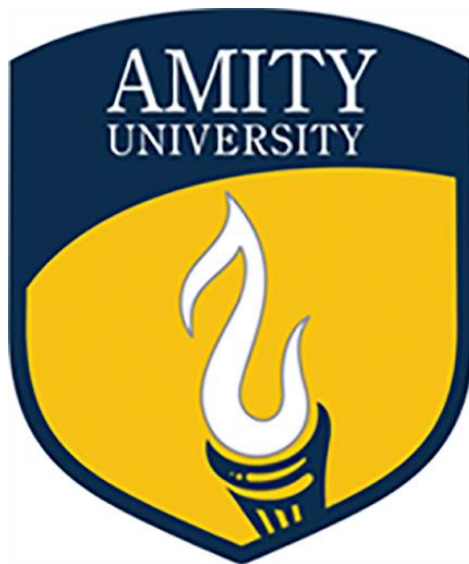


A SOFTWARE PROJECT REPORT
ON
ONLINE EXAMINATION
SYSTEM

FOR THE PARTIAL FULFILMENT OF
MASTERS OF COMPUTER APPLICATION



A PROJECT BY
SUBHRA SAMIR KUNDU AND SUBHAM MAITRA

UNDER
PROF. (DR.) AJANTA DAS

AMITY INSTITUTE OF INFORMATION TECHNOLOGY
AMITY UNIVERSITY, KOLKATA
NOVEMBER 2020

Contents:

- 1) Problem Definition
 - i) Scope of the project
 - ii) Objective of the project
- 2) Summary of the Project
- 3) Requirement Specifications
 - i) Software Specification
 - ii) Hardware Specification
- 4) Feasibility Study
 - i) Technical Feasibility
 - ii) Operational Feasibility
 - iii) Economic Feasibility
- 5) Software Design
 - i) Entity Relationship Diagram
 - ii) Use Case Diagram
 - iii) Logical Structure of the Data
 - iv) Database Tables
- 6) Estimation
- 7) Test Plan and Test Case
 - i) Login Page
 - ii) Add Questions
 - iii) View Questions
 - iv) Manage Questions
- 8) Appendix
- 9) Acknowledgement
- 10) Bibliography

Problem Definition

Scope of the Project:

- Online Examination System is developed or designed for educational institutes like school, colleges, and private institutes to upload, view and manage question papers.
- Design to provide or facilitate Administrator and User.
- Complete and safe information is provided to the user.
- Can be used anywhere anytime as it is a web-based application.

Objective of the Project:

- Online examination can reduce the hectic job of organising the question papers manually and thus manage the question papers without any active overhead
- It will reduce paper work to be an integrated Online Examination System. The question papers can be shown immediately to the all the teachers. It will also be a repository for the old papers.
- Can create various reports and graphs for evaluation purposes almost
- instantly when and where required.

Summary of the Project:

The online examination system developed here can be used to collect question papers unambiguously without any unfair means during this state of pandemic throughout the world due to COVID-19 virus. This can also be used to explain the basic working of a website and how a student can use it to connect to a database and write and fetch data from the same. This system can also be used as a training system by the students for their preparations for the competitive examinations. This is also an example where the students can easily understand the life cycle of a software and its testing. This is developed in python with Django backend and MySQL as its database the default by Django but can easily be converted to PostgreSQL for the real-life systems where terabytes and petabytes of data can be stored with ease. Thus, this is quite scalable and thus future proof. Here we have a secured system of registration and login for the teachers for a proper recording of their actions.

Requirement Specification:

- A need for a web system where the teachers can easily give their question papers easily at instant and thus reduces the burden on organizers and saves a lot of paper.
- In the current scenario of Pandemic when the students are attending the classes from their home and are facing issues to give the exams, this system should address that problem.
- This should be an easy one so that it can be used by the students to understand how websites work and how they can connect the same with the databases and how to fetch that data and change the data should be clearly shown.
- This should be a system which can easily be scaled in the future.
- This should be accessible from anywhere and anytime and any device.
- A proper way to record the data should be there.
- It should be user friendly.
- It can be used as a repository for all the old papers.

Software Specifications

The Software used for building the project are (all described in the Feasibility Analysis):

- a) **Python**
- b) **Streamlit with Flask and Django Backend**
- c) **MySQL**
- d) **HTML**
- e) **CSS**

Hardware Specifications

The Hardware backend required for building the project are (all described in the Feasibility Analysis):

- a) **CPU:** Any Single Core CPU or higher
- b) **Ram:** 1 GB (Minimum for running the Flask and Django Backend for Streamlit) or higher
- c) **Hard Drive Space:** 200 MB or higher
- d) **Operating System:** Preferably Linux (especially Cent OS or Debian). It can run on any platform but in others it will have some overheads.

Feasibility Study

1. Technical feasibility:

For designing purpose as well as for the technical support for this project, the following technologies are going to be used-

- a. **Python 3.7:** As we all know that, Python is an interpreted, high-level and general-purpose programming language & hence Python is going to be the backbone of this project. The version of Python which is going to be used is Python 3.7. Compared with the previous version of Python, this 3.7 release has some updated features which will make this project more feasible in terms of technical feasibility. This huge technical support can be provided by this 3.7 release because of having some more dynamic features like, "Postponed Evaluation of Annotations", "Legacy C locale Coercion", "Forced UTF-8 Runtime Mode", "Built-in breakpoint ()", "Customization of Access to Module Attributes", "New Time Functions with Nanosecond Resolution" etc.
- b. **Django:** Django is a Python-based free and open-source web framework that follows the **model-template-views (MTV)** architectural pattern. It is maintained by the **Django Software Foundation (DSF)**. Despite having its own nomenclature, such as naming the callable objects generating the HTTP responses "**views**", the core Django framework can be seen as an MVC architecture. It consists of an **object-relational mapper (ORM)** that mediates between data models (defined as Python classes) and a relational database ("**Model**"), a system for processing HTTP requests with a web templating system ("**View**"), and a regular-expression-based URL dispatcher ("**Controller**").

Also included in the core framework are:

- a lightweight and standalone web server for development and testing
- a form serialization and validation system that can translate
- between HTML forms and values suitable for storage in the database
- a template system that utilizes the concept of inheritance borrowed
- from object-oriented programming
- a caching framework that can use any of several cache methods
- support for middleware classes that can intervene at various stages
- of request processing and carry out custom functions
- an internal dispatcher system that allows components of an
- application to communicate events to each other via pre-defined signals
- an internationalization system, including translations of Django's
- own components into a variety of languages
- a serialization system that can produce and read XML and/or JSON
- representations of Django model instances
- a system for extending the capabilities of the template engine
- an interface to Python's built-in unit test framework
- Django includes dozens of extras developers can use to handle common Web development tasks. Django takes care of user authentication, content administration, site maps, RSS feeds, and many more tasks — right out of the box.

Django takes security seriously and helps developers avoid many common security mistakes, such as SQL injection, cross-site scripting, cross-site request forgery and clickjacking. Its user authentication system provides a secure way to manage user accounts and passwords. Django can run in conjunction with Apache, Nginx. Django officially supports five database backends: **PostgreSQL, MySQL, MariaDB, SQLite, and Oracle**. Microsoft SQL Server can be used with django-mssql on Microsoft operating systems, while similarly external backends exist for IBM Db2, SQL Anywhere and Firebird.

- c. **Streamlit:** Streamlit lets us turn data scripts into sharable web apps in minutes, not weeks. It's all Python, open-source, and free! And once you've created an app you can use our free sharing platform to deploy, manage, and share your app with the world. Streamlit's simple and focused API lets you build incredibly rich and powerful tools. With less than 500 lines of code we can develop and deploy a web application. It has a simple to use API with Python backend which can cater to the needs of those individuals to whom the data is main rather than the GUI. It allows to integrate all the algorithms of machine learning and deep learning with the powerful python backend which it provides. It takes the properties of famous python web API's such as Flask and Django and mainly uses Nginx server to run upon. It is a very powerful tool for data visualization and presentation. This reduces the overhead knowing the web making tools like HTML and CSS. Just the knowledge of Markdown is more than enough to make the website look presentable to a business. It also allows for easy posting of the website over GITHUB and its own server if it is somewhat in the size.
- d. **MySQL:** A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language programmer uses to create, modify and extract data from the relational database, as well as control user access to the database. Hence, being an open-source relational database management system (RDBMS),MySQL works with an operating system to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.
- e. **HTML & CSS:** Hypertext Mark-up Language (HTML) is the standard mark-up language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items.
- f. **Cascading Style Sheets (CSS)** is a style sheet language used for describing the presentation of a document written in a mark-up language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.
CSS is designed to enable the separation of presentation and content, including layout, colours, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .CSS file which reduces complexity and repetition in the structural content as well as enabling the .CSS file to be cached to improve the page load speed between the pages that share the file and its formatting. Separation of formatting and content also makes it possible to present the same mark-up page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device. Hence, we can easily conclude that the project which we are building with the help of these technologies will undoubtedly become feasible technically.

- g. **Hardware Requirements:** The software with all its merits runs smoothly with the support of a lot of hardware at its backend, here we have tried to list out the minimum requirement of hardware which is required to run this project. They are listed below:
- i) **CPU:** Any Single Core CPU or higher
 - ii) **Ram:** 1 GB (Minimum for running the Flask and Django Backend for Streamlit) or higher
 - iii) **Hard Drive Space:** 200 MB or higher
 - iv) **Operating System:** Preferably Linux (especially Cent OS or Debian). It can run on any platform but in others it will have some overheads.

2. Operational Feasibility:

Operational feasibility of any project can be measured according to the measure of scalability & adaptability. As we have decided to use the technology for the build-up process of the project, we can state that this work will be scalable as well as adaptable because of having the support provided by the technology which we are using.

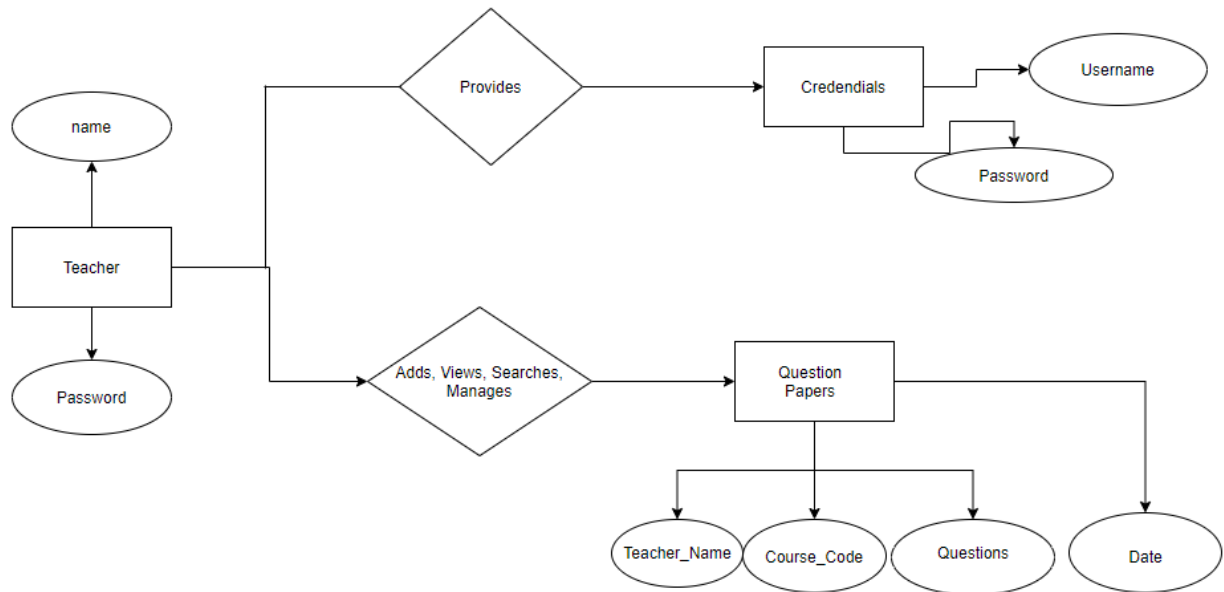
3. Economic Feasibility:

According to the building up the project, we can easily divide the total budget with respect to the modules of the project i.e. server, database & implementation related cost. We have the approximated cost for the respective modules as follows:

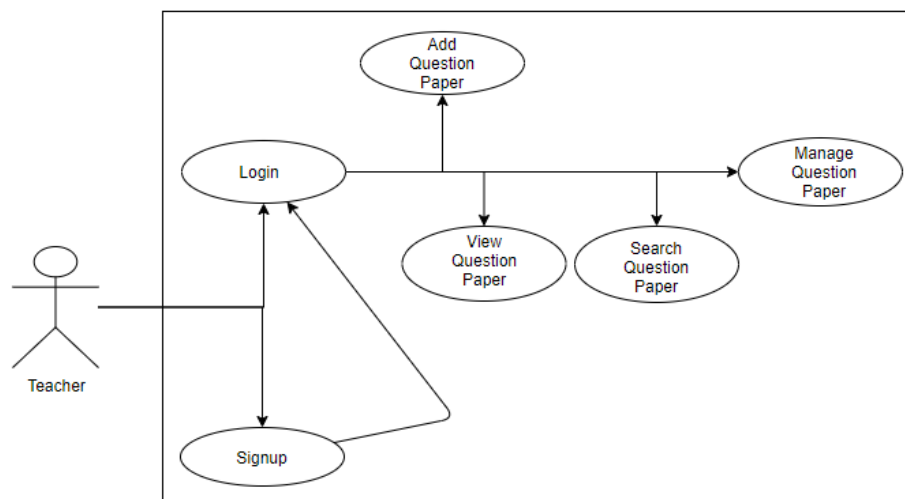
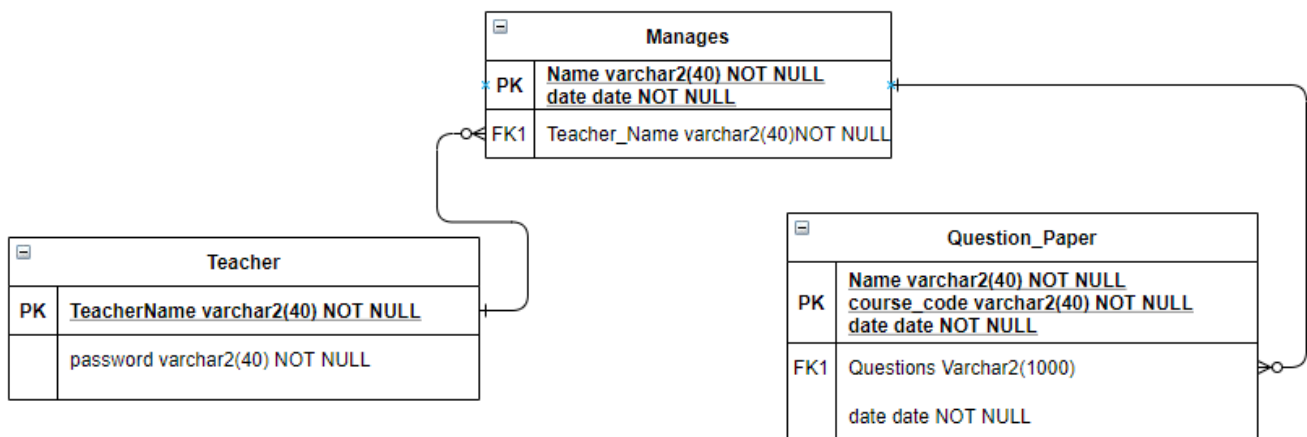
- **Server:** approximately Rs.15,000
- **Database:** approximately Rs.10,000
- **Implementation:** approximately Rs.15000

Hence, we can state that we have the feasible budget for the project with respect to the economic feasibility.

Designs



Entity Relationship Diagram



Use Case Diagram

Logical Structure of the Data

The two sections below show the different types of information used by various functions and the overall data model, respectively.

Types of Information Used

The types of information used by various functions of the website:

| Function | Types of Information Used |
|------------------------------|--|
| Account Registration/Sign up | User information (name, password) |
| Account login | User information (Teacher_name and password) |
| Search | Question_Paper (Teacher_name, Subject_Code) |
| Delete | Question_Paper (Teacher_name, Subject_Code) |

The following diagram describes the attributes and the types of data that shall be used in the operations:

| User | |
|----------|--|
| Username | Unique, at least 5 letters + digits, at most 40 characters |
| Password | At least 8 letters + digits. Up to 40 |

| Question Paper | |
|----------------|-------------|
| Teacher Name | String |
| Subject Code | String |
| Questions | String |
| Date | Date String |

Database Tables:

| UserTable | | |
|-----------|--------------|-------------------------------|
| Attribute | Data Type | Constraint |
| name | Varchar2(40) | Primary Key, NOT NULL DEFAULT |
| password | Varchar2(40) | NOT NULL DEFAULT |

| Question Paper Table | | |
|----------------------|----------------|-----------------------|
| Attribute | Data Type | Constraint |
| Teacher Name | Varchar2(40) | PRIMARY KEY, NOT NULL |
| Subject Code | Varchar2(20) | PRIMARY KEY, NOT NULL |
| Questions | Varchar2(1000) | NOT NULL |
| Date | Date | NOT NULL |

Estimation

For the estimation for this project, we are using the famous COCOMO Model. Because, COCOMO predicts the effort & schedule for a software product development based on inputs relating to the size of the software & number of cost drivers that affect productivity.

COCOMO has three different models that reflect the "**complexity**":

- the Basic Model (Organic Mode)**
- the Intermediate Model (Semi-detached Mode)**
- the Detailed Model (Embedded Mode)**

We have the basic COCOMO model formula as follows:

$$E = a_b (KLOC)^{b_b}$$

$$TDEV = c_b (E)^{d_b}$$

- E** = effort applied in person-months
- TDEV** = development time in chronological months
- KLOC** = estimated number of delivered lines of code for the project

The value of a_b, b_b, c_b, d_b are as the following:

| Values → | a_b | b_b | c_b | d_b |
|---------------|-------|-------|-------|-------|
| Mode ↓ | | | | |
| Organic | 2.4 | 1.05 | 2.5 | 0.38 |
| Semi-detached | 3.0 | 1.12 | 2.5 | 0.35 |
| Embedded | 3.6 | 1.20 | 2.5 | 0.32 |

Estimate Calculation:

We have the total project size as 0.3 KLOC

So, we can have the estimation as the following

Organic Mode:

$$\text{Effort: } 2.4 * (KLOC)^{1.05} = 2.4 * (0.3)^{1.05} = 0.67794 \text{ PM}$$

$$TDEV = 2.5 * (E)^{0.38} = 2.5 * (0.67794)^{0.38} = 2.1567 \text{ P}$$

Considering the cost ppm = Rs.10000 we are calculating the cost as:

$$\text{Cost} = \text{effort} * 10,000 = (0.67794) * 10,000 = \text{Rs.6779.40}$$

The effort since calculated in person month gives us the estimation as to whether a certain number of people will work for 0.67794 Months will complete the project. Thus, it will be the time. Cost is effort times per person month so it is 10000 per person month, giving us a total cost of Rs Rs.6779.40

Test Cases

Login Page

| SI No. | Scenario | Input Step | Expected Result | Actual Output | Tested By, Date |
|--------|--|---|--|---|-------------------|
| 1. | Username should be of less than 150 characters | Username of less than equal to 150 characters were given | Username to the application is less than 150 characters | The application accepts a username of less than equal to 150 characters | Subhra, 2/10/2020 |
| 2. | Username should be of less than 150 characters | Username of more than 150 characters were given | Username to the application is less than 150 characters should be accepted | The application accepts a username of less than equal to 150 characters | Subham, 4/10/2020 |
| 3. | Password should have an uppercase character, a lowercase character, a special character and a number and should be of at least 8 characters and at max 50 characters | A password with an uppercase character, a lowercase character, a number and a special character and of length less than 50 characters and more than 8 characters were given | The password was accepted as it was meeting all the conditions and it had the proper length | The password was accepted | Subhra, 2/10/2020 |
| 4. | Password should have an uppercase character, a lowercase character, a special character and a number and should be of at least 8 characters and at max 50 characters | A password not following some to none of the conditions was given | The proper password meeting all the criteria should be given and until it is done the input would not be accepted. | Until the password meeting the proper criterions, the password was not accepted | Subham, 4/10/2020 |
| 5. | When a new user who is not registered tries to log in | A new user who was not registered was trying to log in | The user should not be identified to log in and should be redirected to registration page | The data of the user was said to be not found and sent to registration page for registering | Subham, 4/10/2020 |

The Add Question Paper Page

*This page requires the login step aforementioned

| Sl No. | Scenario | Input Step | Expected Result | Actual Output | Tested By, Date |
|--------|---|-----------------------------|--|---|-------------------|
| 1. | A registered users username is needed to upload the questions | A registered name was taken | The user should be allowed to upload the questions | The user is allowed to upload the questions | Subhra, 2/10/2020 |
| 2. | A registered users username is needed to upload the questions | A registered name was taken | The user should be allowed to upload the questions | The user is allowed to upload the questions | Subham, 4/10/2020 |
| 3. | A registered users username is needed to upload the questions | A registered name was taken | The user should be allowed to upload the questions | The user is allowed to upload the questions | Subham, 6/10/2020 |

The View Question Paper Page

*This page requires the login step aforementioned

| SL No | SCENARIO | INPUT | EXPECTED OUTPUT | ACTUAL OUTPUT | Tested BY |
|-------|---|---------------------------------|---|---|-------------------|
| 1. | A registered user username is needed to VIEW the questions. | A registered name was taken | The user should be allowed to VIEW the questions | The user is allowed to VIEW the questions | Subham, 2/10/2020 |
| 2. | A registered user username is needed to VIEW the questions. | A registered name was not taken | The user should be not be allowed to VIEW the questions | The user is not allowed to VIEW the questions | Subhra, 4/10/2020 |
| 3. | A subject name is needed to VIEW the questions. | A subject name was taken | The user should be allowed to VIEW the questions | The user is allowed to VIEW the questions | Subhra, 6/10/2020 |

The Search Question Paper Page

*This page requires the login step aforementioned

| SL No | SCENARIO | INPUT | EXPECTED OUTPUT | ACTUAL OUTPUT | Tested BY |
|-------|---|---------------------------------|--|---|-------------------|
| 1. | A registered user username is needed to SEARCH the questions. | A registered name was taken | The user should be allowed to SEARCH the questions w.r.t TEACHER & SUBJECT | The user is allowed to SEARCH the questions | Subham, 2/10/2020 |
| 2. | A registered user username is needed to SEARCH the questions. | A registered name was not taken | The user should not be allowed to SEARCH the questions w.r.t TEACHER & SUBJECT | The user is not allowed to SEARCH the questions | Subhra, 4/10/2020 |
| 3. | A subject name is needed to SEARCH the questions. | A subject name was taken | The user should be allowed to SEARCH the questions w.r.t TEACHER & SUBJECT | The user is allowed to SEARCH the questions | Subham, 6/10/2020 |

The Delete Question Paper Page

*This page requires the login step aforementioned

| SI No. | Scenario | Input Step | Expected Result | Actual Output | Tested By, Date |
|--------|---|---------------------------------|--|---|-------------------|
| 1. | A registered users username is needed to DELETE the questions | A registered name was taken | The user should be allowed to DELETE the questions | The user is allowed to DELETE the questions | Subhra, 2/10/2020 |
| 2. | A registered users username is needed to DELETE the questions | A registered not name was taken | The user should not be allowed to DELETE the questions | The user is allowed not to DELETE the questions | Subham, 4/10/2020 |
| 3. | A course name is needed to DELETE the questions | A course name was taken | The user should be allowed to DELETE the questions | The user is allowed to DELETE the questions | Subhra, 6/10/2020 |

Appendix

×

Menu

Login

User Name

smaitra

Password

....

☒ Login

QP Menu

Add Question Papers

Teacher Name:smaitra

Enter Subject Code

CSIT2022

8/50

Paste your question set here

Q1
Q2
Q3
Q4
Q5

Press Ctrl+Enter to apply

Enter The Date

2020/11/23

Add

Add Question Paper Module and Login Module on the left sidebar

×

Menu

Login

User Name

smaitra

Password

....

☒ Login

QP Menu

View Question Papers

View Question Papers

smaitra

Question Papers in Database

Teacher:smaitra
Subject:CSIT2020

2020-11-23

Q1 Q2 Q3 Q4

Teacher:smaitra
Subject:CSIT2019

2020-11-23

Q1 Q2 Q3 Q4 Q5 Q6

View Question Paper Module and Login Module on the left sidebar

×

Menu

Login

User Name

smaitra

Password

....

☒ Login

QP Menu

Manage Question Papers

Manage Question Papers Database

| | Teacher | Subject | Questions | Post Date |
|---|---------|----------|---------------------------------|------------|
| 0 | qw | qwe | fgvdfgbdicb cfbhcvbvc b dgbv... | 2020-11-03 |
| 1 | sm | opit | skdjgfsdkgsdkj ksjdfhgsdkf... | 2020-11-11 |
| 2 | sm | qwe | sdidsfsdf dfbgfhg xfgfdgdf... | 2020-11-11 |
| 3 | qw | tera | Hsiaskfsdhfkdsj ahnfskdjfh... | 2020-11-11 |
| 4 | qw | pot | zdcfxz didfs sdfsd fsd zcvx... | 2020-11-11 |
| 5 | qw | pot2 | hikusddjksfgh asdfhjvakjgh... | 2020-11-11 |
| 6 | smaitra | CSIT2020 | Q1 Q2 Q3 Q4 | 2020-11-23 |
| 7 | smaitra | CSIT2022 | Q1 Q2 Q3 Q4 Q5 | 2020-11-23 |

Teacher Name:smaitra

Unique Subjects

CSIT2020

Delete

Manage Question Paper Module and Login Module on the left sidebar

Menu

Login

User Name

smaitra

Password

••••

☒ Login

QP Menu

Search Question Papers

It is developed by the abovesigned.
Contact [here](#)

Logged In as smaitra. Please Select the sidepane and options

Search Question Papers

Enter Search Term

Field to Search By

☒ teacher

☐ subject

Search

Made with [Streamlit](#)

Search Question Paper Module and Login Module on the left sidebar

Menu

Login

User Name

smaitra

Password

••••

☒ Login

QP Menu

Search Question Papers

Search

Teacher:smaitra
Subject:CSIT2020

2020-11-23

Q1 Q2 Q3 Q4

Teacher:smaitra
Subject:CSIT2022

2020-11-23

Q1 Q2 Q3 Q4 Q5

Search Question Paper Module and Login Module on the left sidebar

Acknowledgement

The success and the final outcome of this project required a lot of guidance and assistance from many people and we are extremely privileged to have got this all along the completion of our project. All that we've done is only due to such supervision and assistance and we would not forget to thank them.

We respect and thank **Dr. Ajanta Das, Co-ordinator, Research Activity Sub-committee Amity Institute of Information Technology Amity University, Kolkata** for providing a golden opportunity to do the project under her guidance & we are also extremely thankful to our honourable **HOI sir, Prof. Dhrubasish Sarkar Assistant Director & Head of Institution Amity Institute of Information Technology** for providing such a nice support and guidance, although he had a busy schedule managing the corporate affairs.

We are very much thankful and fortunate enough to get constant encouragement, support and guidance from all teaching staff of Amity Institute of Information Technology who helped me in successfully completing the project work.

This project has given us a lot of insights about how the software project development and its life cycle occurs in the real life. Also, this project has taught us how to do proper testing and estimation. We shall always be thankful to Ajanta Ma'am for giving us such an opportunity.

Bibliography

This project has given us a lot of insights about how the software project development and its life cycle occurs in the real life. Also, this project has taught us how to do proper testing and estimation. We shall always be thankful to Ajanta Ma'am for giving us such an opportunity.

For the completion of the project there were some “fountainheads” in the words of Ayn Rand. The first would be the software suite and the support by the python community, the Streamlit community and the Django community which helped me solving all the problems which we had to encounter during the time of development.

There are also few people without whose support it would have been impossible to do it. It would be our family and the friends and classmates. Lastly, our companion google was a true guidepost in this.

The following are the links in this regard:

- <https://www.python.org/>
- <https://www.streamlit.io/>
- <https://www.djangoproject.com/>
- <https://www.google.com/>

Find the code and instructions to run here at:

<https://github.com/S2KtheGeek/Online-Examination-System>