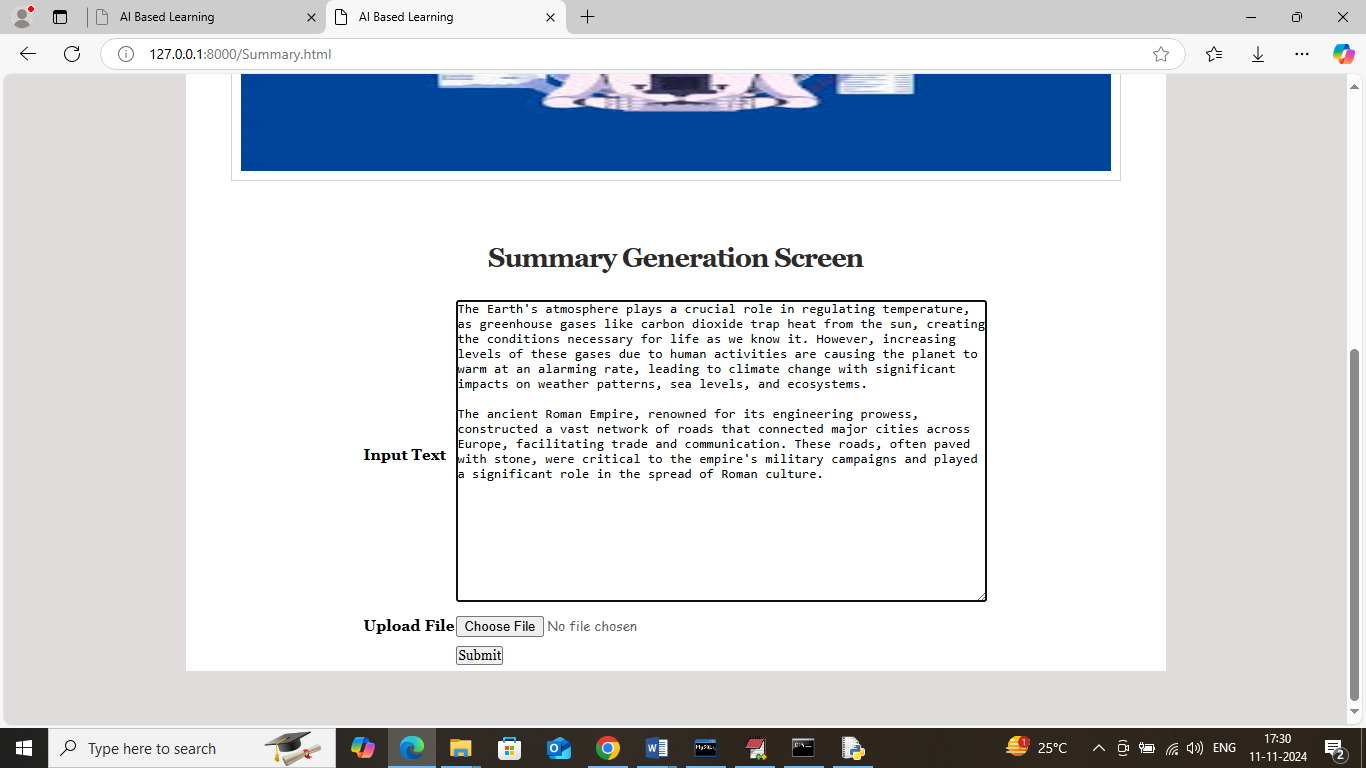
In above screen student sign up completed and now click on ‘Faculty Login’ link to get below page



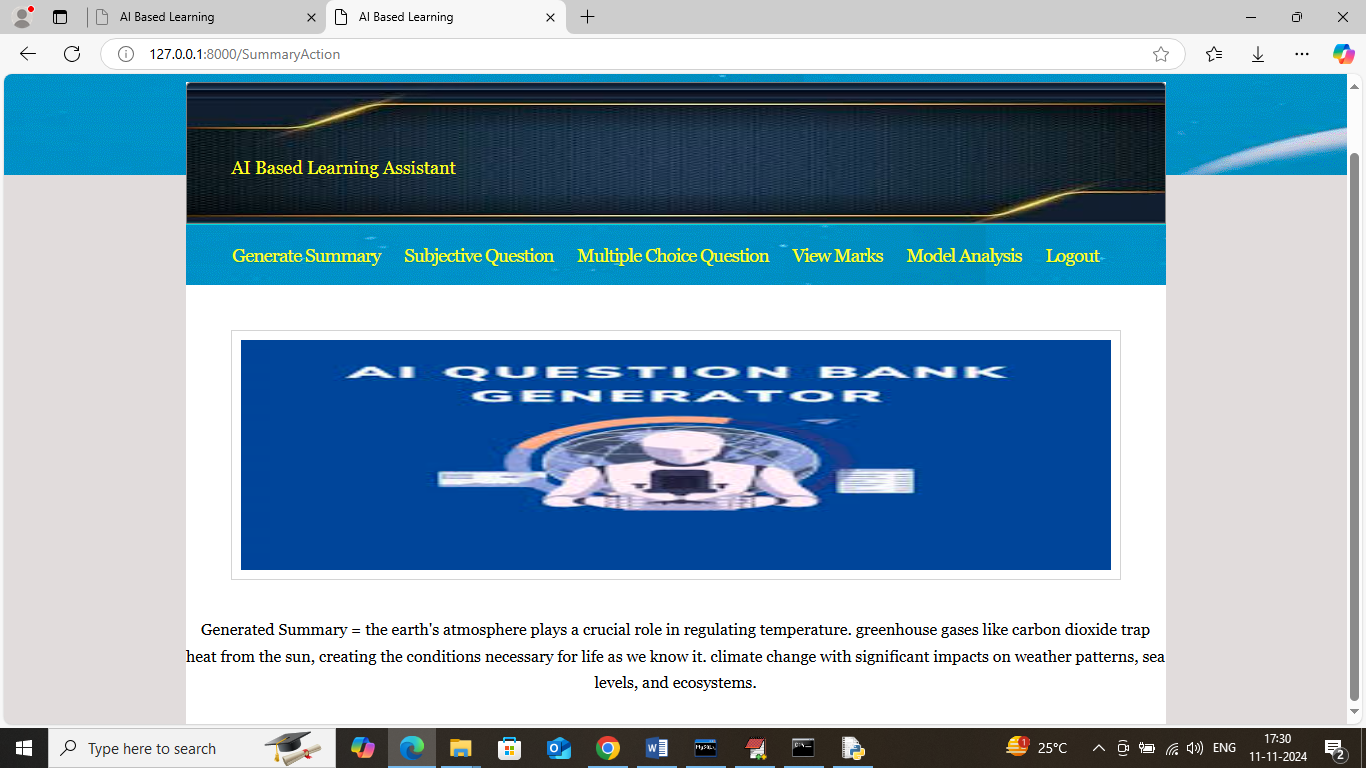
In above screen faculty is login and after login will get below page



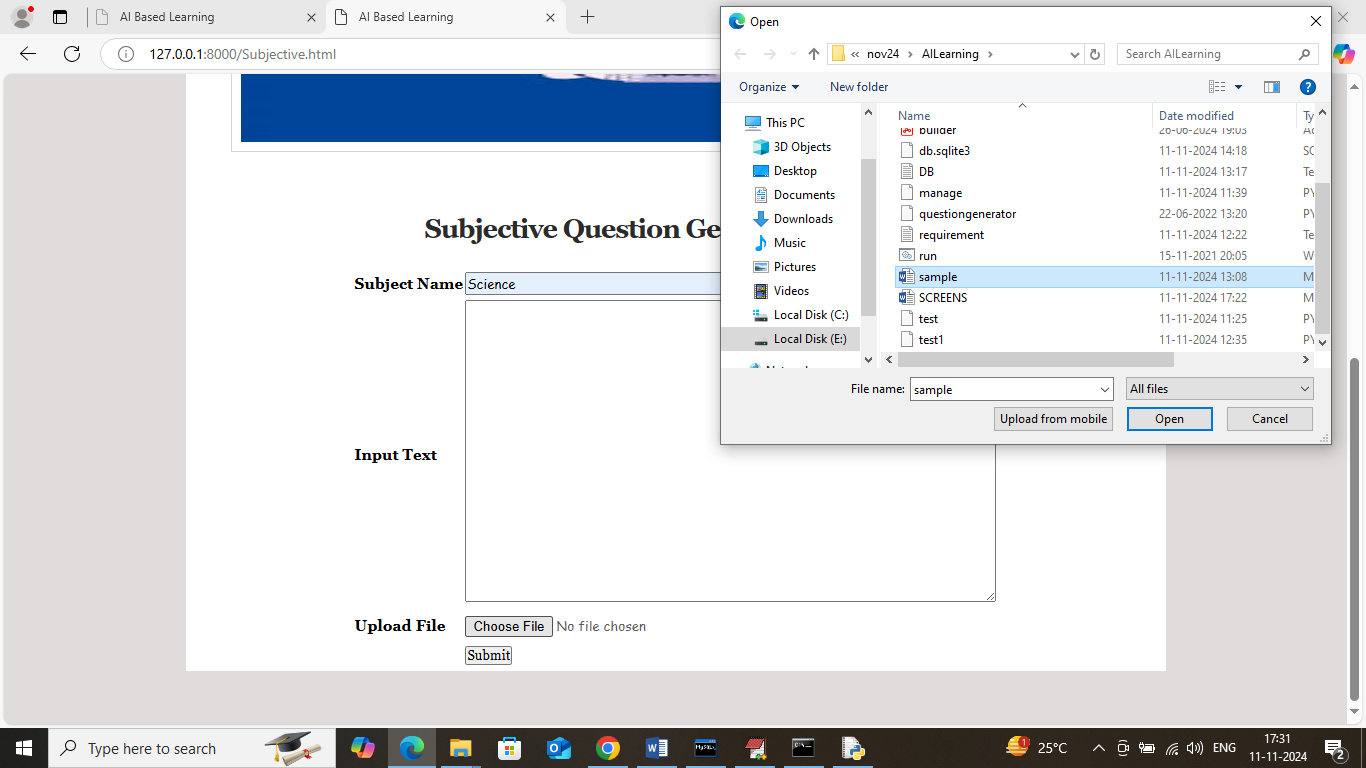
In above screen click on ‘Generated Summary’ link to get below page



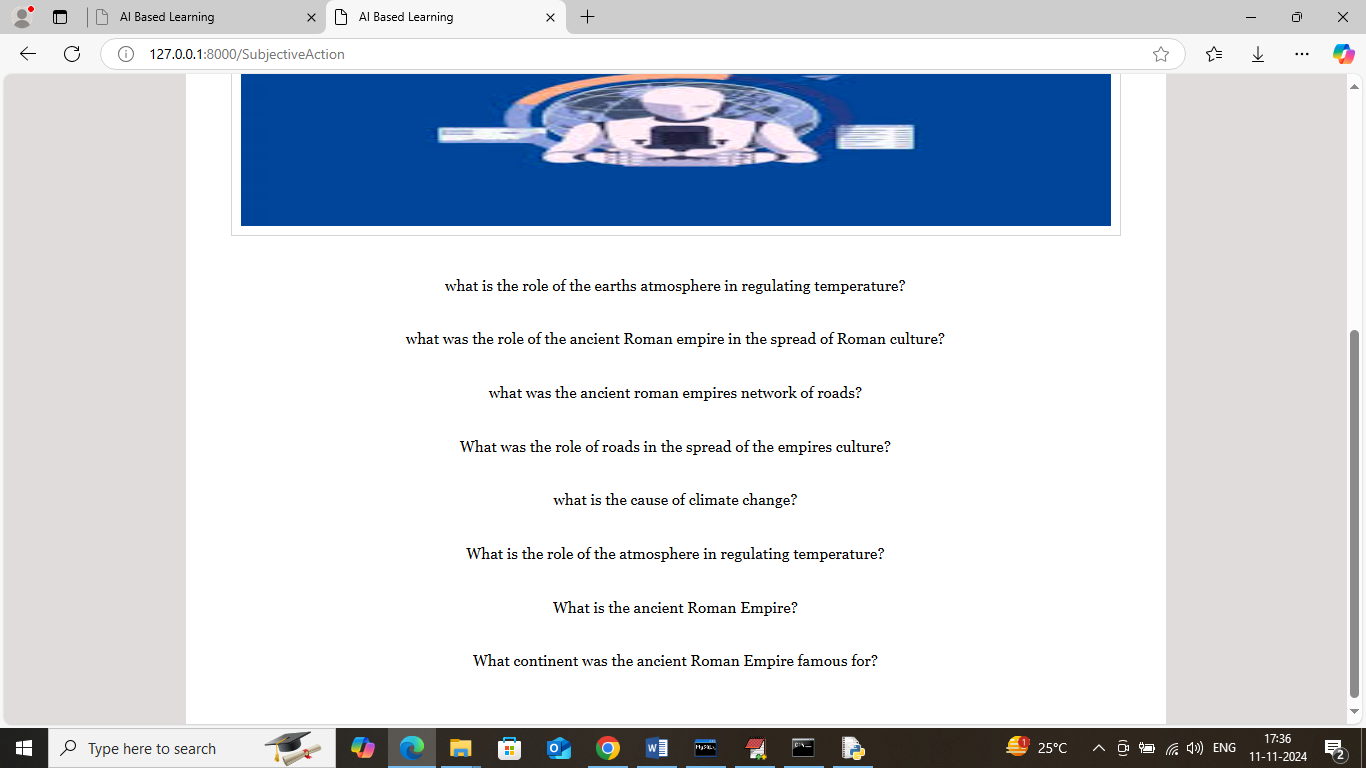
In above screen pasting some text and then click on button to get below page with summary



In above screen can see summary generated from given text and similarly you can upload and test small word or PDF file. Now click on ‘Subjective Question’ link to get below page



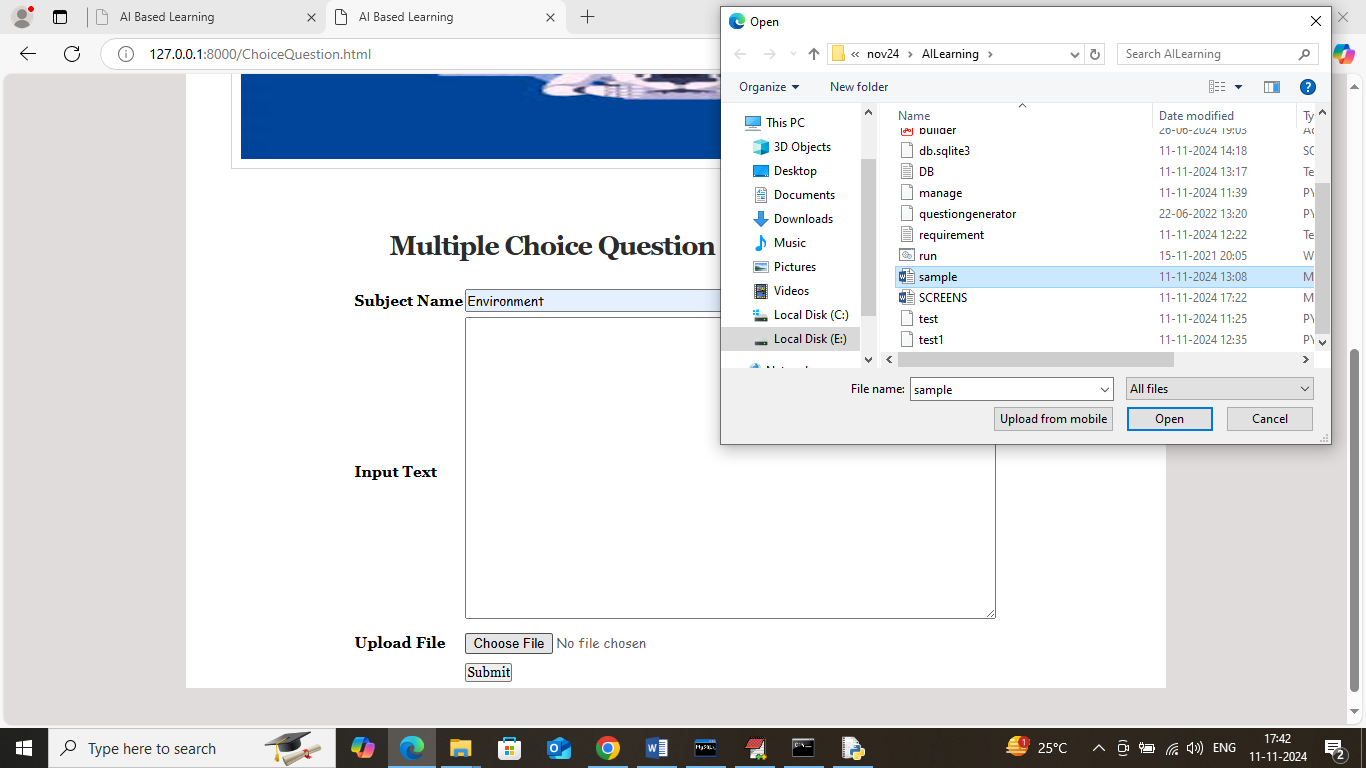
In above screen enter subject name and then upload word or PDF file or enter text in text area and then click button to get below generated questions and all this generated questions and answers will saved in database.



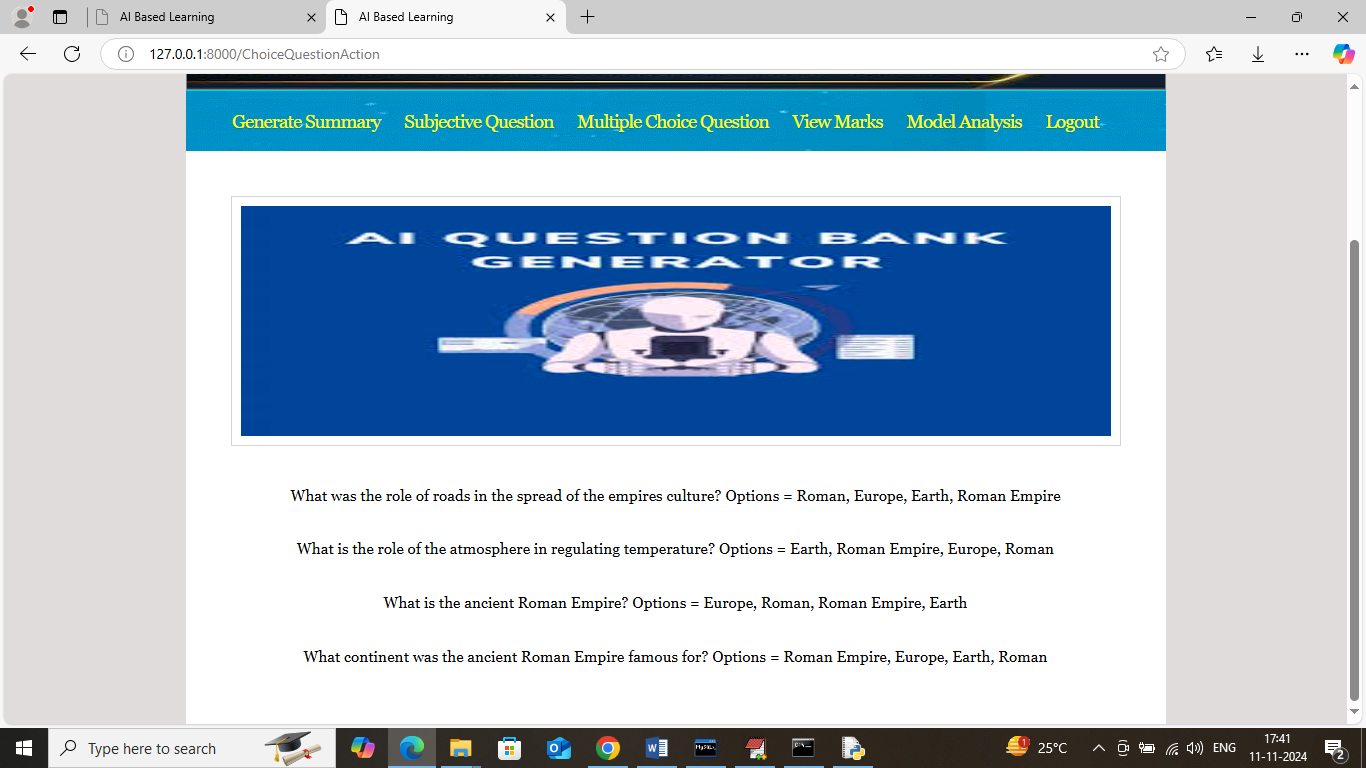
In above screen can see generated question from given text and can see both question and answers in database like below screen



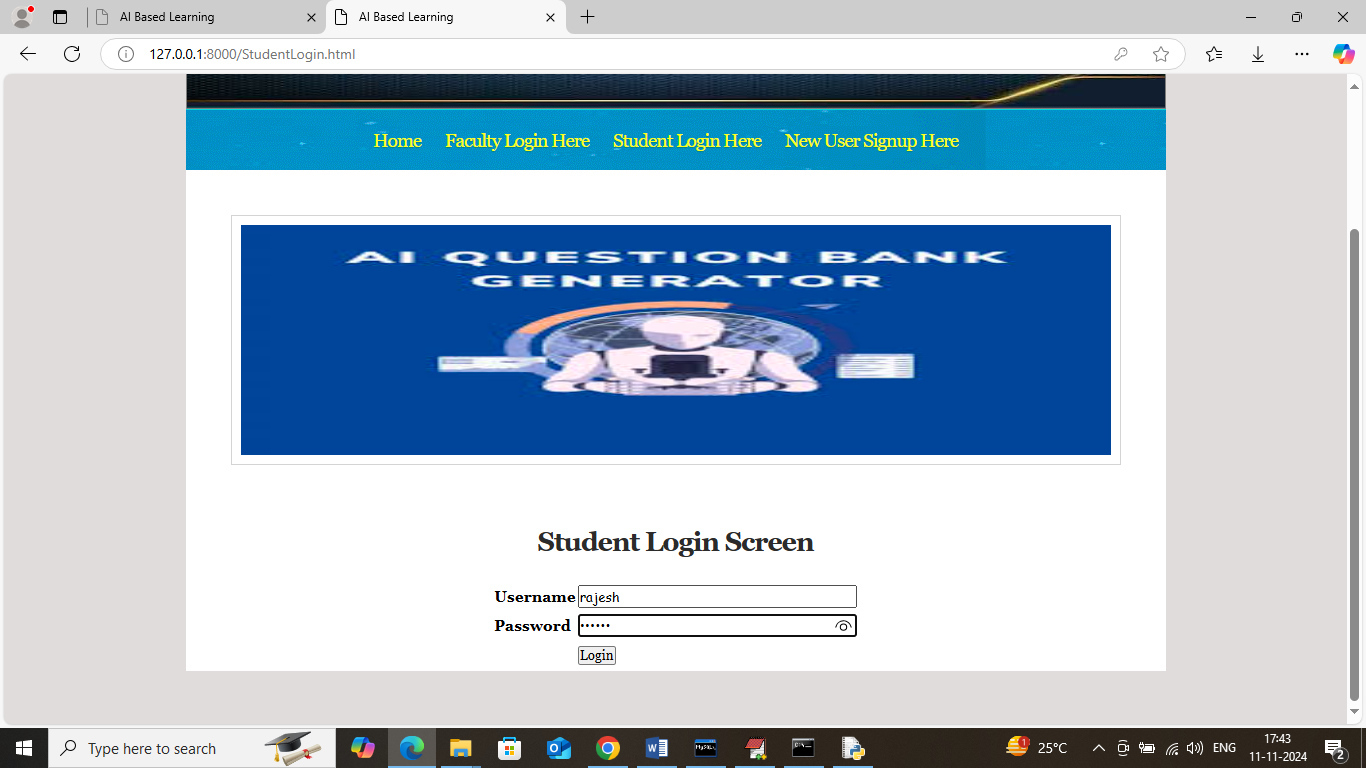
In above database screen can see generated question and answers from uploaded text and now in application click on ‘Multiple Choice Question’ link to get below page



In above screen selecting and uploading word file and then click on ‘Submit’ button to generate multiple choice question and get below page



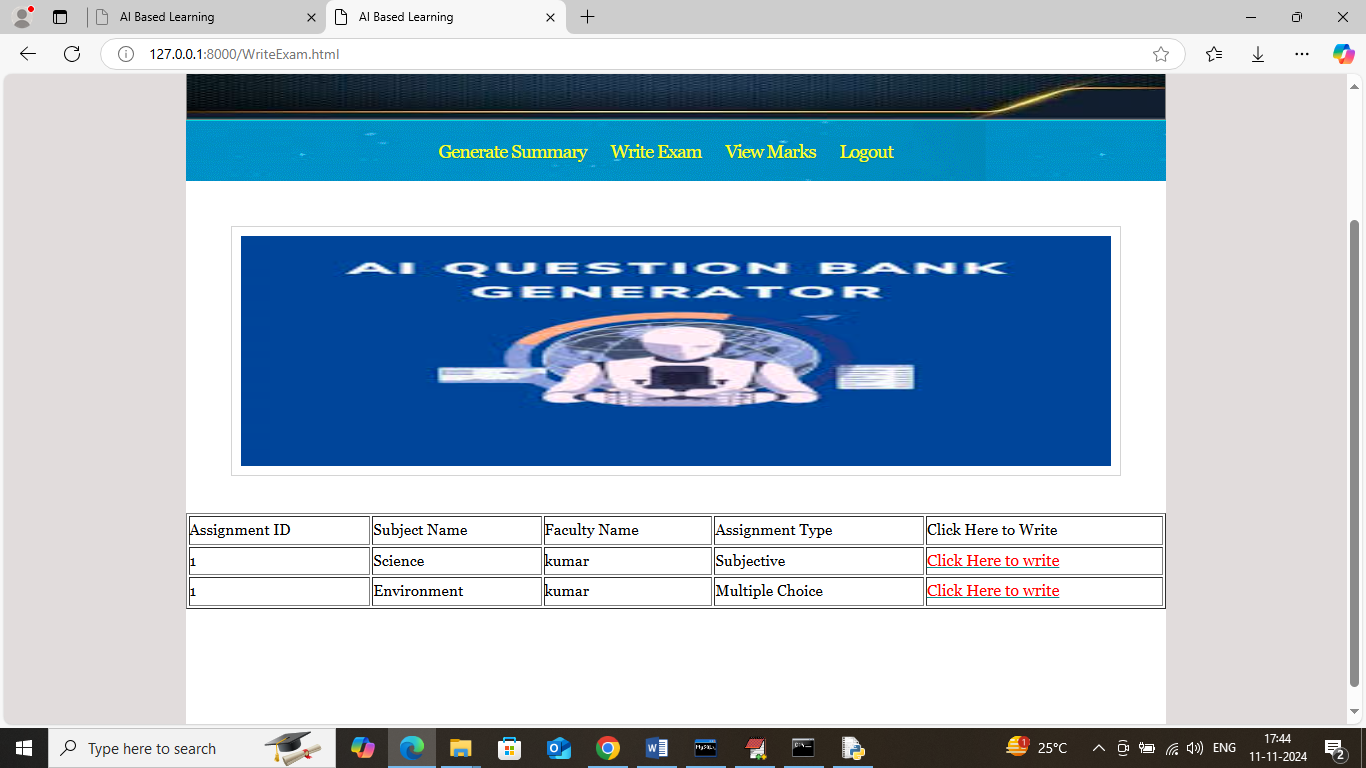
In above screen can see generated questions and their multiple choice options and this questions will saved in database so students can write exams. Now logout and login as students to write exam



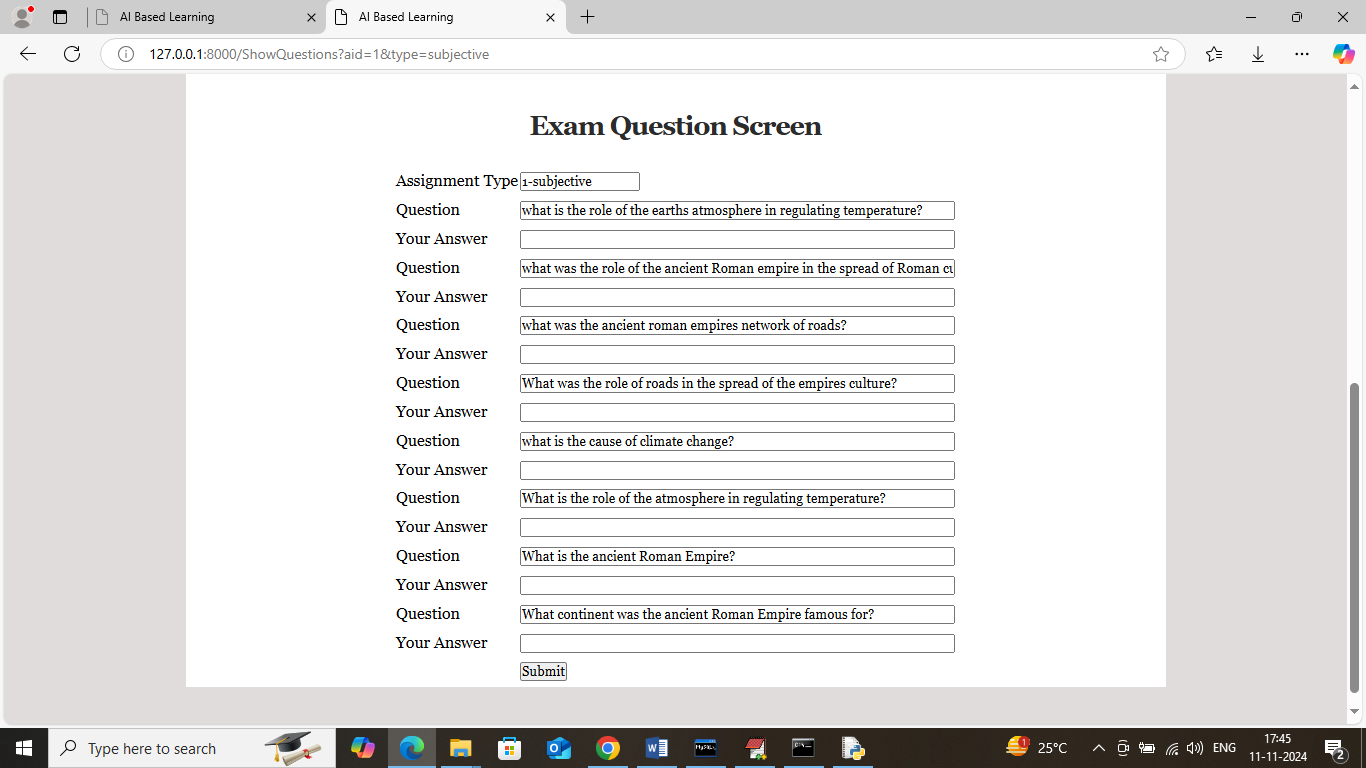
In above screen student is login and after login will get below page



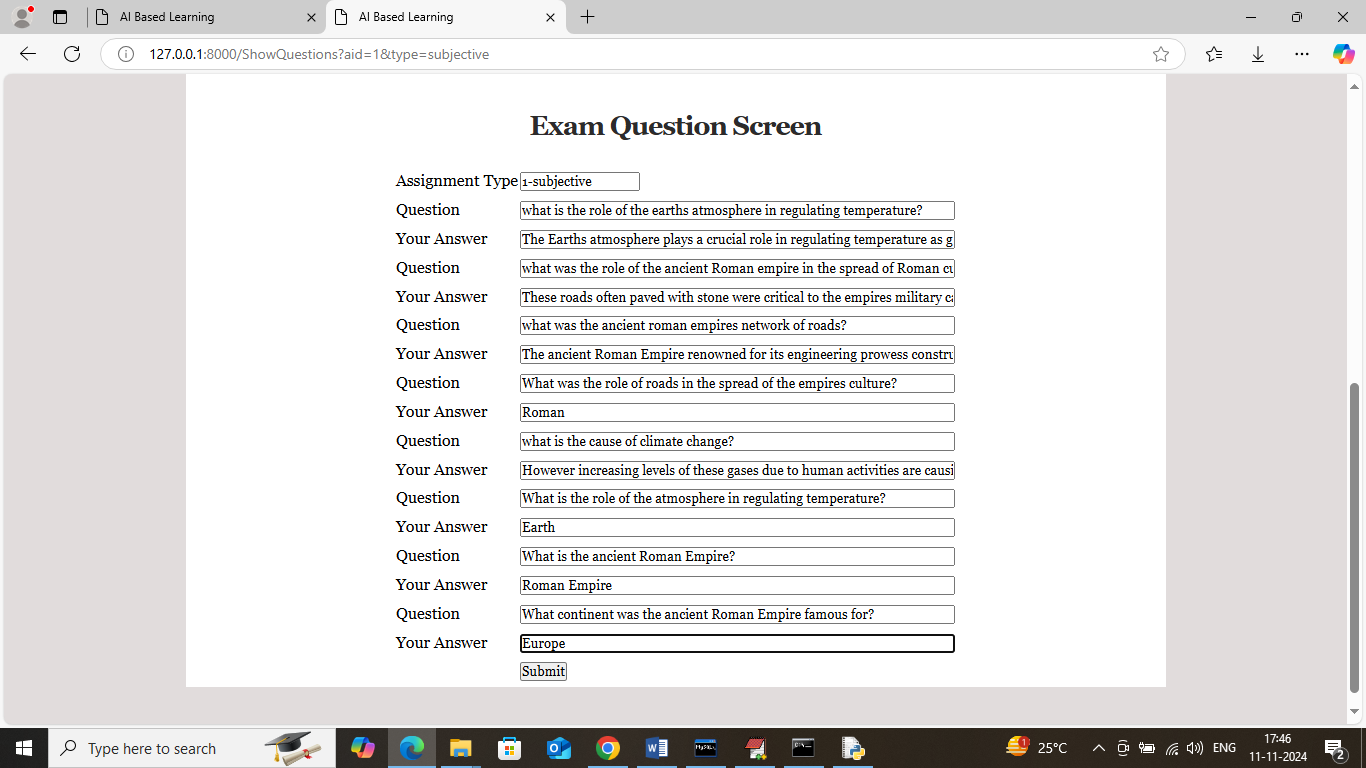
In above screen student can also generated summary for any text and now click on ‘Write Exam’ link to get below page.



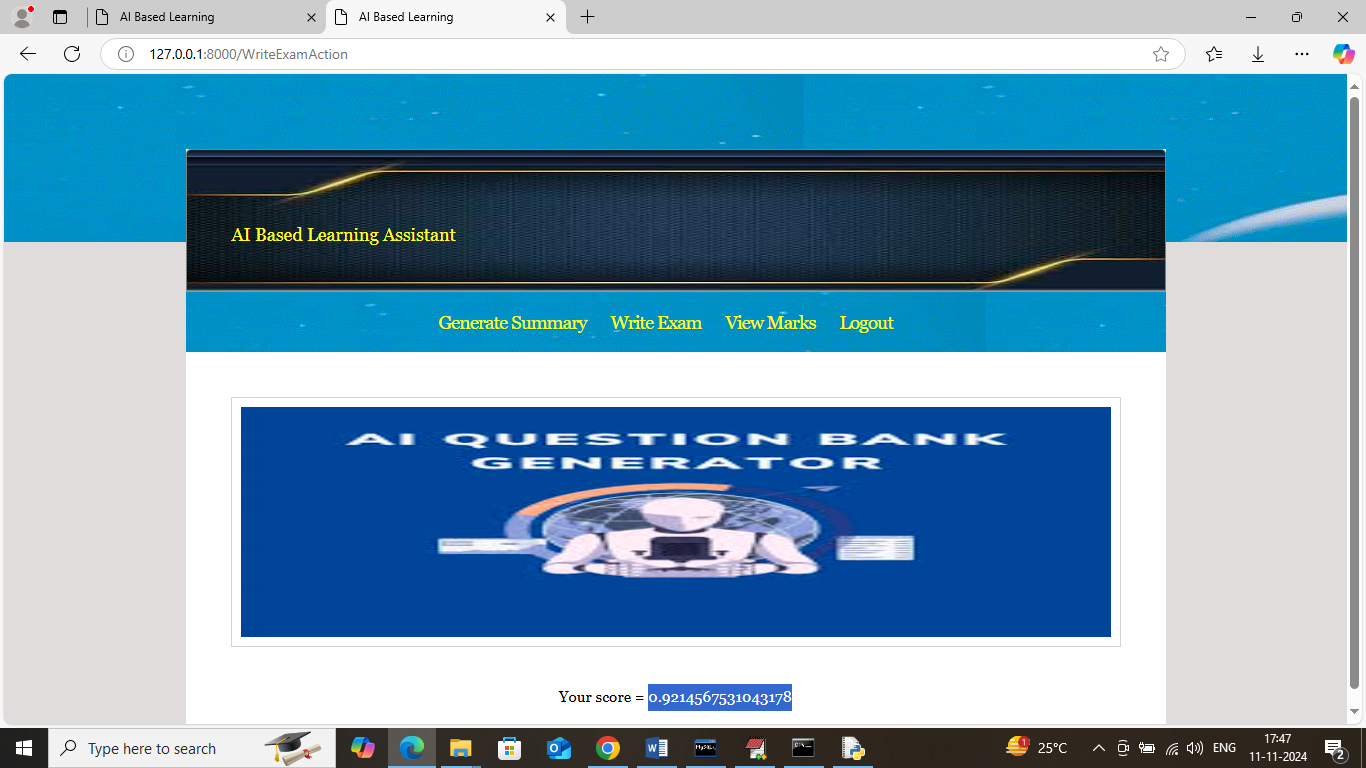
In above screen application will display number of subjective and multiple choice questions assignment and student can click on ‘Click Here’ desired link to get questions and write exam



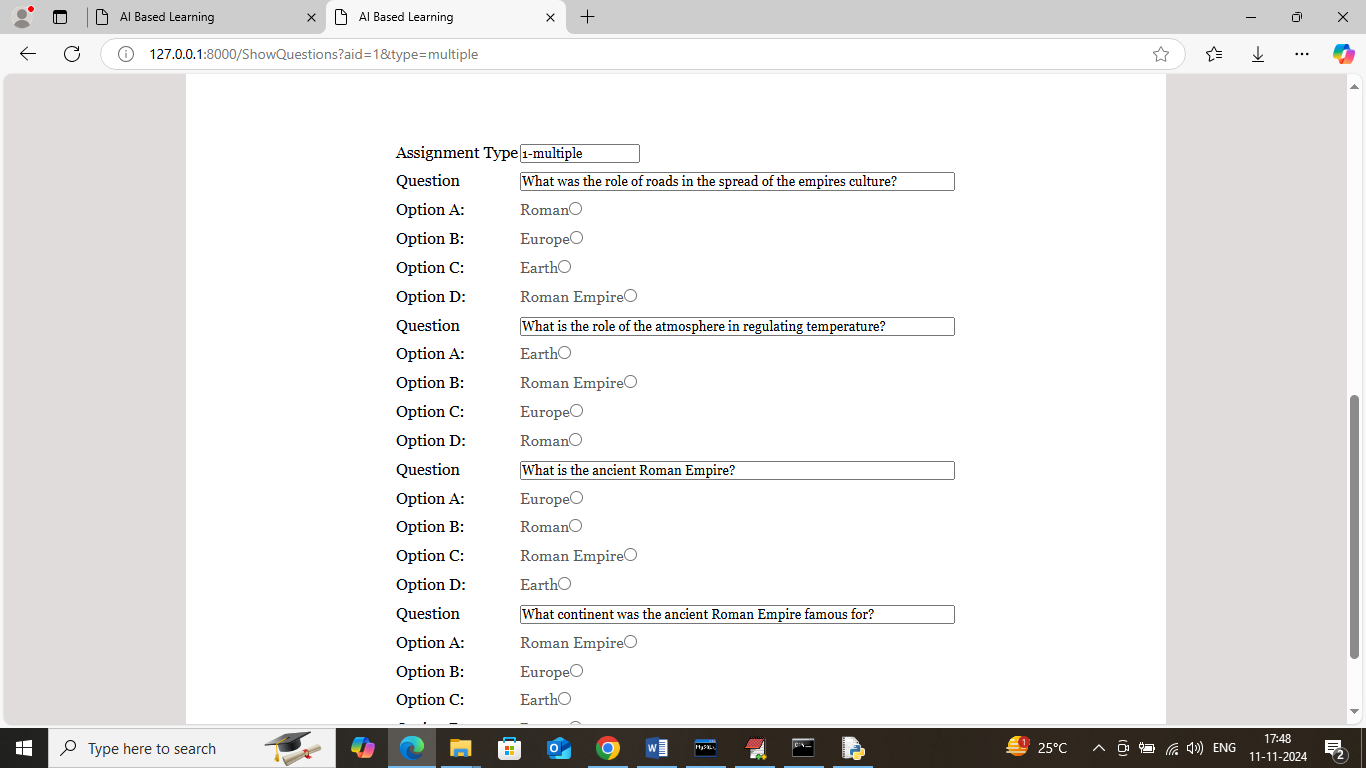
In above screen student can see all questions and write answers for each question



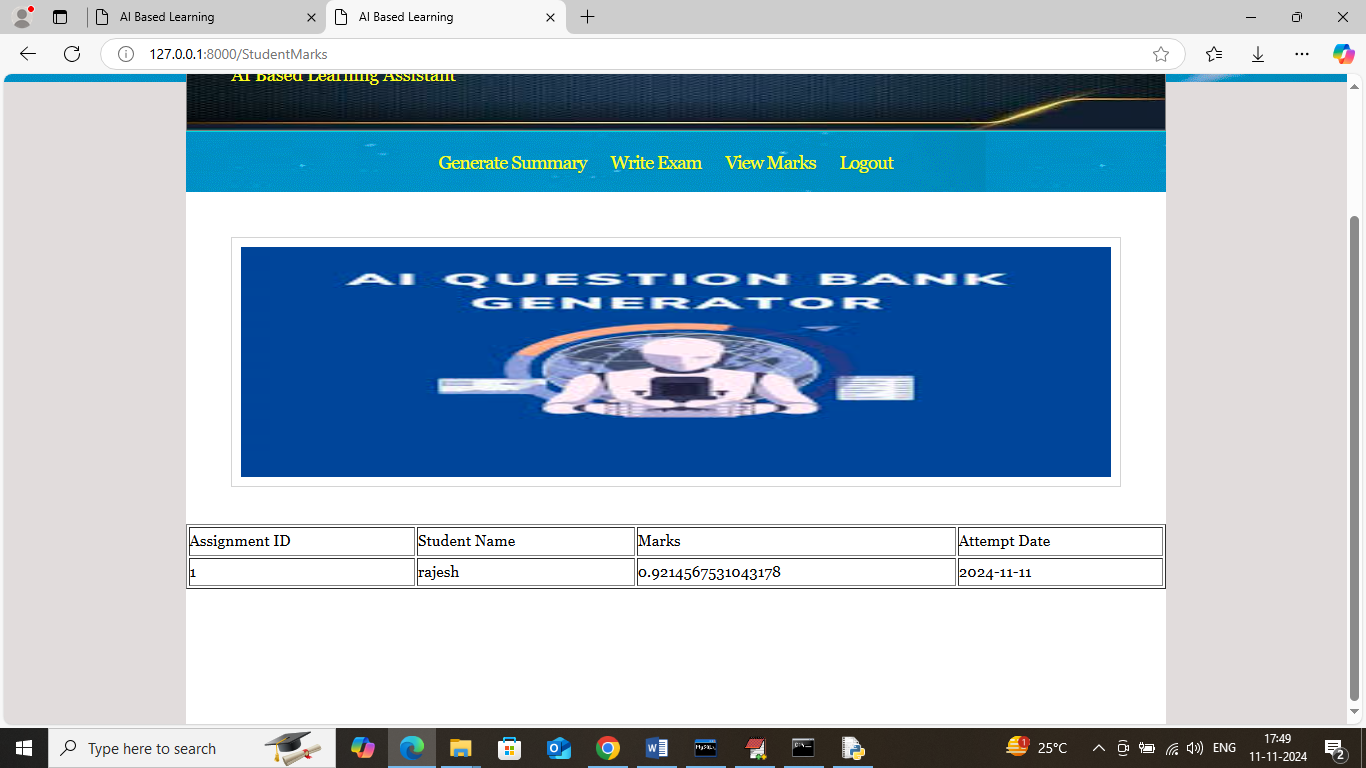
In above screen student will write answers for each question and then press button to allow ML algorithm to evaluate both database and students answers to get below score



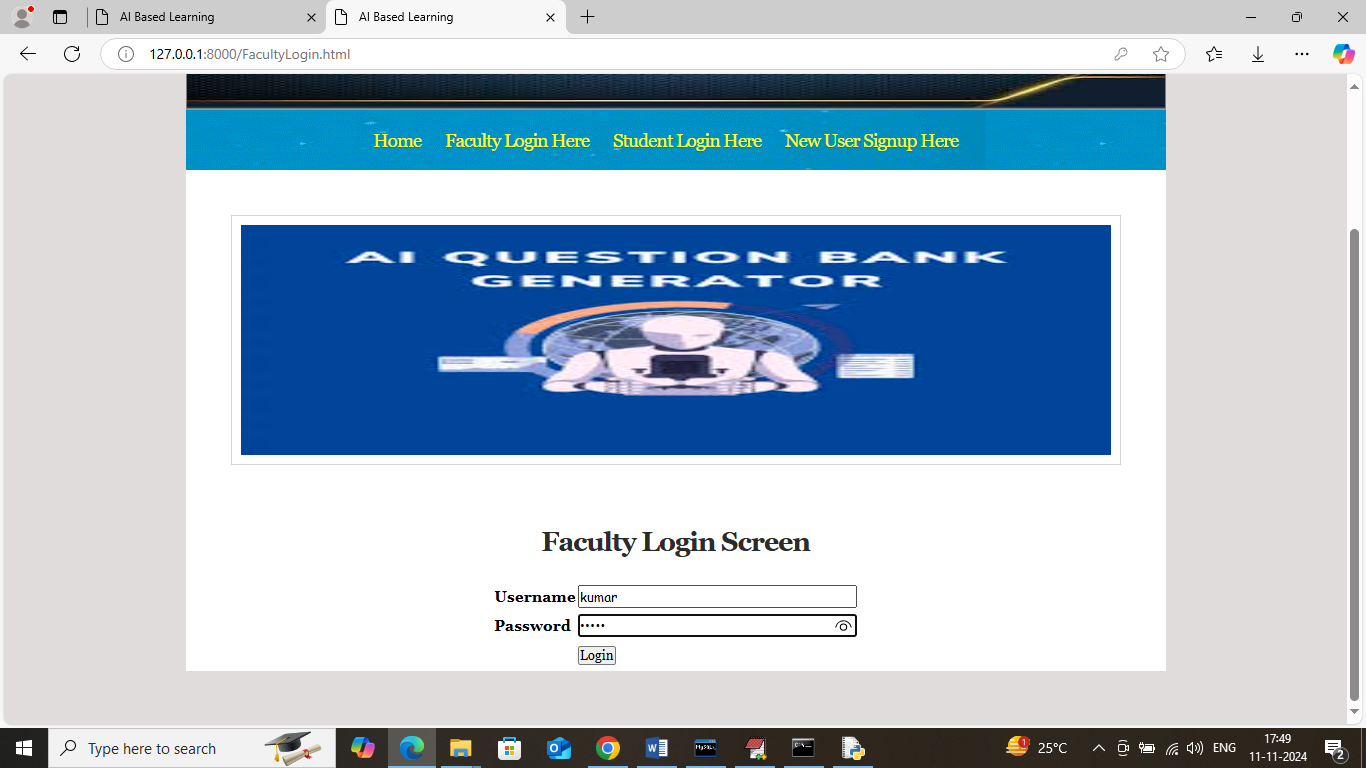
In above screen student got 92% score and similarly students can select multiple choice option to write multiple choice exam



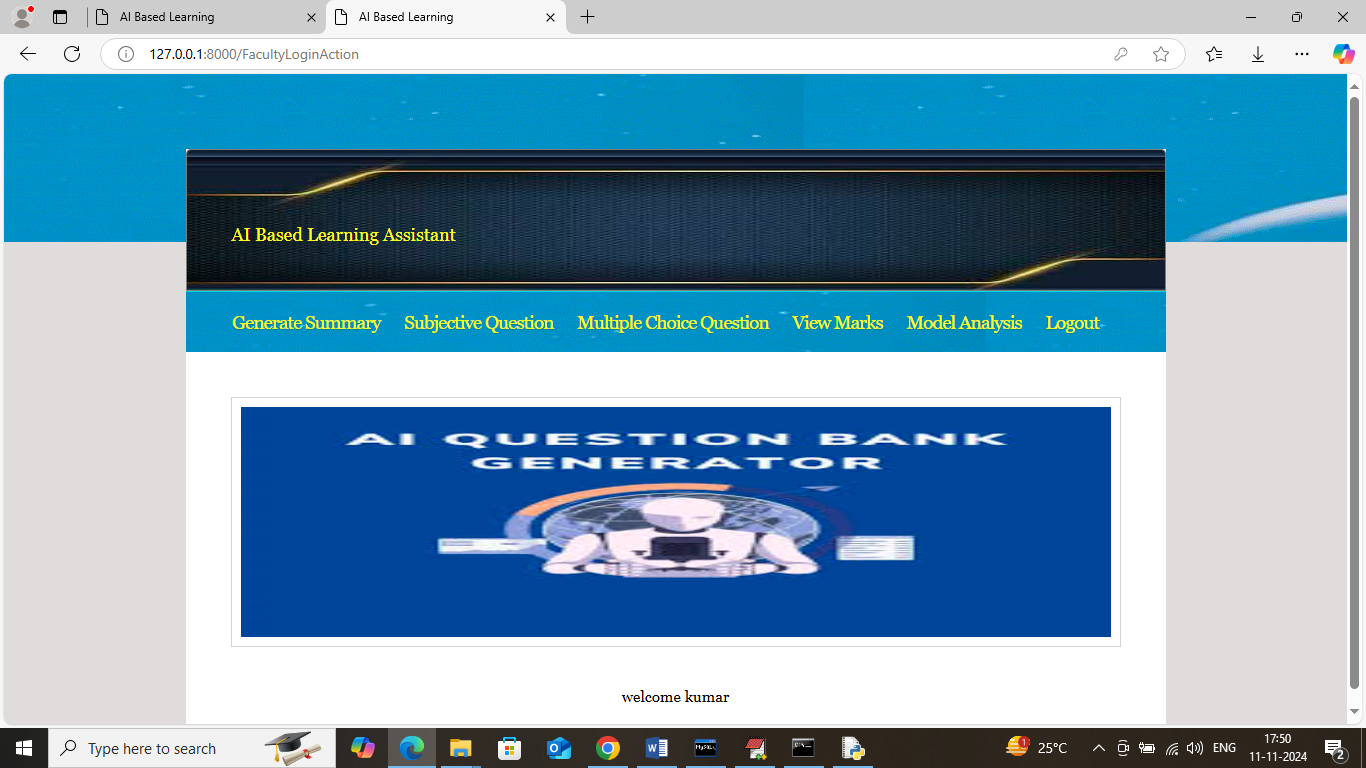
In above screen students can see all questions along with answers and can write exam to get score and now click on ‘View Marks’ link to get below page



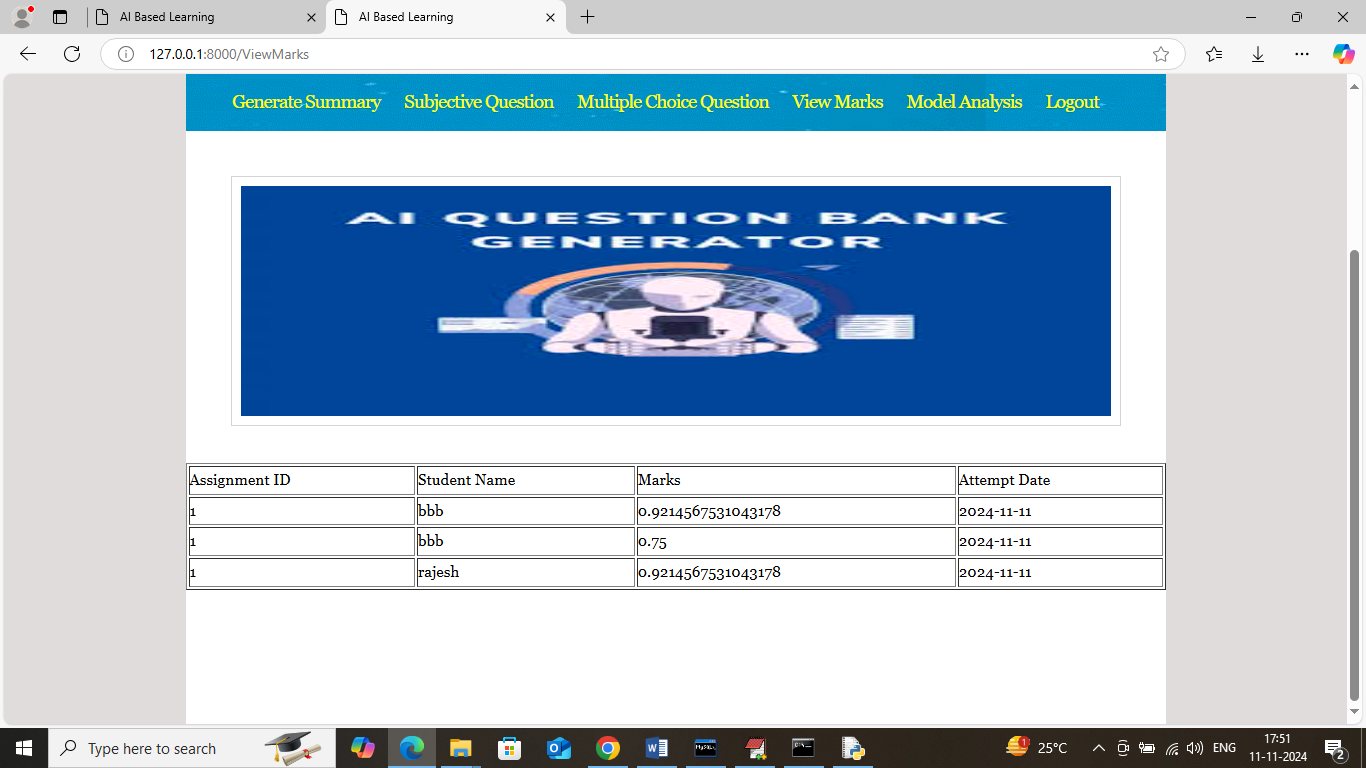
In above screen students can see marks obtained in given assignment ID and now logout and login as faculty to perform model analysis



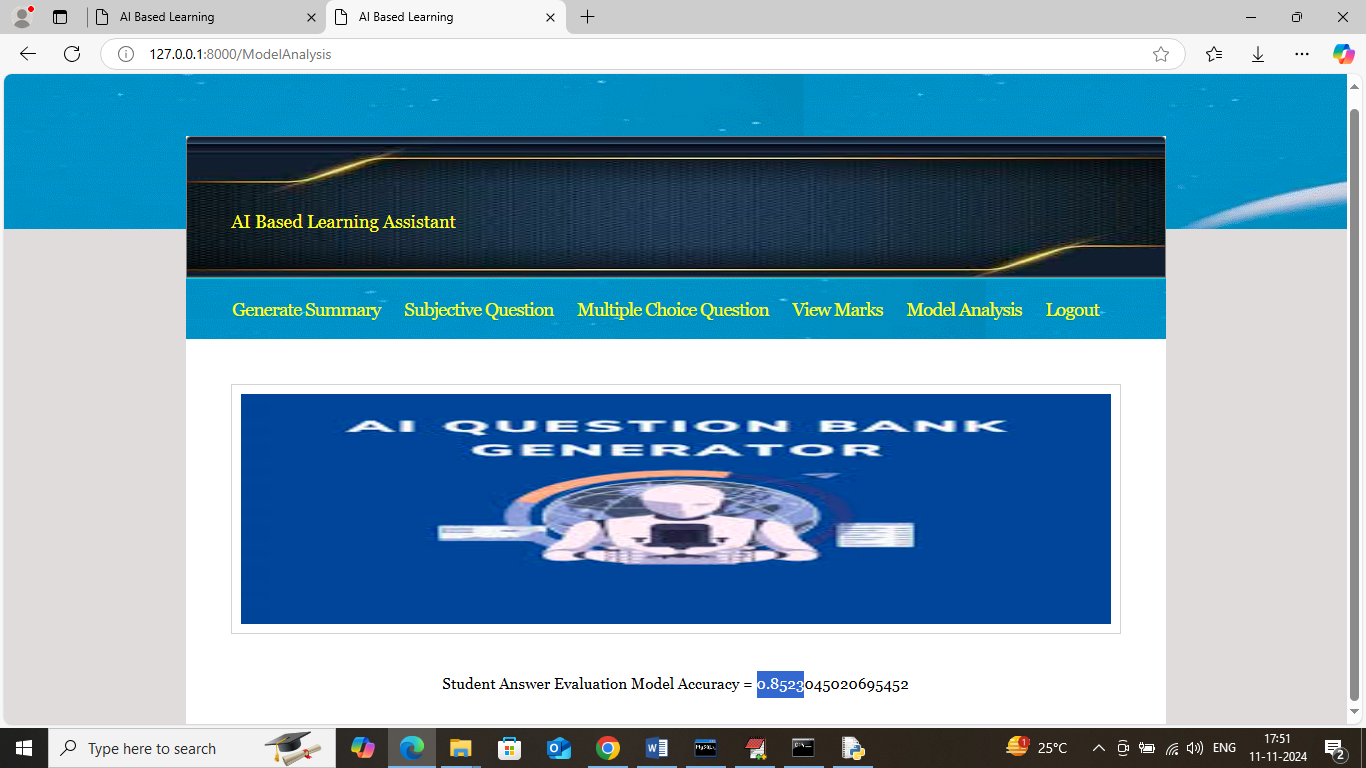
In above screen faculty is login and after login will get below page



In above screen click on ‘View Marks’ link to view all assignments marks and get below page



In above screen faculty can view marks for all students assignment and now click on ‘Model analysis’ link to get below page



In above screen model got 85% accuracy to evaluate all student’s answers.

Similarly by following above screens you can generate automatic questions and answers and students can write exams

**8. CONCLUSION:**

The existence of artificial intelligence may be able to provide knowledge to students, but developing character cannot be done. That is an educator’s job. How to inspire, motivate, make students become good students.” So the role of the teacher in providing motivation, inspiration, and developing character are what AI cannot replace because AI is not given feelings and emotions like humans in general. In the end, if we look at technological developments, we must be able to adapt as technology advances. If we do not adjust, we are an educator (teacher/lecturer) may be replaced by technology.

**9. REFERENCES:**

[1] Abu Ghali, M. J., Abu Ayyad, A., Abu-Naser, S. S., & Abu Laban, M. (2018). An Intelligent Tutoring System for Teaching English Grammar.

[2] Adi, R. (2021). Metodologi Penelitian Sosial dan Hukum. Yayasan Obor Indonesia. Alhabbash, M. I., Mahdi, A. O., & Naser, S. S. A. (2016). An Intelligent Tutoring System for Teaching Grammar English Tenses. European Academic Research, 4(9), 1–15.

[3] Bin, Y., & Mandal, D. (2019). English teaching practice based on artificial intelligence technology. Journal of Intelligent & Fuzzy Systems, 37(3), 3381–3391. <https://doi.org/10.3233/JIFS-179141>

[4] Canbek, N. G., & Mutlu, M. E. (2016). On the track of Artificial Intelligence: Learning with Intelligent Personal Assistants. Journal of Human Sciences, 13(1), 592–601.

[5] Fitria, T. N. (2021a). Grammarly as AI-powered English Writing Assistant: Students’ Alternative for Writing English. Metathesis: Journal of English Language, Literature, and Teaching, 5(1), 65–78. https://doi.org/10.31002/metathesis.v5i1.3519 Fitria,

[6] T. N. (2021b). QuillBot as an online tool: Students’ alternative in paraphrasing and rewriting of English writing. Englisia: Journal of Language, Education, and Humanities, 9(1), 183–196. <https://doi.org/10.22373/ej.v9i1.10233>

[7] Haryanto, H., Rosyidah, U., & Kardianawati, A. (2018). Model Elemen Game Imersif Berbasis Appreciative Learning dan Kecerdasan Buatan Pada Game Pembelajaran. Proceeding SENDI\_U. <https://unisbank.ac.id/ojs/index.php/sendi_u/article/view/5970>

[8] Hwang, Y. S., & Vrongistinos, K. (2012). Using Blackboard and Skype for Mentoring Beginning Teachers. American Journal of Distance Education, 26(3), 172–179.

[9] Jain, S., & Alam, M. A. (2020). Comparative Study of Artificial Intelligence-Based Teaching With Human Interactive Teaching: In E. C. Idemudia (Ed.), Advances in Business Strategy and Competitive Advantage (pp. 68–100). IGI Global. <https://doi.org/10.4018/978-1-7998-3351-2.ch005>

[10] Klamma, R., Lange, P. de, Neumann, A. T., Hensen, B., Kravcik, M., Wang, X., & Kuzilek, J. (2020). Scaling Mentoring Support with Distributed Artificial Intelligence. Intelligent Tutoring Systems, 38–44. https://doi.org/10.1007/978-3-030-49663-0\_6

[11] Mufdalifah, M. (2017).Personalized Learning dan Multimedia Berbasis Komputer Masih Perlukah Guru? JINOTEP (Jurnal Inovasi Dan Teknologi Pembelajaran): Kajian Dan Riset Dalam Teknologi Pembelajaran, 1(1), 50–57. https://doi.org/10.17977/um031v1i12014p050

[12] Sari, R. K. (2021). Penelitian Kepustakaan Dalam Penelitian Pengembangan Pendidikan Bahasa Indonesia. Jurnal Borneo Humaniora, 4(2), 60–69. <https://doi.org/10.35334/borneo_humaniora.v4i2.2249>

[13] Terzopoulos, G., & Satratzemi, M. (2019). Voice Assistants and Artificial Intelligence in Education. Proceedings of the 9th Balkan Conference on Informatics, 1–6. https://doi.org/10.1145/3351556.3351588