INDIRA GANDHI NATIONAL OPEN UNIVERSITY

IPS

(IGNOU Prashnottari Sewa)
Version 1.1

Developed by

ACIIL, IGNOU

Under Guidance
of

Prof. K. R. Srivathsan
Pro-Vice Chancellor, IGNOU

Indira Gandhi National Open University
Maidan Garhi
New Delhi - 110068

Acknowledgement

As anyone who has written a project work, or research work, it is quite impossible to acknowledge by name every individual who has played some part in this work. I feel it is difficult to express in words my profound sense of gratitude to most respected persons who helped us to make this work possible.

On the behalf of Development Team, ACIIL, IGNOU I acknowledge our gratitude to respected Mr. Rejith R.; who coordinated and guided us at each and every step in project. Our special thanks and gratitude to Hon'ble Prof. K. R. Srivathsan, Pro-Vice Chancellor, IGNOU; who brought the idea of IPS and gave us a great opportunity to work on it. He has not only guided us, showed us a bigger picture of the world and also supported as a great teacher. We thank to Prof. K. Subramanian, Director, ACIIL; who provided us all the administrative and financial supports. I thank to my development team Ms. Ridhi Khera, Neeraj Singh and Seema Sahi because of them IPS could be developed. We thank to Mr. Sukant Kole and his team Mr. Ganesh Tamang and Mr. Sanjay Kumar; who supported and resolved our all the network level problems and made successful and secured deployment of IPS. We thank to Ms. Divyaraj, Mr. Vivek Ranjan and ETITS team who suggested for the improvements and linked IPS with the ODTEL wiki to make IPS live and visible.

Last but not least; we are thankful to entire IGNOU staff for giving us enough support to study the existing system. Finally of course great debts are owed to Open Source Community whose wholehearted support has given us the inspiration and dedication to complete this work.

Anand Shankar
Consultant, ACIIL, IGNOU

CONTENTS AT A GLANCE

1. An introduction

The developed system **IPS** stands for **"IGNOU Prashnottari Sewa"** is a live project and developed in the campus of IGNOU headquarter at ACIIL, IGNOU, Maidan Garhi, New Delhi instructed and guided by **Pro Vice Chancellor Prof. K. R. Srivathsan and** coordinated by **Mr. Rejith R.**. There are four members in our team **Anand Shankar, Ridhi Khera, Seema Sahi** and **Neeraj Kumar Singh**. The **IPS** project is a **Query Management System**, conceptualized, designed and is being implemented for the **Indira Gandhi National Open University Govt. of India** for the Student, IGNOU staff and others also who is not belonging to IGNOU. IPS is a part of ***Vedyadhara***. It is a pilot implementation of application of Information Technology through relevant information systems and networking to aggregate, share and disseminates information of importance and interest to the students, lecturers, workers and officials in ways that enhance the total Education development and student's welfare in the country. The main aim of the IPS project is to establish a student centric, integrated, distributed information system for education, which would focus on catering to the various requirements of the students across the India. The core deliverable of the project is the integrated, multi-component, multi-model and geographically distributed Educational Information System that is accessible anywhere anytime by all concerned. With the On Line IGNOU Prashnottari Sewa; web software we can drastically accelerate student Query response process for a particular university. We can now give response student registered for various courses existing in a university in a very short period of time. It will not only help to solve administrative queries of student, academic query can be solved also through IPS which will be routed directly to concern Programme or Course Coordinator.

1.1 Background

The student grievances process was being done manually with lots of paper work and postal delay which is the most time consuming, unreliable and cumbersome activities. As a result of this precious productive time is lost and the daily routine work remains pending and gets piling up and after some time a situation of no come back arises.

Maintaining all the records pertaining to the student query for a particular course in a university is quite tedious task which involved a lot of paper work as administrator had to maintain records for each and every student regarding their queries related to courses offered ,educational qualification ,personal details of students as well as acknowledgement through mail etc .every now and then new students get added which leads to updating the records , entering all these details manually which is quite tough or could be inappropriate also because of human errors and it is be very time consuming .The IGNOU Prashnottari Sewa project is there to help to ease all the above work .

Existing System is mail based therefore getting an email id of a particular expert and ask him to question is not a good system. Existing system has following draw backs.

1. No academic support:-There is no academic support in existing system. So user cannot post general queries .There are no subject experts to answer your subject related query.

2. No *FAQ* updating facility:-There is no FAQ updating facility in existing system so users unnecessarily post their query which is of general nature even though their queries can be added to FAQ thus creating huge traffic on server.

3. No *alert* system: - In existing system there are no Alert systems. That can help experts to know for how long this query is pending.

4. No report generation facility: - In existing system there is no provision to view report daily, weekly and monthly.

1.2. Objective

Our objective is to handle all types of queries related to IGNOU either *Academic* or *Administrative*; IGNOU is responsible to give the satisfactory answer to the student. The developed system IPS has facility to answer any type of Query and can cater to any type of user. Proposed system has some advantage over Exiting system.

1. Department officials can post the answer: - Department Administrator as well as Department experts can answer the query.

2. Provision for external experts to answer: - If an external expert suggestion is required for a particular query then that query can be sent to external experts through our IPS system.

3. Query gets posted to concerned department official directly: - To avoid bottle neck there is provision to send query in individual departments or in case of heavy traffic on same server we can transfer the query to external experts of concerned department.

4. Academic support: - General user can send any query that can be taken care of.

5. Alert system incase the query is not answered in a particular time: - In general users will receive his/her answer in a speculated time i.e. within 24 hours. But if an external expert is to be consulted.

6. Report generation facility: - There will be a facility to generate reports daily, weekly or monthly so that the activities being carried out in QMS can be monitored.

7. FAQ facility: - There will be a frequently asked questions (FAQ) facility, so that the common queries which are getting posted often will be marked as FAQ and there will be no need for the students to post the query again. He will have the accessibility to FAQ and get the answer for that.

8. Useful for common people: - This facility allows every user, it need not be a registered student of IGNOU only, to post the query and get the answer.

9. SMS facility: - In the second phase of development of this QMS, we are planning to integrate an SMS facility by which the student can post the query through SMS and get back the answer also through SMS.

10. *P-P* chat facility: - This facility is also planned to be implemented on the next phase of development. This allows the student to have a Person to Person chat.

11. Acknowledgement through Email: - Sending email to the student regarding their queries Token ID as well as reply automatically.

12. Track Query Status: - Interface for students or general users to get their query's status.

2. A Complete Project Description

IGNOU PRASHNOTTARI SEWA

IGNOU Prashnottari Sewa (IPS) is the vital and integral part of IGNOU portal. In fact it is one such service that will cater to the needs of the students by providing solutions to their problems. This whole system is designed for the general users like students and those interested in educational practices. Through the portal the user can post queries and the experts in the field will answer them. IPS with its versatility and credibility has now become a buzzword among the student. The new IPS on the anvil is an upgraded version of the existing IPS. New IPS is much more compatible and has more features to deliver the queries in an efficient and effective way. The IPS has an eminent team of experts from diverse fields viz educational, research and development backed by dedicated support of system administrators for the smooth functioning of the system.

The main functions rendered by this module are registrations by the different users/clients, posting of queries, answering the queries by internal/external experts, searching queries (*archives*) and generating category wise reports for quick viewing from the archives. The key player involved in this module includes:

- General user
    - Internal Expert
    - External expert
    - Query Admin
- *Super Admin*

***General* User**

The general user is the key player for which the query management system is developed. General user can be students, department officials, lecturers,

entrepreneurs and research scientists who can post their query and get the answers from the experts. The queries will be entertained general users.

Query Admin

Query admin is the main pivot around which the whole IGNOU Prashnottari Sewa revolves. The main duties vested with the query admin is filtering and organizing the queries according to its category and forwarding it to the internal experts.

Internal Expert

Internal expert processes the queries forwarded by the system admin and takes necessary steps to answer the queries. If the query doesn't fall under his purview of specialization or if the internal expert needs further guidance he has the option to forward it to external experts. The internal expert also has the privilege to edit (in case of any mistakes) the queries and organize them into category and sub-category so that it becomes easy for the users to search the archives.

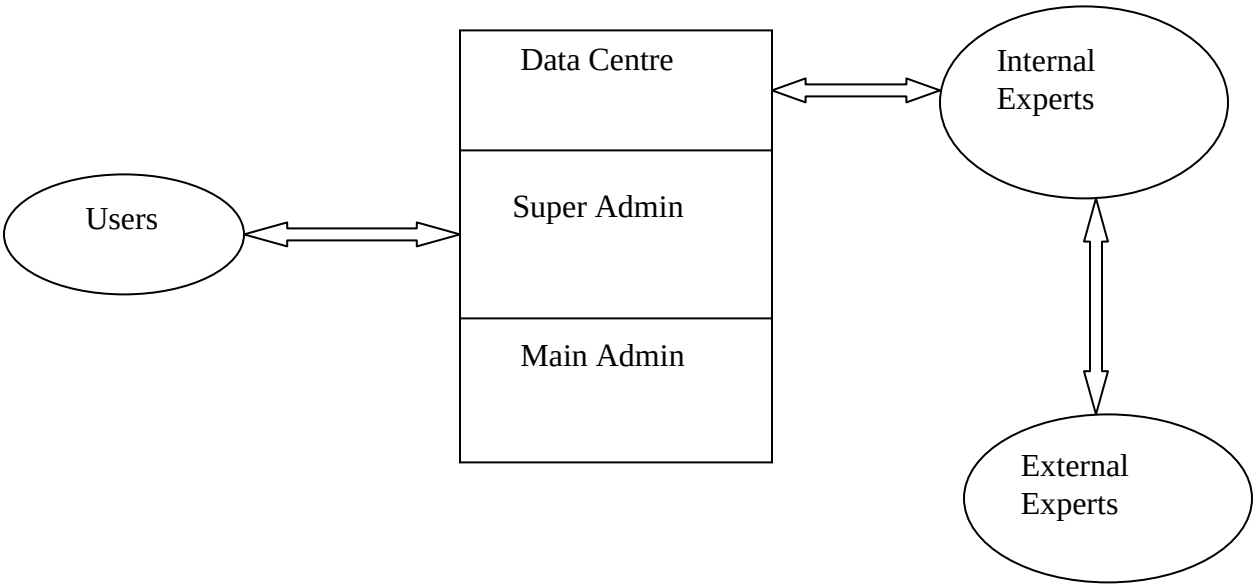
External Expert

External experts are the subject experts in various fields of specialization. The queries posted (forwarded) by the internal expert will be answered by them. The system admin also maintains a subject-wise list of all the concerned external experts who are connected to IGNOU team both directly and indirectly.

Super Administrator

The super admin has the overall responsibility of the entire query IGNOU Prashanottari Sewa. Administrator monitors and controls the system flow process on an up-to-date basis.

The schematic flow diagram of the IGNOU Prashanottari Sewa



User Registration

The queries will be entertained from any user. In the registration process the user is requested to fill up certain fields (there are mandatory and optional fields). Fields like**, Name, Login, Password, Address, Date of Birth, Contact number, and Email Id** are mandatory and common to all users. The General Users have been categorized into five classes and there are mild variations in the fields required to be filled up by each class. First category is General users from outside **IGNOU (It will be automatically captured).** Second category is **Indian Students** who has additional mandatory fields like **Enrolment number**, **Course Code, (applicable only if the user is an IGNOU student)**. The third category is **International Student (applicable only if the user is an IGNOU student) .**Fourth category is IGNOU Operational where internal user of all departments of IGNOU can interact. Users are a part of **IGNOU (IGNOU Administration as super admin,**

Department admin, backend support and lecturers, department Expert) and they have additional mandatory fields like **Employee id, Specialization and Department**. Fifth category is Regional Services to operationalise the Regional Centers, Study Centers, and Student Support Services of the university across the length and breadth of the country. On next level there is also provision to give SMS alerts and Emails to the users when a query is being answered for which the users are required to give their mobile number and Email IDs. This privilege is only for users residing in India .The user can also avail the opportunity of getting **Student** news-letters for which a separate field is set up.

Query Posting

Query posting which is one of the major components of IGNOU Prashnottari Sewa. It mainly focuses on managing the flow of queries between the users/clients and the internal and external experts, Department. The questions will be entertained from the registered users.

When the user posts a query, this will be intimated to the System Admin. The posted queries will also have the provision to hold two attachments of a particular size (max 10MB).

When the query posted by the user Confirmation page will be displayed to that user with a unique token id and the response will be given afterwards by identifying the *token id* posted to that user.

Internal Expert's Dash board

Internal expert's dashboard is one of the vital components in query management system as it includes a number of fields for answering the queries, viewing the already answered queries, deleting the queries and editing the query which are forwarded to the

external experts and pending queries **(all the privileges assigned by the Administrator to the desired or specific user)**.

Posted Query from Registered users and **User details** which has a hyperlink that enables the internal expert to view the details of the user. If the user has any attachments that will also be highlighted here with a hyperlink. As depicted the internal expert has four options to process the query.

As illustrated the internal expert can edit the answer and send it to the user directly. The internal expert also organizes the answer into main and sub- categories. For easy navigation while searching, the internal expert also has the privilege to impart **Keywords** while categorizing.

Internal Experts Dash board – New Queries (Forward)

Internal Expert can forward the question to **External Expert** (can select more than one) if the internal expert requires further guidance. It contains fields for external expert's name, specialization and attached documents (if any).

Internal Experts Dash board – New Queries (Edit &Forward)

the process if the internal expert gets multiple queries from a single user. The internal expert can edit the query and forward it to the respective external expert with attachments (if any). If a user includes multiple queries in a single query, the internal expert can edit it and forward the relevant question with attachment, external expert's name, his category and specialization to the respective external expert.

Internal Experts Dash board – New Queries (Delete)

Illustrates internal expert's privilege to delete the queries if found irrelevant.

Internal Experts Dash board – Forwarded Queries (Answer yet to be received)

Illustrates the Forwarded queries which are yet to be answered in the internal experts dash board. It has fields such as Posted Query from Registered users, posted date, attachments (if any), Name and Specialization of the external expert.

Internal Experts Dash board – Answered Queries

Illustrates the Answered queries in the internal expert's dashboard. It has fields such as Posted query from the registered users, User Details with a hyperlink to view the user details, date of posting, name of the expert (internal) who answered the question and posted date (by the expert).

Internal Experts Dash board – Forwarded Queries (Answer Received)

Illustrates the Forwarded queries which are answered by the external expert.
It has the same fields as that of the Answered queries by the internal expert.

Internal Experts Dash board – Deleted Queries

Illustrates the privilege of the internal expert to view the deleted queries.

Internal Experts Dash board – Answered Queries (multiple answers from external experts)

Illustrates the process if a query is answered by more than one external expert. Here the internal expert has the liberty to edit and select the answers of his choice. There is also provision to attach documents (if any).

External Expert's Dashboard

Illustrates the dashboard maintained by the **External expert.** It has fields for **New queries (forwarded by the internal expert)** and **Answered queries** by the external expert.

External Expert's Dashboard - New Queries

Illustrates **New Queries** field in external expert's dashboard. It has fields such as posted **Query, User details** with a hyperlink**, date of posting (user), name of the internal expert** who forwarded the query**, date of posting (internal expert).** The external expert can answer the query only through the **Answer** button provided at the bottom of the page, which displays the required page for answering the queries.

External Expert's Dashboard - New Queries(Answer)

Illustrates **Answer** field in detail. It has fields for answering the query as well as adding any attachments.

External Expert's Dashboard – Answered Queries

Illustrates the **Answered Queries** (external expert) in the external expert's dashboard. It has fields such as **posted query, date of posting (user), attachments(user), name of internal expert who forwarded the query, date of posting(internal expert) and the date of posting the answer(external expert) and attachments(provided by the external expert).**

User's Dashboard

Illustrates the dashboard for the user. It has fields such as Answered queries and Queries yet to be answered. The users can post their queries using the Post a question hyperlink, which will display the Query posting page . It has the provision of viewing the whole query with user details.

User's Dashboard- Answered queries

Illustrates the **Answered queries** field in the user's dashboard. It has fields such as **posted query, date of posting (user), answers, date of posting the answers (internal expert)**. The user also has the provision to download the corresponding **attachments** (if any).

User's Dashboard- Queries yet to be answered

Illustrates the **Queries yet to be answered** in the user's dashboard.

Search Answered Queries

The users can search the answered queries using a particular **Keyword or Category**. While clicking any of these links, a corresponding page having the **keyword Search** page or **Category Search** page will be displayed.

3. System Analysis

The software industry considers software development as a process. Software engineering is field, which combines process, methods and tools for the development of software. The concept of process is the main step in software engineering approach. Thus a software process is a set of activities. The various steps (called phases) which are adopted in the development of this process are collectively termed as Software Development Life Cycle (SDLC). System Analysis is the first phase of Software Development Life Cycle. The various activities are involved in the phase. How, IPS is developed according to the Software Development Life Cycle, are described below:

3.1. Identification of Need

Indira Gandhi National Open University is the largest university of the world. There are approx 3 millions of student around the world, enrolled in Indira Gandhi National Open University. For solving student's grievances, IGNOU has a centre known as **SSC** (Student Support Centre). Most of the student's general queries are solved by **SSC**. Some administrative queries are solved by SED (Student Evaluation Centre), Regional Centres and Study Centres etc. All the queries are solved by papers or through email. There is not any kind of support to IGNOU students to ask their academic related queries to Programme Coordinator or Course Coordinator. IGNOU is an open university. Therefore students don't attend daily classes and any kind of interaction with regular teachers. There is only Study Centre's facility, where students can interact with the Counselors for very limited period of time. Students are not able to directly interact with the School's faculty.

While carrying out system study, discussions where held with the senior officials in the fields of university Management and IT Professionals. During the discussions, as it found that over the years, the university Query Response was being done manually involving a lot of paper work as well as postal delay, which was the most time-consuming, unreliable and cumbersome activity. As a result of this, precious productive time is lost and the daily routine work remains pending and gets piling up and after sometimes, a situation of no comeback arises. Therefore, an urgent need was felt to introduce a computerized **'On line Query Management System'** which will provide them a solution to all the above referred bottlenecks of the existing manual system.

The system is developed to be able to provide accurate, fast and precise information on the current positions of the Web Based on line Query Response.
Various problems in existing manual system as explained below:-

1 **Inability of modification of data:-** The managing of huge data effectively and efficiently for efficient results, storing the details of the student etc. in such a way that the database can be modified as not possible in the manual system. Avoid redundancy and maintain consistency.

2 **Not User friendly:-** Any manual system normally is not user friendly because the retrieval and storing of data is slow and data is not maintained efficiently. Here comes the need of a GUI (Graphical user interface) based Web Application which is very much user Friendly.

3 **Difficulty in reports generation & Analysis:-** Either no report is generated in a manual system or they are generated with great difficulty. Reports take time to generate in the manual system. At times reports became useless by the time they are produced and the decision if any had already been taken of the assumption.

4 **Manual operator Control:-** Manual intervention is there and leads to a lots of chaos and errors.

5 **Lot of paper work:-** Existing system requires lot of paper work and even a small transaction require many papers fill. Moreover any unnatural cause (such as fire in the organization) can destroy all data of the organization. Some times few pages may lead to difficult situation.

6 **Postal Delay:-** Due to huge rush as well as due to current postal delay there were lot of uncertainties about the timely dispatch of the letters as well as receipt (ack.) of the form by the Universities.

3.2. Preliminary Investigation

We carried out preliminary investigation on IGNOU existing Student Feedback Form and came out with the following conclusions:

IGNOU Existing Student Feedback Form	**On Line IGNOU Prashnottari Sewa**
It is not capable of maintaining large number of records.	System is capable of maintaining, handling as well as processing large number of records.
Student has the option to send only limited number of predefined queries.	Any type of queries can be sent like the query may be related to Academic, Administrative, etc.
This facility is only for registered students of IGNOU.	In IPS any person can send query and get satisfactory response well in time .He is not required to be a registered student of IGNOU.
There is no provision to track the status of the query .In other words there is no accountability to the query as who should reply and within what time.	IPS has the tracking facility inbuilt which insures that the query is routed to the concerned department and replied well in time.
There is no time limit set for the query to be answered ultimately it is the student	IPS has an ALERT system which insures that the query is answered well in time(max

who has to take the brunt of it.	time is 24 hours) if not then the query is automatically transferred to the higher authority with a alert message.
There is no provision for file upload and download.	IPS has the file upload and download facilities in order to facilitate the student as well as the faculties to send some document if they feel so.
There is no provision for inter departmental query sending within IGNOU.	Through IPS one department can send query to another department and can get reply well in time.
No provision for Regional Centers.	IPS provides RC to RC / RC to SC / Dept to Dept / RC to Dept and vice versa interaction and message passing.
Current system has no provision for Report generation.	IPS automatically generates weekly/ Monthly and Yearly reports of all the queries received, replied as well as pending queries.
No provision for FAQ (Frequently Asked Questions).	IPS has self generated FAQ database which gets updated automatically as and when any common query is replied.

Final Report of Preliminary Investigation - Needs of the system to be designed

Keeping in view of the information gathered about the existing system, we compared the efficiencies of both the systems and noted down the results of the preliminary investigations which are given below:-

1. Existing system is slow, redundant and inefficient.

2. Supervision and maintenance in the existing system is not up to the standards.

3. Keeping records, generating reports and tracking the queries it is not possible.

4. No accountability to the queries whether it is being answered or not or whether the reply is up to the satisfaction level of the student or not.

5. Timely response of the query is not feasible.

3.3. Feasibility Study

A feasibility study is the process of planning to develop a new system or improving an existing one. It includes developing plans for designing the system, testing it, introducing it to the organization, and maintaining it. A feasibility study not only states that an idea is feasible; it includes data gathering, analysis, and design of the overall system process. The purpose of the feasibility study is to convert a goal desired by management into a plan to archive the goal.

After the system is proposed, the first system activity, which is the preliminary investigation, begins. The main objective of the preliminary investigation is to determine whether the system requested is feasible or not. After discussing goals and objectives for the new system in a review with the administrative, officers and the clerical staff working in different departments, the following conclusion was drawn.

To successfully initiate a feasibility study, the project leader must accomplish two tasks:

1. Clearly define the objectives of the study.
2. Device implementations plan so that the corporation can allocate the necessary resources to assist him.

Our team carried out the feasibility study of the project as follows:

(a) Technical Feasibility:-

This study is conducted to determine whether the current level of technology can support the proposed system or not.

Assessment Factors for Technical Feasibility	**Is it feasible in On Line IGNOU Prashnottari Sewa**
Knowledge of current and emerging technological solution.	Yes, The management is aware of the emerging technological solutions; that is why they had chosen Java/J2EE technology as front end in their software.
Availability of technical qualified staff in-house for the duration of the project and subsequent maintenance phase.	Yes, MySQL Server DBA's and Java Programmers are already available within the organization.
Availability of infrastructure in-house to support the development and maintenance of the proposed system.	Yes, after the study it was found that the company has WINDOWS 2000, XP word environment. The company has got all the required equipments (printers, hubs, LAN cards etc.)
Capacity of the proposed system to accommodate increasing levels of use over the medium term.	Yes, enough room would be provided in the proposed system to accommodate increasing levels of use over the medium term.
The capacity of the proposed system to meet initial performance expectations and accommodate new functionality over the medium term.	Yes, it will be feasible.

Hence, the system was found to be technically feasible.

(b) Economical Feasibility:-

This involves measurement of the cost effectiveness of the project. A system development project may be regarded as economically feasible or good value to the organization if its anticipated benefits outweigh its estimated costs.

However; many of the benefits arising from computerizing the On Line Query Management System are intangible and may be hard to quantify. Examples of intangible benefits include:

1 Improved; efficient and faster service providence.
2 Better management of registration procedure.
3 Improved consistency, continuity, efficiency, and productivity in service delivery, decision making, management and administration, etc.

Moreover, the resultant system will be to a great help to the company in future, as it will save the efforts and time involved to maintain the records and to prepare the reports. This way the project is found to be economically feasible.

(c) Operational Feasibility:-

This study is involved in conducting the extent the proposed system will fulfill the organization's requirements. That is whether the proposed system covers all aspects of the wording system and whether it has considerable improvements.

User acceptance is an important determinant of operational feasibility.

Assessment factors for Operational Feasibility	**Is it feasible in IGNOU Prashnottari Sewa**
Management support for the new system	Yes, the management has been extending its full support towards the project and it intends to do so in the future.
The nature and level of user involvement in the development and implementation of the system	Users help will be needed the most during the analysis phase where we are getting to know the system. Till now the response has been positive
Direct and indirect impacts of the new system on word practices.	The implementation of system will bring good impact on the work practices in more than one ways.
Anticipated performance and outcomes of the new system compared with the existing system.	The new Web based Query Management System is intended to improve the performance of the existing manual system.

After understanding the existing system and becoming aware of the limitations of the software being used, it was found that the system is functionally feasible i.e. it will perform all the functions which are expected out of it and will be able to replace the existing system.

3.4. Project Planning and Scheduling

Planning and scheduling is a complicate part of software development. Planning, for our purposes, can be though of as determining all the small tasks that must be carried out in order to accomplish the goal. Planning also takes into account, rules, and known as constraints, which control when certain tasks can or cannot happen. Scheduling can be thought of as determining whether adequate resources are available to carry out the plan. Gantt chart, Resource utilization chart and Pert Chart are given below that show how project was scheduled and resources were utilized.

3.4.1. Gantt Chart:-

A **Gantt chart** is a type of bar chart that illustrates a project schedule. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project. Terminal elements and summary elements comprise the work breakdown structure of the project. The Gantt charts for IPS project are given below:

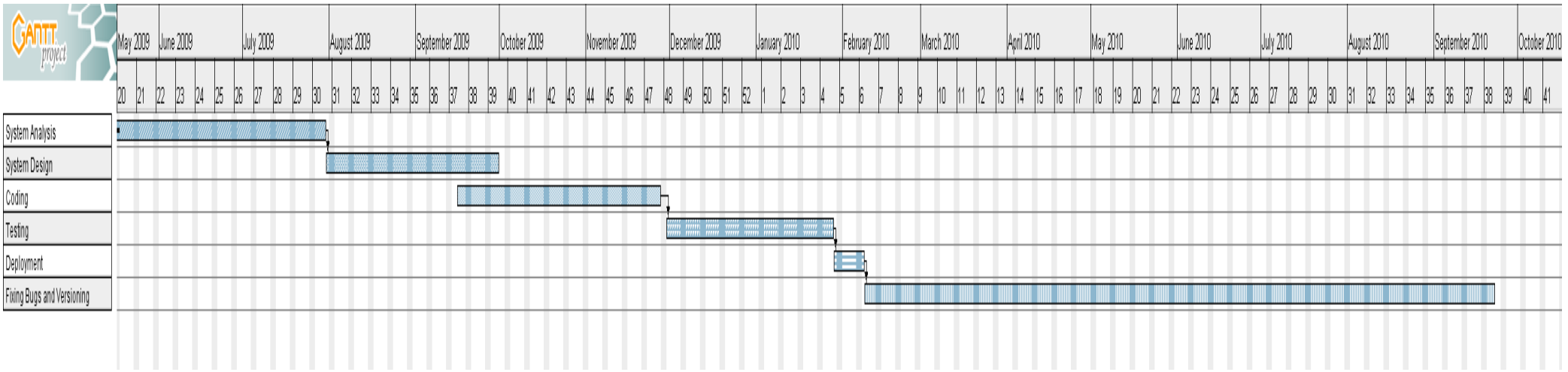


Fig.- Gantt Chart for Project Scheduling

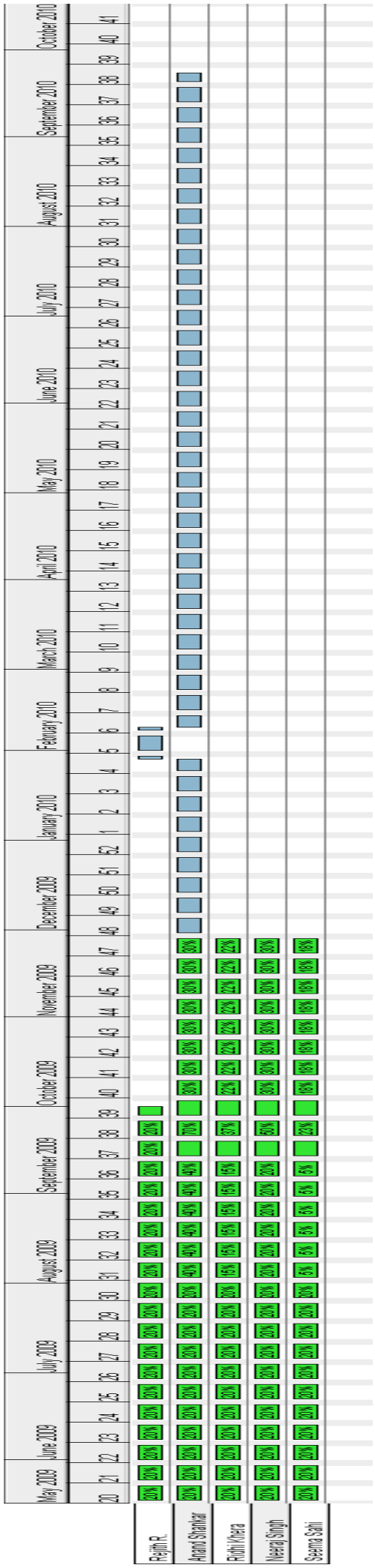
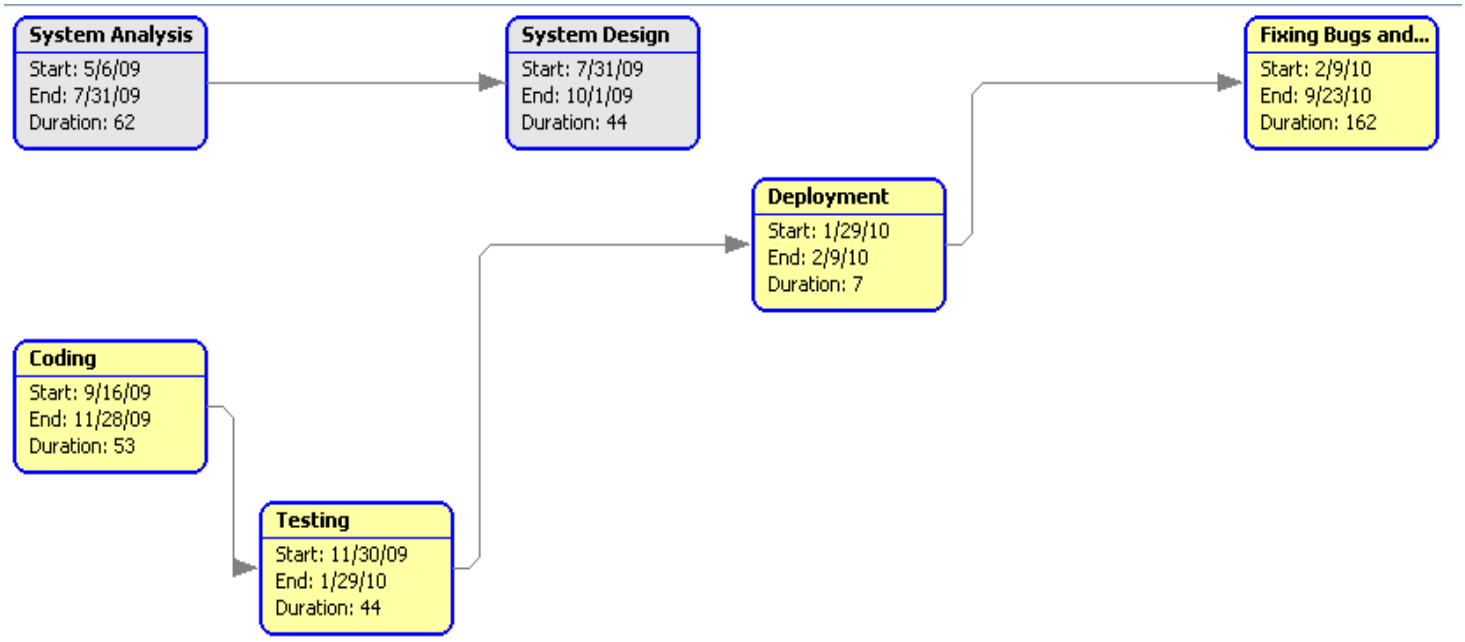


Fig.- Gantt Chart for Resource Scheduling

3.4.2. Pert Chart:-

PERT was developed primarily to simplify the planning and scheduling of large and complex projects. It was able to incorporate uncertainty by making it possible to schedule a project while not knowing precisely the details and durations of all the activities. It is more of an event-oriented technique rather than start- and completion-oriented, and is used more in projects where time, rather than cost, is the major factor. It is applied to very large-scale, one-time, complex, non-routine infrastructure and Research and Development projects. Pert Chart for IPS is given below:



3.5. Software Requirement Specification (SRS):-

The system should be developed in such a way as to serve as a decision support tool for the management. The management should be provided with a wide variety of informative reports, so that important decisions can be made on time. Track of each query and its response, analysis of whole data and the student satisfaction level is at the priority level.

3.5.1. Requirements of the new computerized system:-

1 It should provide correct and complete information.

3 It should be such that his information remains secured.

4 It must be cost effective.

5 It should be such that the information can easily be changed, when changes arise by authorized person.

6 Reports should be generated easily with correct information through computers.

7 There should be no or very few paper work.

8 Login for the Super Administration and Department Administration who want to see the system.

9 User Authentication.

10 User information.

11 Providing on line student query response like query related to student academic information to the administration staff of the university.

12 View the allotment of Token Id to the student whose query has been registered.

13 Maintaining statistics of the queries of the student in a particular month/year according to the records in the database.

3.5.2. Software and Hardware Requirements

ABSTRACT

Front End	**:**	JAVA / J2EE
Back End	**:**	My SQL Server
Platform	**:**	Linux

Hardware Specification for Developing the Application:

CPU:	P4 class processor or above.
RAM:	1 GB
Hard Disk (free space):	40 GB
Display Device:	VGA Monitor
Input Device:	Keyboard, Mouse

Software Specification for Developing the Application:

Programming Language:	JAVA / J2EE
Tool:	JDK 1.6
Compiler:	javac
Run Time Environment:	Java Virtual Machine
IDE:	NetBeans6.7, DreamWeaver
Database:	My SQL Server
Application Server:	GlassFishV3
Operating System:	Linux (Ubuntu)
Documentation Tools:	Open Office, VP Suite, Dia, *C11_2000.*

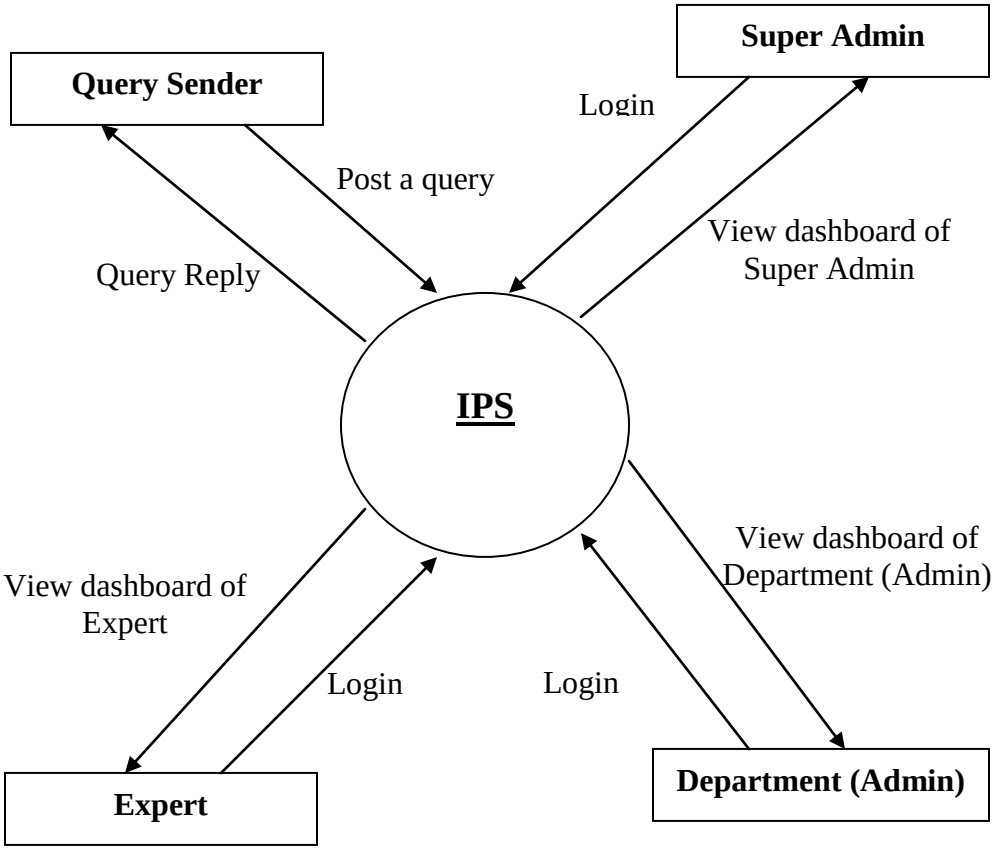
Note: - Hardware configuration is decided with several considerations in mind like speedy processing, easy availability, high efficiency, hardware requirements of operating system and other software tools used in the project, cost considerations, easy maintainability etc.

3.6. Data Flow Diagram (DFD):-

Data flow diagrams (also data flow graphs) are commonly used during problem analysis. Data flow diagrams (DFDs) are quite general and are not limited to problem analysis for software requirements specification.

A DFD shows the flow of data through a system. It views a system as a function that transforms the inputs into desired outputs. Any complex system will not perform the transformation in a "single step," and a data will typically undergo a series of transformations before it becomes the output. The DFD aims to capture the transformations that take place within a system to the input data so that eventually the output data is produced. The agent that performs the transformation of data from one state to another is called a process (or a bubble). So, a DFD shows the movement of data through the different transformations of processed in the system. The Data Flow Diagrams of for IPS are given below:

0-Level DFD


Super Admin
Query Sender
Login
Post a query
View dashboard of
Super Admin
Query Reply
IPS
View dashboard of
Department (Admin)
View dashboard of
Expert
Login
Login
Department (Admin)
Expert

Description of 0- level DFD

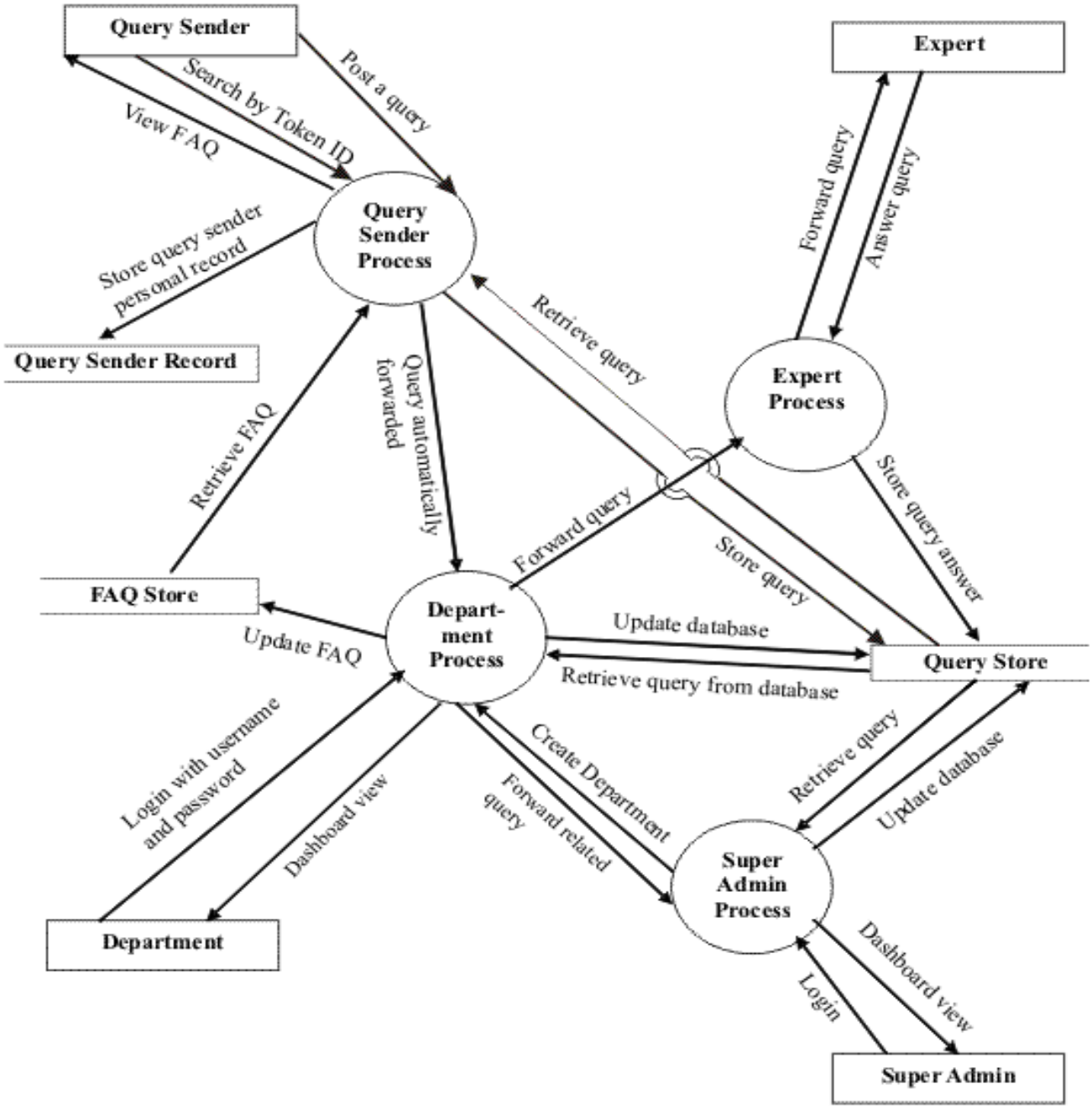
Query Sender:- A person can post any query and get answer of the posted query by Expert through E- mail. If that query was useful to give common information then query and answer of the query will be listed in FAQ (Frequently Asked Question) and anyone can see.

Expert:- Expert might be a counselor or an employee of IGNOU who will be expert in own subject and subject related queries will be send to consult expert by Department. When an Expert login with own username and password the dashboard of Expert will be displayed.

Department (Admin):- Department might be a School or Division of IGNOU. A Department will be working as an Admin which has a number expert related to subject and forward queries to consult expert. When a Department login with own username and password the dashboard of that Department will be displayed.

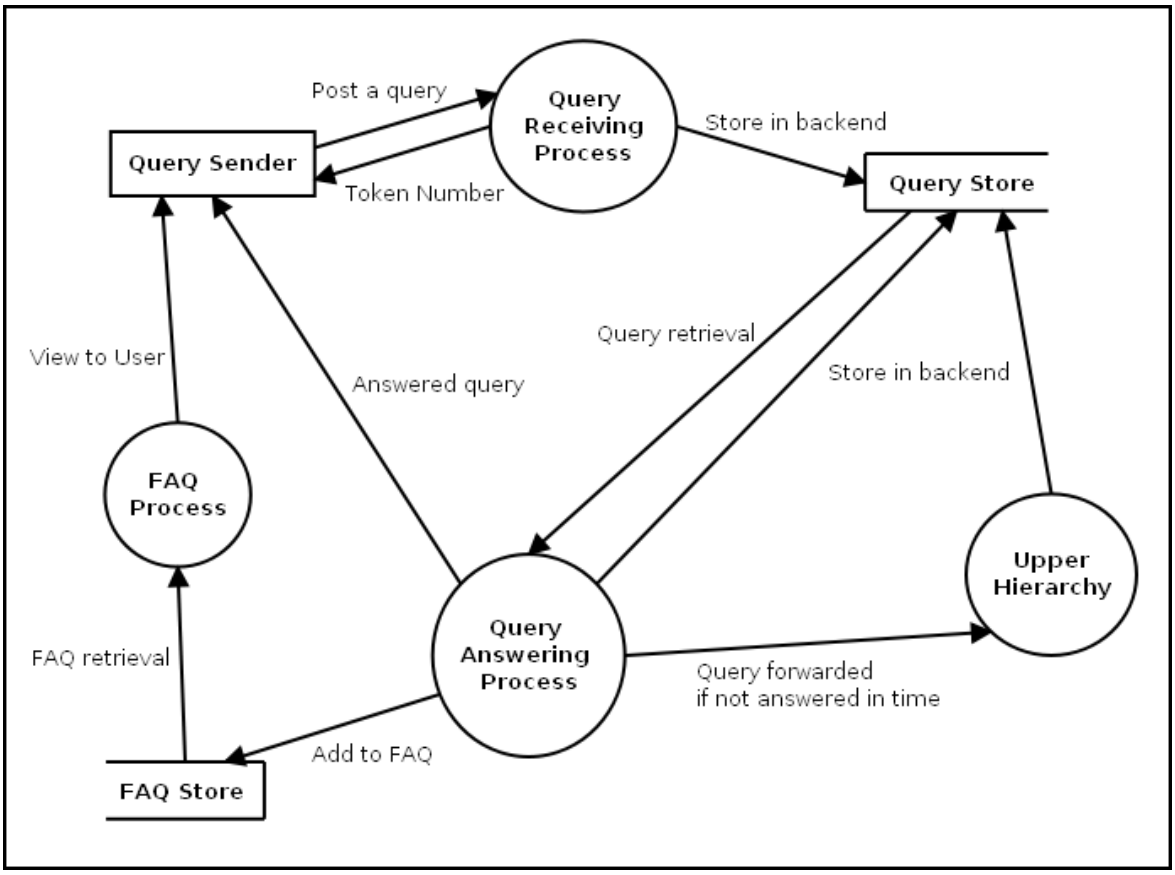
Super Admin:- Super admin can control whole system as Department, Expert and all the queries. Super Admin has own secret username and password. When Super Admin login its dashboard will be open, where he will get all the option for controlling.

1- Level DFD


Query Sender
Post a query
Search by Token ID
View FAQ
Query Sender Process
Store query sender personal record
Query Sender Record
Retrieve FAQ
Query automatically forwarded
Retrieve query
Expert
Forward query
Answer query
Expert Process
Store query answer
Forward query
Store query
FAQ Store
Depart-ment Process
Update FAQ
Update database
Query Store
Retrieve query from database
Login with username and password
Dashboard view
Create Department
Forward related query
Retrieve query
Update database
Super Admin Process
Department
Dashboard view
Login
Super Admin

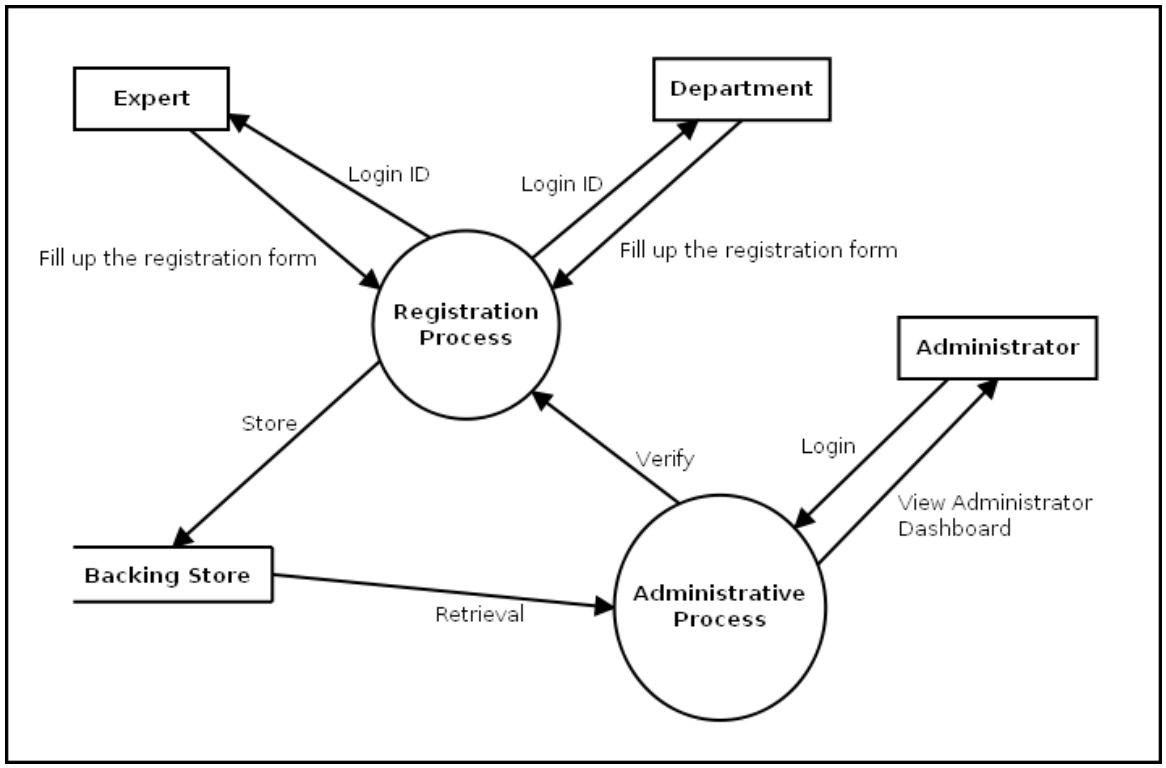
2-Level DFD

(Query Sending and Answering Process)

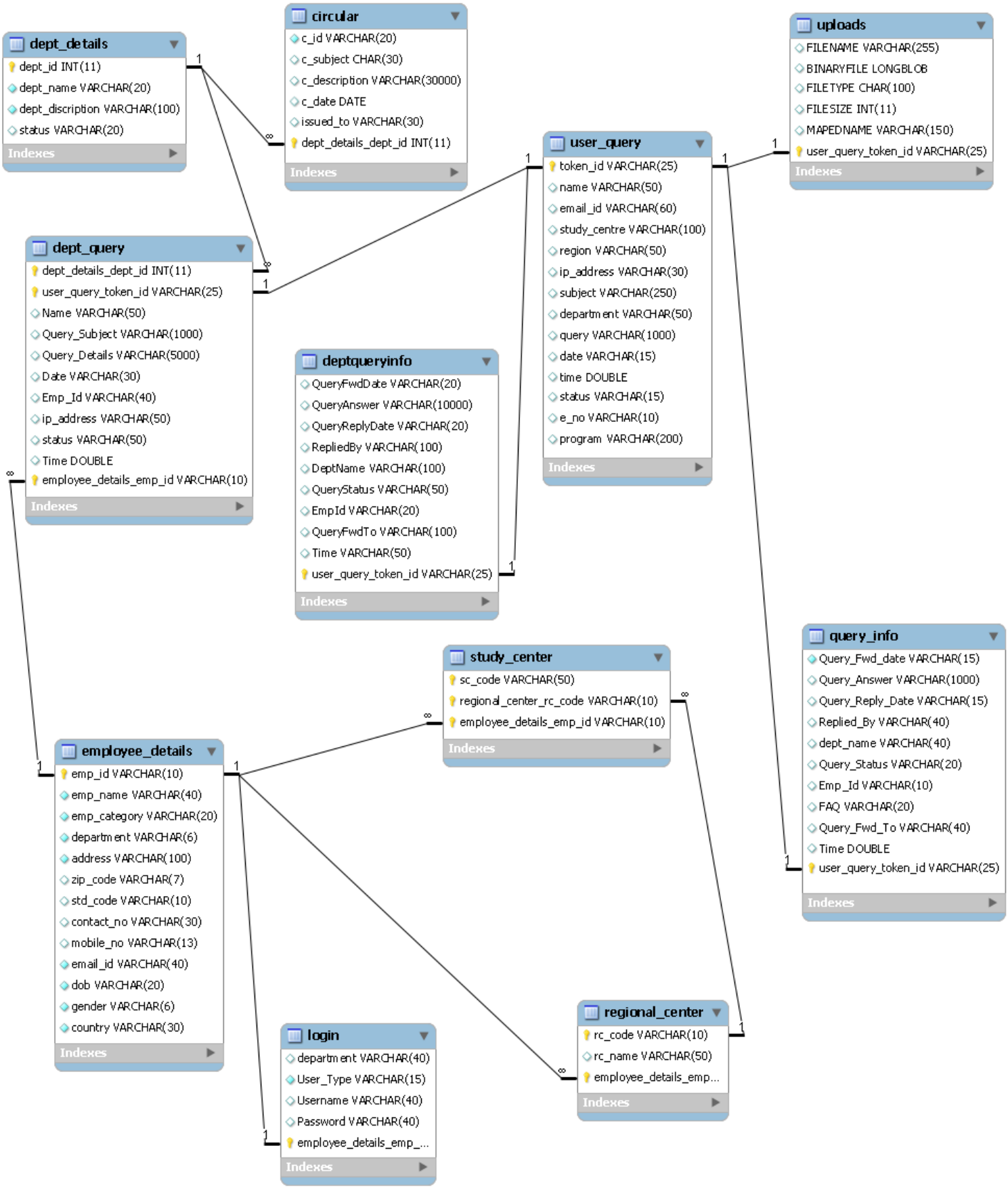


2-Level DFD

(Expert/Department Registration Process)



3.7. Entity Relationship Diagram (ERD):-



4. System Design

4.1. Design Overview:-

Design is the phase where quality is forecasting in software development. Design provides with representative of software that can be accessed for quality. Design is the only way that we can accurately accessed for quality. During design we make decisions that well ultimately affect the success of software construction and as important, the ease with which software can be maintained.

Design is the only way that can accurately translate a customer's requirement into a finished system or product. Software design serves as the foundation of all steps of software engineering and maintenance that follows. Without design, we risk building an unusable system, one that fail when small changes are made or one that may be difficult to test. Three characteristics that serve as a guide for the evaluation of good design:-

1. The design must implement the entire explicit requirement in the analysis model and it must accommodate the entire implicit requirement desired by the customer.
2. The design must be readable and understandable guide for those who test code and subsequently maintain the software.

1 The design should provide a complete picture of the software, addressing the data functional and behavioral domains from an implement perspective.

Our system is organized into subsystems based on both the analysis structure and the proposed architecture. System design is the first design stage in which the basic Approach to solving the problem is selected. During system design, the overall structure and style are decided. The system architecture is the overall organization of the system into components called subsystem.

4.2 Design Principle:-

Software design is both a process and a model. The design process is set of iterative steps that enable the designer to describe all aspects software to be built. Basic design principle includes:-

1 The design process should not suffer from tunnel vision.
2 The design should not be traceable to the analysis model.
3 The design should exhibit uniformity and integration.
4 The design should be structured to accommodate change.
5 The design should be reviewed to minimize conceptual errors.
6 The design should be addressed of quality as it being created.

4.3 Design Concept:-

Design concepts provide the software designer with a foundation from which more sophisticated design can be applied. During detailed design the data structure and the algorithms used by different modules are designed. The outcome of detailed is usually known as the Module Specification Document.

PROCESS LOGIC

1. **IGNOU Student Process**
2. **International Student Process**
3. **Others (General User) Process**
4. **View Query Process**
5. **User Authentication Process**
6. **IGNOU Internal Operation Process**

4.4. Modularization Details:-

Modular programming is a software design technique that increases the extent to which software is composed of separate, interchangeable components, called **modules**. Conceptually, modules represent a separation of concerns, and improve maintainability by enforcing logical boundaries between components. Modules are typically incorporated into the program through interfaces. A module interface expresses the elements that are provided and required by the module. The elements defined in the interface are detectable by other modules. The implementation contains the working code that corresponds to the elements declared in the interface. The IPS project has divided mainly into four modules:

1. **Query Sender Module**
2. **Expert Module**
3. **Department Admin Module**
4. **Super Admin Module**
5. **File Upload and Download Module**
6. **Circular Module**

All these modules are describes above in detail.

4.5. Database Design:-

Tables Structures

qmsdb

circular

dept_details

dept_query

deptqueryinfo

employee_details

login

query_info

regional_center

study_center

uploads

user_query

circular

Fields

Field	Type	Collation	Null	Key	Default	Extra	Privileges
c_id	varchar(20)	latin1_swedish_ci	NO	UNI	(NULL)		select,insert,update,references
c_subject	char(30)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
c_description	varchar(30000)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
c_date	date	(NULL)	YES		(NULL)		select,insert,update,references
issued_to	varchar(30)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
dept_details_dept_id	int(11)	(NULL)	NO	PRI	(NULL)		select,insert,update,references

Indexes

Table	Non unique	Key name	Seq in index	Column name	Collation	Cardinality	Sub part	Packed	Null	Index type
circular	0	PRIMARY	1	dept_details_dept_id	A	0	(NULL)	(NULL)		BTREE
circular	0	c_id	1	c_id	A	0	(NULL)	(NULL)		BTREE
circular	1	fk_circular_dept_details1	1	dept_details_dept_id	A	(NULL)	(NULL)	(NULL)		BTREE

dept_details

Fields

Field	Type	Collation	Null	Key	Default	Extra	Privileges
dept_id	int(11)	(NULL)	NO	PRI	(NULL)		select,insert,update,references
dept_name	varchar(20)	latin1_swedish_ci	NO		(NULL)		select,insert,update,references
dept_discription	varchar(100)	latin1_swedish_ci	NO		(NULL)		select,insert,update,references
status	varchar(20)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references

Indexes

Table	Non unique	Key name	Seq in index	Column name	Collation	Cardinality	Sub part	Packed	Null	Index type
dept_details	0	PRIMARY	1	dept_id	A	0	(NULL)	(NULL)		BTREE

dept_query

Fields

Field	Type	Collation	Null	Key	Default	Extra	Privileges
dept_details_dept_id	int(11)	(NULL)	NO	PRI	(NULL)		select,insert,update,references
user_query_token_id	varchar(25)	latin1_swedish_ci	NO	PRI	(NULL)		select,insert,update,references
Name	varchar(50)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
Query_Subject	varchar(1000)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
Query_Details	varchar(5000)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
Date	varchar(30)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
Emp_Id	varchar(40)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
ip_address	varchar(50)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
status	varchar(50)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
Time	double	(NULL)	YES		(NULL)		select,insert,update,references
employee_details_emp_id	varchar(10)	latin1_swedish_ci	NO	PRI	(NULL)		select,insert,update,references

Indexes

Table	Non unique	Key name	Seq in index	Column name	Collation	Cardinality	Sub part	Packed	Null	Index type
dept_query	0	PRIMARY	1	dept_details_dept_id	A	0	(NULL)	(NULL)		BTREE
dept_query	0	PRIMARY	2	employee_details_emp_id	A	0	(NUL	(NULL)		BTREE

							L)			
dept_query	0	PRIMARY	3	user_query_token_id	A	0	(NUL L)	(NULL)		BTREE
dept_query	1	fk_dept_query_dept_details1	1	dept_details_dept_id	A	0	(NUL L)	(NULL)		BTREE
dept_query	1	fk_dept_query_employee_details1	1	employee_details_emp_id	A	0	(NUL L)	(NULL)		BTREE
dept_query	1	fk_dept_query_user_query1	1	user_query_token_id	A	0	(NUL L)	(NULL)		BTREE

Foreign Key Relationships

FK Id	Reference Table	Source Column	Target Column	Extra Info
fk_dept_query_dept_details1	dept_details	`dept_details_dept_id`	`dept_id`	ON DELETE NO ACTION ON UPDATE NO ACTION,
fk_dept_query_employee_details1	employee_details	`employee_details_emp_id`	`emp_id`	ON DELETE NO ACTION ON UPDATE NO ACTION,
fk_dept_query_user_query1	user_query	`user_query_token_id`	`token_id`	ON DELETE NO ACTION ON UPDATE NO ACTION

deptqueryinfo

Fields

Field	Type	Collation	Null	Key	Default	Extra	Privileges
QueryFwdDate	varchar(20)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
QueryAnswer	varchar(10000)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
QueryReplyDate	varchar(20)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
RepliedBy	varchar(100)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
DeptName	varchar(100)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
QueryStatus	varchar(50)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
EmpId	varchar(20)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
QueryFwdTo	varchar(100)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
Time	varchar(50)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
user_query_token_id	varchar(25)	latin1_swedish_ci	NO	PRI	(NULL)		select,insert,update,references

Indexes

Table	Non unique	Key name	Seq in index	Column name	Collation	Cardinality	Sub part	Packed	Null	Index type
deptqueryinfo	0	PRIMARY	1	user_query_token_id	A	0	(NULL)	(NULL)		BTREE
deptqueryinfo	1	fk_deptqueryinfo_user_query1	1	user_query_token_id	A	(NULL)	(NULL)	(NULL)		BTREE

employee_details

Fields

Field	Type	Collation	Null	Key	Default	Extra	Privileges
emp_id	varchar(10)	latin1_swedish_ci	NO	PRI	(NULL)		select,insert,update,references
emp_name	varchar(40)	latin1_swedish_ci	NO		(NULL)		select,insert,update,references
emp_categor	varchar(20)	latin1_swedis	NO		(NULL)		select,insert,update,references

y		h_ci					
department	varchar(6)	latin1_swedish_ci	NO		(NULL)		select,insert,update,references
address	varchar(100)	latin1_swedish_ci	NO		(NULL)		select,insert,update,references
zip_code	varchar(7)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
std_code	varchar(10)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
contact_no	varchar(30)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
mobile_no	varchar(13)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
email_id	varchar(40)	latin1_swedish_ci	NO		(NULL)		select,insert,update,references
dob	varchar(20)	latin1_swedish_ci	NO		(NULL)		select,insert,update,references
gender	varchar(6)	latin1_swedish_ci	NO		(NULL)		select,insert,update,references
country	varchar(30)	latin1_swedish_ci	NO		(NULL)		select,insert,update,references

Indexes

Table	Non unique	Key name	Seq in index	Column name	Collation	Cardinality	Sub part	Packed	Null	Index type
employee_details	0	PRIMARY	1	emp_id	A	0	(NULL)	(NULL)		BTREE

login

Fields

Field	Type	Collation	Null	Key	Default	Extra	Privileges
department	varchar(40)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
User_Type	varchar(15)	latin1_swedish_ci	NO		(NULL)		select,insert,update,references
Username	varchar(40)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
Password	varchar(40)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
employee_details_emp_id	varchar(10)	latin1_swedish_ci	NO	PRI	(NULL)		select,insert,update,references

Indexes

Table	Non unique	Key name	Seq in index	Column name	Collation	Cardinality	Sub part	Packed	Null	Index type
login	0	PRIMARY	1	employee_details_emp_id	A	0	(NULL)	(NULL)		BTREE
login	1	fk_login_employee_details1	1	employee_details_emp_id	A	0	(NULL)	(NULL)		BTREE

Foreign Key Relationships

FK Id	Reference Table	Source Column	Target Column	Extra Info
fk_login_employee_details1	employee_details	`employee_details_emp_id`	`emp_id`	ON DELETE NO ACTION ON UPDATE NO ACTION

query_info

Fields

Field	Type	Collation	Null	Key	Default	Extra	Privileges
Query_Fwd_date	varchar(15)	latin1_swedish_ci	NO		(NULL)		select,insert,update,references
Query_Answer	varchar(1000)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
Query_Reply_Date	varchar(15)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
Replied_By	varchar(40)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
dept_name	varchar(40)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
Query_Status	varchar(20)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
Emp_Id	varchar(10)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
FAQ	varchar(20)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
Query_Fwd_To	varchar(40)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
Time	double	(NULL)	YES		(NULL)		select,insert,update,references
user_query_token_id	varchar(25)	latin1_swedish_ci	NO	PRI	(NULL)		select,insert,update,references

Indexes

Table	Non unique	Key name	Seq in index	Column name	Collation	Cardinality	Sub part	Packed	Null	Index type
query_info	0	PRIMARY	1	user_query_token_id	A	0	(NULL)	(NULL)		BTREE
query_info	1	fk_query_info_user_query	1	user_query_token_id	A	0	(NULL)	(NULL)		BTREE

Foreign Key Relationships

FK Id	Reference Table	Source Column	Target Column	Extra Info
fk_query_info_user_query	user_query	`user_query_token_id`	`token_id`	ON DELETE NO ACTION ON UPDATE NO ACTION

regional_center

Fields

Field	Type	Collation	Null	Key	Default	Extra	Privileges
rc_code	varchar(10)	latin1_swedish_ci	NO	PRI	(NULL)		select,insert,update,references
rc_name	varchar(50)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
employee_details_emp_id	varchar(10)	latin1_swedish_ci	NO	PRI	(NULL)		select,insert,update,references

Indexes

Table	Non unique	Key name	Seq in index	Column name	Collation	Cardinality	Sub part	Packed	Null	Index type
regional_center	0	PRIMARY	1	rc_code	A	0	(NULL)	(NULL)		BTREE
regional_center	0	PRIMARY	2	employee_details_emp_id	A	0	(NULL)	(NULL)		BTREE
regional_center	1	fk_regional_center_employee_details1	1	employee_details_emp_id	A	0	(NULL)	(NULL)		BTREE

Foreign Key Relationships

FK Id	Reference Table	Source Column	Target Column	Extra Info
fk_regional_center_employee_details1	employee_details	`employee_details_emp_id`	`emp_id`	ON DELETE NO ACTION ON UPDATE NO ACTION

study_center

Fields

Field	Type	Collation	Null	Key	Default	Extra	Privileges
sc_code	varchar(50)	latin1_swedish_ci	NO	PRI	(NULL)		select,insert,update,references
regional_center_rc_code	varchar(10)	latin1_swedish_ci	NO	PRI	(NULL)		select,insert,update,references
employee_details_emp_id	varchar(10)	latin1_swedish_ci	NO	PRI	(NULL)		select,insert,update,references

Indexes

Table	Non unique	Key name	Seq in index	Column name	Collation	Cardinality	Sub part	Packed	Null	Index type
study_center	0	PRIMARY	1	sc_code	A	0	(NULL)	(NULL)		BTREE
study_center	0	PRIMARY	2	regional_center_rc_code	A	0	(NULL)	(NULL)		BTREE
study_center	0	PRIMARY	3	employee_details_emp_id	A	0	(NULL)	(NULL)		BTREE
study_center	1	fk_study_center_regional_center1	1	regional_center_rc_code	A	0	(NULL)	(NULL)		BTREE
study_center	1	fk_study_center_employee_details1	1	employee_details_emp_id	A	0	(NULL)	(NULL)		BTREE

Foreign Key Relationships

FK Id	Reference Table	Source Column	Target Column	Extra Info
fk_study_center_employee_details1	employee_details	`employee_details_emp_id`	`emp_id`	ON DELETE NO ACTION ON UPDATE NO ACTION,
fk_study_center_regional_center1	regional_center	`regional_center_rc_code`	`rc_code`	ON DELETE NO ACTION ON UPDATE NO ACTION

uploads

Fields

Field	Type	Collation	Null	Key	Default	Extra	Privileges
FILENAME	varchar(255)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
BINARYFILE	longblob	(NULL)	YES		(NULL)		select,insert,update,references
FILETYPE	char(100)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
FILESIZE	int(11)	(NULL)	YES		(NULL)		select,insert,update,references
MAPEDNAME	varchar(150)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
user_query_token_id	varchar(25)	latin1_swedish_ci	NO	PRI	(NULL)		select,insert,update,references

Indexes

Table	Non unique	Key name	Seq in index	Column name	Collation	Cardinality	Sub part	Packed	Null	Index type
uploads	0	PRIMARY	1	user_query_token_id	A	0	(NULL)	(NULL)		BTREE
uploads	1	fk_uploads_user_query1	1	user_query_token_id	A	0	(NULL)	(NULL)		BTREE

Foreign Key Relationships

FK Id	Reference Table	Source Column	Target Column	Extra Info
fk_uploads_user_query1	user_query	`user_query_token_id`	`token_id`	ON DELETE NO ACTION ON UPDATE NO ACTION

user_query

Fields

Field	Type	Collation	Null	Key	Default	Extra	Privileges
token_id	varchar(25)	latin1_swedish_ci	NO	PRI	(NULL)		select,insert,update,references
name	varchar(50)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
email_id	varchar(60)	latin1_swedish	YES		(NULL)		select,insert,update,references

		_ci					
study_centre	varchar(100)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
region	varchar(50)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
ip_address	varchar(30)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
subject	varchar(250)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
department	varchar(50)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
query	varchar(1000)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
date	varchar(15)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
time	double	(NULL)	YES		(NULL)		select,insert,update,references
status	varchar(15)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
e_no	varchar(10)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references
program	varchar(200)	latin1_swedish_ci	YES		(NULL)		select,insert,update,references

Indexes

Table	Non unique	Key name	Seq in index	Column name	Collation	Cardinality	Sub part	Packed	Null	Index type
user_query	0	PRIMARY	1	token_id	A	0	(NULL)	(NULL)		BTREE

5. Coding

5.1. Complete Project Coding:-

The IPS project is a extension of **Vedyadhara** and mainly it implemented for Indira Gandhi National Open University. IPS is developed using Open Source tools only and distributed in Open Source Community of the world. The complete code of IPS can be downloaded from

http://code.google.com/p/querymanagementsystem/downloads/list

IPS is totally free and Open Source software developed and distributed under LGPL license. Therefore code is not included in this document. Link is provided for downloading IPS with complete source code and documentation.

5.2. Standardization of the coding and Comments and Descriptions:-

The IPS is developed and distributed in Open Source project. Therefore the coding standard of IPS is as Open Source Software standard which comes under LGPL license. IPS is developed using Java/J2EE Technology. All the JSP pages, Servlets and Bean classes started with the general information of Author, Date and time of the file creation, etc. Each line of the code written with proper comments that, what is the purpose of the writing the code, from where it is linked, etc.

5.3. Code Efficiency:-

The goal of coding or programming phase is to translate the design of the system produced during the design phase into code in a given programming language, which can be executed by a computer and that performs the computation specified by the design.

The coding phase affects both testing and maintenance profoundly. Thus, it should be clear that the goal during coding should not be to reduce the implementation cost, but the goal should be to reduce the cost of later phases, even if it means that the cost of this phase has to increase.

The efficiency of the code can be determined on various criteria:

1 Readability
2 Size of the program
3 Execution time
3 Understandability

The whole coding is done in a modular structure. The coding of each module is done separately. Unit testing is done after the completion of each section .the sections are integrated after completion of each section. Integration testing is performed at the end.

Since the coding is done in the modular structure, on the basis of their functionality and complexity involved in it, the modules are again decomposed into sub modules. Let's see whether the coding of IPS meets the criteria for the code efficiency or not.

Readability: The coding of web based course registration form is readable enough because of the modular structure pattern. Further, the names and variable declarations itself presents the functionality of the code.

Program size: The size of the program is optimized to a great extent. For instance, the coding is being reused at several places, in other words inheritance is implemented which is also one of the feature available with JAVA /J2EE.

Execution time: **E**xecution time is not so high. For instance, it uses a disconnected architecture as far as database connectivity is concerned. Since, web based course registration form is the database driven application so execution time should be quite. As because of the disconnected architecture the data which all can be required in the future it stores maintains it in the local memory. Thus next time when the query executes it first checks for it in the memory if not found then requests the database server for the response.

Understandability: The code is very much understandable because of the incorporation of comments wherever required, further the names of the variables as well as procedures itself tells it function.

5.4. Optimization of Code:-

A reasonably skilled programmer will not write a grossly inefficient program. At least not deliberately. Optimization is what you do when the performance is insufficient. Sometimes the optimizations are easy, sometimes they are hard.

Sometimes they fly in the face of your original design, and sometimes they require that you grossly violate your beautiful abstractions in your class system.

5.5. Exception and Error Handling:-

A good application is written always with proper Exception and error handler because this is common that running software throws exception or error in some condition like Memory overflow, Stack overflow, Null pointer exception, Resource not found, etc. In these cases program are terminated by Virtual Machine. So, to avoid the termination of the program we add error pr exception handler so that exception or error can be caught and appropriate action can be taken. IPS follows these standards. Wherever needed, try and catch blocks are placed in all java files to handle the exception. All JSP page's exceptions are handled by an error handler JSP page. If any JSP page throws exception then it will be handled by a JSP error handler. This JSP error handler catches the errors and traces it; if we want.

5.6. Validation Checks:-

Every JSP page which takes input from the user, are validated on client side using Java Script so that server trip can be minimized. For example, IPS login page ask to user to enter their login id and password to see their dashboard and queries. Username field does not accept any character or symbol except A-Z, a-z, 0-9 and underscore (_). If user does mistake then a message will pop up to fill up the correct characters in the field. Like this all the form are validated properly at client side using Java Script. Some validations and verifications done at Server side also like username and password, the type and size of the attachment, etc.

6. Testing

Testing is the most important activity in any project development. Testing involves operation of a system under controlled conditions and evaluating the results. Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding testing requires that the review of specification, design and coding testing requires that the developer discard preconceive notions of the " correctness" of software just developed and overcome a conflict of interest that occurs when errors are uncovered.

Testing is the process of executing a program care is one that has a high probability of finding as yet undiscovered errors. A successful test is one that uncovers an as-yet undiscovered error. Testing is the necessary part of the project the various methods are used for testing the software. The new look to the project is giving by the perfect testing. There are White box and Black box testing. Black-box testing focuses on the functional requirements of the software. That is black-box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program. Black box testing is not an alternative to white-box techniques. Rather it is a complementary approach that is likely to uncover a different class of errors than white-box methods.

Black-box testing attempts to find errors in following categories:-

- Incorrect or missing functions.
- Interface errors.
- Errors in data structures or external data base access.
- Performance errors.
- Initialization and termination errors.

Firstly test the incorrect or missing function in the project design, testing all modules and their dependent modules such as for new care, detail of old care, forwarding of case reminders of case and disposal of case.

Secondary, test the interfaces of the software. The inputs and outputs of the project also depend upon the interface use, so that all links should be displayed accordingly.
Thirdly, testing on the data that are used, it should be properly fitted. The connectivity should be proper.
Fourthly, the performance error, testing is also done on the performance error, checking the display time, loading time, clear picture and message.
Finally testing on the idealization and termination error, test the proper loading of the input data easily connection of the data and display of cases. In the project development, I divided the testing process into unit testing and integration testing. The **IPS** is tested as the following testing mechanism:

Unit Testing

Unit testing also called module or component testing is testing individual software units independently of the other units in the same system. Unit testing is done for each sub module of a system run in isolation, separate from any other sub module. Unit testing is recognized as one of the most efficient ways to reduce the density and proliferation of errors in a software application. Unit testing of IPS has done both manually as well as by Automatic Testing Tool (NetBeans6.9).

Integration Testing

Integration testing is done when unit testing has already been done for each separate module and all modules have been integrated. In integration testing, application is tested with respect to its typical working environment. Consequently for many process no clear division between validation and system testing can be made. After integration of all the modules of IPS an Integration testing has done manually.

Note:- IPS is tested manually as well as though Automatic Testing Tool. We used NetBeans6.9 for jUnit Testing. Some jUnit Testing Report is given below and rest of the Test Reports are uploaded in SVN server.

Test cases for IPSConnection class

```
package IPSdb;


import java.sql.Connection;
import org.junit.After;
import org.junit.AfterClass;
import org.junit.Before;
import org.junit.BeforeClass;
import org.junit.Test;
import static org.junit.Assert.*;


/**
*
* @author anand
*/
public class IPSConnectionTest {


public IPSConnectionTest() {
}


@BeforeClass
public static void setUpClass() throws Exception {
```

```
}

@AfterClass
public static void tearDownClass() throws Exception {
}

@Before
public void setUp() {
}

@After
public void tearDown() {
}

/**
* Test of getConnection method, of class IPSConnection.
*/
@Test
public void testGetConnection() {
System.out.println("getConnection");
IPSConnection instance = new IPSConnection();
Connection expResult = null;
Connection result = instance.getConnection();
```

```
assertEquals(expResult, result);
// TODO review the generated test code and remove the default call to fail.
fail("The test case is a prototype.");
}


}
```

Test cases for User_Query class

```
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.junit.After;
import org.junit.AfterClass;
import org.junit.Before;
import org.junit.BeforeClass;
import org.junit.Test;
import static org.junit.Assert.*;

/**
*
* @author anand
*/
public class User_QueryTest {

public User_QueryTest() {
}

@BeforeClass
public static void setUpClass() throws Exception {
}
```

```
@AfterClass
public static void tearDownClass() throws Exception {
}

@Before
public void setUp() {
}

@After
public void tearDown() {
}

/**
* Test of doPost method, of class User_Query.
*/
@Test
public void testDoPost() throws Exception {
System.out.println("doPost");
HttpServletRequest request = null;
HttpServletResponse response = null;
User_Query instance = new User_Query();
instance.doPost(request, response);
```

```
// TODO review the generated test code and remove the default call to fail.

fail("The test case is a prototype.");

}


}
```

Test cases for CentralControl class

```
package ignou.aciil.ips;


import java.util.Hashtable;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.junit.After;
import org.junit.AfterClass;
import org.junit.Before;
import org.junit.BeforeClass;
import org.junit.Test;
import static org.junit.Assert.*;


/**
*
* @author anand
*/
public class CentralControlTest {


public CentralControlTest() {
}
```

```
@BeforeClass
public static void setUpClass() throws Exception {
}

@AfterClass
public static void tearDownClass() throws Exception {
}

@Before
public void setUp() {
}

@After
public void tearDown() {
}

/**
* Test of processRequest method, of class CentralControl.
*/
@Test
public void testProcessRequest() throws Exception {
System.out.println("processRequest");
HttpServletRequest request = null;
```

```
HttpServletResponse response = null;
CentralControl instance = new CentralControl();
Hashtable expResult = null;
Hashtable result = instance.processRequest(request, response);
assertEquals(expResult, result);
// TODO review the generated test code and remove the default call to fail.
fail("The test case is a prototype.");
}

/**
* Test of doGet method, of class CentralControl.
*/
@Test
public void testDoGet() throws Exception {
System.out.println("doGet");
HttpServletRequest request = null;
HttpServletResponse response = null;
CentralControl instance = new CentralControl();
instance.doGet(request, response);
// TODO review the generated test code and remove the default call to fail.
fail("The test case is a prototype.");
}
```

```
/**
* Test of doPost method, of class CentralControl.
*/
@Test
public void testDoPost() throws Exception {
System.out.println("doPost");
HttpServletRequest request = null;
HttpServletResponse response = null;
CentralControl instance = new CentralControl();
instance.doPost(request, response);
// TODO review the generated test code and remove the default call to fail.
fail("The test case is a prototype.");
}

/**
* Test of getServletInfo method, of class CentralControl.
*/
@Test
public void testGetServletInfo() {
System.out.println("getServletInfo");
CentralControl instance = new CentralControl();
String expResult = "";
String result = instance.getServletInfo();
```

```
assertEquals(expResult, result);
// TODO review the generated test code and remove the default call to fail.
fail("The test case is a prototype.");
}


}
```

Test suit for IPS package

```
package ignou.aciil.ips;


import org.junit.After;

import org.junit.AfterClass;

import org.junit.Before;

import org.junit.BeforeClass;

import org.junit.runner.RunWith;

import org.junit.runners.Suite;


/**

*

* @author anand

*/

@RunWith(Suite.class)

@Suite.SuiteClasses({ignou.aciil.ips.UploadTest.class,ignou.aciil.ips.DownloadFileTes
t.class,ignou.aciil.ips.CentralControlTest.class})

public class IpsSuite {


@BeforeClass

public static void setUpClass() throws Exception {

}
```

```
@AfterClass
public static void tearDownClass() throws Exception {
}

@Before
public void setUp() throws Exception {
}

@After
public void tearDown() throws Exception {
}

}
```

7. System Security Measures

To secure the IPS Application tier, Database tier and Network tier are very secure and securities are implemented as follows:

Security on Application Tier:- There are given below that how many securities are implemented at application level:

- Passwords are stored in database in encrypted format of 128 bit encryption.
- There are roll based login for users. Therefore a user cannot see the other user's data.
- A supper Administrator controls over the users.
- Login pages are very secure through HTTPS.
- Admin port of the Application Server is not open to access from outside the IGNOU campus and authorized person with given login ID and password.

Security on Database Tier:-

- MySQL firewall is installed between Application Server and MySQL Server to prevent database from SQL injections and others database attach by hackers or crackers.
- Database is protected through username and password. Only authorized user can access the database.
- Database connection from the application level is very secure. From the application level only DML (Data Manipulation Language) queries can be invoked; DDL (Data Definition Language) and DCL (Data Control Language) Query can't be invoked from application tier.
- To prevent SQL injection attack by the hackers IPS use a java security like Prepared Statement. So that hacker can not pass any type of SQL query to inject the database from the Form field of web pages.

Network Level:-

- IPS is running in the IGNOU network, so whatever security is provided by Computer Division; application for the IPS also.
- There are some extra securities provided by ACIIL network team to secure IPS is such as IPS is running on Linux Operating system so Linux is already secured.
- A network firewall is installed between IGNOU network and ACIIL server.

Thus IPS is running in very secure environment.

8. Cost Estimation of the Project

IPS cost is estimated by COCOMO (Constructive Cost Model). The most fundamental calculation in the COCOMO model is the use of the Effort Equation to estimate the number of Person-Months required developing a project. Most of the other COCOMO results, including the estimates for Requirements and Maintenance, are derived from this quantity.

Project Name	IPS													
Total Size	504139													
Total Effort	2756.397774													
Overall	Schedule (%)	Schedule (Months)	Effort (%)	Effort	Staff									
Plans And Requirements	22.15%	10.091333	7.00%	192.947844	19.120155									
Product Design	27.08%	12.333666	17.00%	468.587622	37.992566									
Programming	43.69%	19.901333	54.77%	1509.634609	75.855954									
Integration and Test	29.23%	13.314999	28.23%	778.175544	58.44353									
EFFORT	Plans and Requirements	Product Design	Programming	Integration and Test	PERSONNEL	Plans and Requirements	Product Design	Programming	Integration and Test	EFFORT (%)	Plans and Requirements	Product Design	Programming	Integration and Test
Requirements Analysis	86.677566	58.573453	60.385384	19.454389	Requirements Analysis	8.589308	4.749071	3.034238	1.461088	Requirements Analysis	44.92%	12.50%	4.00%	2.50%
Product Design	33.840355	192.120925	120.770769	38.908777	Product Design	3.353408	15.576952	6.068476	2.922176	Product Design	17.54%	41.00%	8.00%	5.00%
Programming	10.761095	63.440214	852.943554	304.690031	Programming	1.06637	5.143662	42.858614	22.883218	Programming	5.58%	13.54%	56.50%	39.15%
Test Planning	7.792396	28.296142	83.612654	23.645659	Test Planning	0.772187	2.29422	4.201359	1.775866	Test Planning	4.04%	6.04%	5.54%	3.04%
Verification and Validation	14.54557	35.324956	128.901693	220.878853	Verification and Validation	1.441392	2.864108	6.477038	16.588725	Verification and Validation	7.54%	7.54%	8.54%	28.38%
Project Office	23.969517	46.496993	89.995326	54.171896	Project Office	2.375258	3.769925	4.522075	4.068487	Project Office	12.42%	9.92%	5.96%	6.96%
CM/QA	5.713953	11.533806	97.543499	61.953651	CM/QA	0.566224	0.935148	4.901355	4.652922	CM/QA	2.96%	2.46%	6.46%	7.96%
Manuals	9.647392	32.801134	75.48173	54.472288	Manuals	0.956008	2.65948	3.792798	4.091047	Manuals	5.00%	7.00%	5.00%	7.00%

Module Name	Query Sender Module														
Total Size	297780														
Total Effort	1628.122659														
Overall	Schedule (%)	Schedule (Months)	Effort (%)	Effort	Staff										
Plans And Requirements	22.15%	10.091333	7.00%	113.968586	11.29371										
Product Design	27.08%	12.333666	17.00%	276.780852	22.441085										
Programming	43.69%	19.901333	54.77%	891.696524	44.805869										
Integration and Test	29.23%	13.314999	28.23%	459.645283	34.520865										
EFFORT	Plans and Requirements	Product Design	Programming	Integration and Test	PERSONNEL	Plans and Requirements	Product Design	Programming	Integration and Test	EFFORT (%)	Plans and Requirements	Product Design	Programming	Integration and Test	
Requirements Analysis	51.197875	34.597606	35.667861	11.491132	Requirements Analysis	5.07345	2.805136	1.792235	0.863022	Requirements Analysis	44.92%	12.50%	4.00%	2.50%	
Product Design	19.988497	113.480149	71.335722	22.982264	Product Design	1.980759	9.200845	3.58447	1.726043	Product Design	17.54%	41.00%	8.00%	5.00%	
Programming	6.356261	37.472258	503.808536	179.971392	Programming	0.629873	3.038209	25.315316	13.51644	Programming	5.58%	13.54%	56.50%	39.15%	
Test Planning	4.602738	16.713694	49.387523	13.966791	Test Planning	0.456108	1.355128	2.481619	1.048952	Test Planning	4.04%	6.04%	5.54%	3.04%	
Verification and Validation	8.591638	20.865407	76.138418	130.466607	Verification and Validation	0.851388	1.691744	3.825795	9.798469	Verification and Validation	7.54%	7.54%	8.54%	28.38%	
Project Office	14.158085	27.464399	53.157578	31.997737	Project Office	1.402995	2.226783	2.671056	2.403135	Project Office	12.42%	9.92%	5.96%	6.96%	
CM/QA	3.375063	6.812678	57.61606	36.59419	CM/QA	0.334452	0.552364	2.895085	2.748343	CM/QA	2.96%	2.46%	6.46%	7.96%	
Manuals	5.698429	19.37466	44.584826	32.17517	Manuals	0.564686	1.570876	2.240293	2.416461	Manuals	5.00%	7.00%	5.00%	7.00%	

Module Name	Expert Module													
Total Size	75238													
Total Effort	411.36642													
Overall	Schedule (%)	Schedule (Months)	Effort (%)	Effort	Staff									
Plans And Requirements	22.15%	10.091333	7.00%	28.795649	2.853503									
Product Design	27.08%	12.333666	17.00%	69.932291	5.670033									
Programming	43.69%	19.901333	54.77%	225.298754	11.320787									
Integration and Test	29.23%	13.314999	28.23%	116.135374	8.722147									
EFFORT	Plans and Requirements	Product Design	Programming	Integration and Test	PERSONNEL	Plans and Requirements	Product Design	Programming	Integration and Test	EFFORT (%)	Plans and Requirements	Product Design	Programming	Integration and Test
Requirements Analysis	12.935811	8.741536	9.01195	2.903384	Requirements Analysis	1.281873	0.708754	0.452831	0.218054	Requirements Analysis	44.92%	12.50%	4.00%	2.50%
Product Design	5.050354	28.672239	18.0239	5.806769	Product Design	0.500465	2.324713	0.905663	0.436107	Product Design	17.54%	41.00%	8.00%	5.00%
Programming	1.605992	9.467855	127.293796	45.472119	Programming	0.159146	0.767643	6.396245	3.415105	Programming	5.58%	13.54%	56.50%	39.15%
Test Planning	1.162942	4.222933	12.478402	3.528892	Test Planning	0.115242	0.342391	0.627013	0.265031	Test Planning	4.04%	6.04%	5.54%	3.04%
Verification and Validation	2.170789	5.271917	19.237364	32.96409	Verification and Validation	0.215114	0.427441	0.966637	2.475711	Verification and Validation	7.54%	7.54%	8.54%	28.38%
Project Office	3.577225	6.939238	13.430955	8.084646	Project Office	0.354485	0.562626	0.674877	0.607183	Project Office	12.42%	9.92%	5.96%	6.96%
CM/QA	0.852754	1.721312	14.557449	9.245999	CM/QA	0.084504	0.139562	0.731481	0.694405	CM/QA	2.96%	2.46%	6.46%	7.96%
Manuals	1.439782	4.89526	11.264938	8.129476	Manuals	0.142675	0.396902	0.566039	0.61055	Manuals	5.00%	7.00%	5.00%	7.00%

Module Name	Department Module													
Total Size	46385													
Total Effort	253.611624													
Overall	Schedule (%)	Schedule (Months)	Effort (%)	Effort	Staff									
Plans And Requirements	22.15%	10.091333	7.00%	17.752814	1.759214									
Product Design	27.08%	12.333666	17.00%	43.113976	3.495633									
Programming	43.69%	19.901333	54.77%	138.898997	6.979382									
Integration and Test	29.23%	13.314999	28.23%	71.598652	5.377293									
EFFORT	Plans and Requirements	Product Design	Programming	Integration and Test	PERSONNEL	Plans and Requirements	Product Design	Programming	Integration and Test	EFFORT (%)	Plans and Requirements	Product Design	Programming	Integration and Test
Requirements Analysis	7.97506	5.389247	5.55596	1.789966	Requirements Analysis	0.790288	0.436954	0.279175	0.134432	Requirements Analysis	44.92%	12.50%	4.00%	2.50%
Product Design	3.113595	17.67673	11.11192	3.579933	Product Design	0.308542	1.43321	0.558351	0.268865	Product Design	17.54%	41.00%	8.00%	5.00%
Programming	0.990111	5.83703	78.477933	28.034028	Programming	0.098115	0.47326	3.943351	2.105447	Programming	5.58%	13.54%	56.50%	39.15%
Test Planning	0.716966	2.603481	7.693063	2.175598	Test Planning	0.071048	0.211087	0.38656	0.163395	Test Planning	4.04%	6.04%	5.54%	3.04%
Verification and Validation	1.338314	3.250191	11.860033	20.3227	Verification and Validation	0.13262	0.263522	0.595942	1.526301	Verification and Validation	7.54%	7.54%	8.54%	28.38%
Project Office	2.205396	4.278112	8.280322	4.984267	Project Office	0.218544	0.346865	0.416069	0.374335	Project Office	12.42%	9.92%	5.96%	6.96%
CM/QA	0.525731	1.061206	8.974817	5.700254	CM/QA	0.052097	0.086041	0.450966	0.428108	CM/QA	2.96%	2.46%	6.46%	7.96%
Manuals	0.887641	3.017978	6.94495	5.011906	Manuals	0.087961	0.244694	0.348969	0.376411	Manuals	5.00%	7.00%	5.00%	7.00%

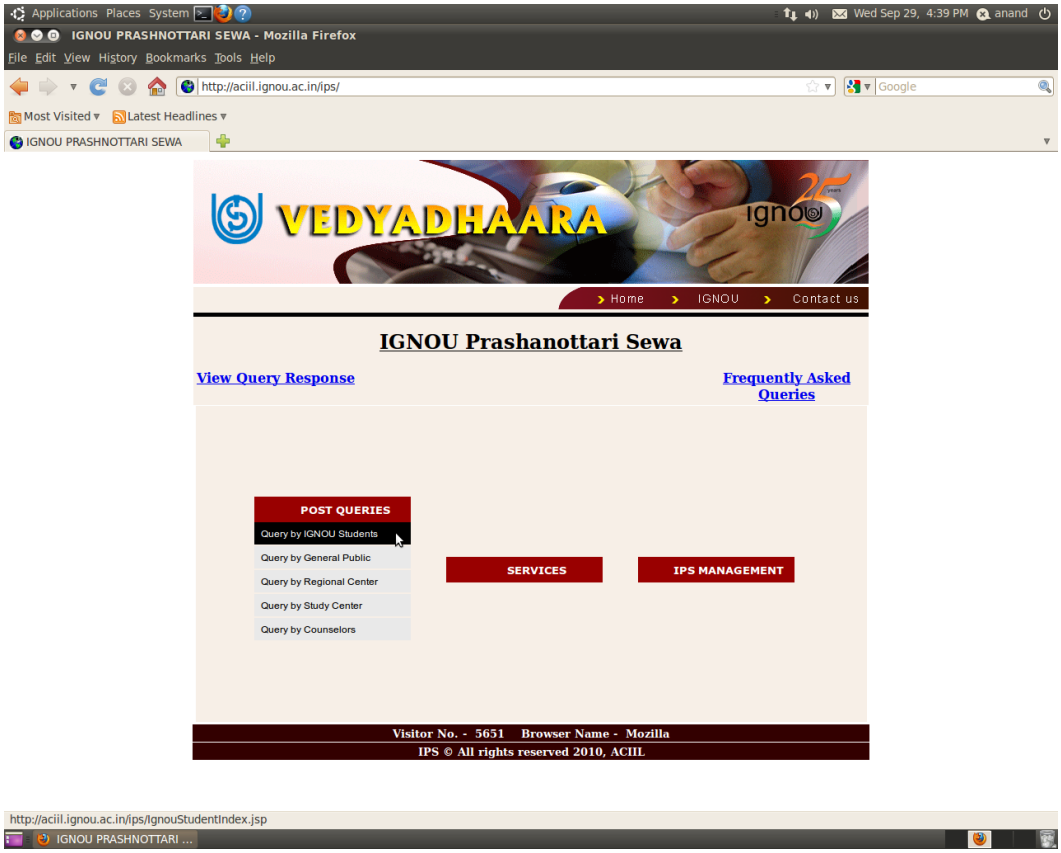
Module Name	Super Admin Module													
Total Size	31773													
Total Effort	173.719999													
Overall	Schedule (%)	Schedule (Months)	Effort (%)	Effort	Staff									
Plans And Requirements	22.15%	10.091333	7.00%	12.1604	1.205034									
Product Design	27.08%	12.333666	17.00%	29.5324	2.394454									
Programming	43.69%	19.901333	54.77%	95.143642	4.780767									
Integration and Test	29.23%	13.314999	28.23%	49.043957	3.683362									
EFFORT	Plans and Requirements	Product Design	Programming	Integration and Test	PERSONNEL	Plans and Requirements	Product Design	Programming	Integration and Test	EFFORT (%)	Plans and Requirements	Product Design	Programming	Integration and Test
Requirements Analysis	5.462792	3.69155	3.805746	1.226099	Requirements Analysis	0.541335	0.299307	0.191231	0.092084	Requirements Analysis	44.92%	12.50%	4.00%	2.50%
Product Design	2.132764	12.108284	7.611491	2.452198	Product Design	0.211346	0.981726	0.382461	0.184168	Product Design	17.54%	41.00%	8.00%	5.00%
Programming	0.67821	3.998274	53.756158	19.202871	Programming	0.067207	0.324176	2.701133	1.442198	Programming	5.58%	13.54%	56.50%	39.15%
Test Planning	0.49111	1.783344	5.269628	1.490251	Test Planning	0.048667	0.144592	0.264788	0.111923	Test Planning	4.04%	6.04%	5.54%	3.04%
Verification and Validation	0.916724	2.22633	8.123937	13.920732	Verification and Validation	0.090843	0.180508	0.408211	1.045493	Verification and Validation	7.54%	7.54%	8.54%	28.38%
Project Office	1.510662	2.93044	5.671891	3.414145	Project Office	0.149699	0.237597	0.285001	0.256413	Project Office	12.42%	9.92%	5.96%	6.96%
CM/QA	0.360118	0.72691	6.147609	3.904585	CM/QA	0.035686	0.058937	0.308904	0.293247	CM/QA	2.96%	2.46%	6.46%	7.96%
Manuals	0.60802	2.067268	4.757182	3.433077	Manuals	0.060252	0.167612	0.239038	0.257835	Manuals	5.00%	7.00%	5.00%	7.00%

Module Name	File Upload and Download Modul													
Total Size	40220													
Total Effort	219.904269													
Overall	Schedule (%)	Schedule (Months)	Effort (%)	Effort	Staff									
Plans And Requirements	22.15%	10.091333	7.00%	15.393299	1.525398									
Product Design	27.08%	12.333666	17.00%	37.383726	3.031031									
Programming	43.69%	19.901333	54.77%	120.438022	6.051756									
Integration and Test	29.23%	13.314999	28.23%	62.082522	4.662601									
EFFORT	Plans and Requirements	Product Design	Programming	Integration and Test	PERSONNEL	Plans and Requirements	Product Design	Programming	Integration and Test	EFFORT (%)	Plans and Requirements	Product Design	Programming	Integration and Test
Requirements Analysis	6.9151	4.672966	4.817521	1.552063	Requirements Analysis	0.685251	0.378879	0.24207	0.116565	Requirements Analysis	44.92%	12.50%	4.00%	2.50%
Product Design	2.699769	15.327328	9.635042	3.104126	Product Design	0.267533	1.242723	0.484141	0.23313	Product Design	17.54%	41.00%	8.00%	5.00%
Programming	0.858516	5.061234	68.047482	24.308044	Programming	0.085075	0.410359	3.419242	1.825614	Programming	5.58%	13.54%	56.50%	39.15%
Test Planning	0.621674	2.257454	6.670583	1.886441	Test Planning	0.061605	0.183032	0.335183	0.141678	Test Planning	4.04%	6.04%	5.54%	3.04%
Verification and Validation	1.16044	2.81821	10.283723	17.621623	Verification and Validation	0.114994	0.228497	0.516735	1.323442	Verification and Validation	7.54%	7.54%	8.54%	28.38%
Project Office	1.912278	3.709511	7.17979	4.321811	Project Office	0.189497	0.300763	0.360769	0.324582	Project Office	12.42%	9.92%	5.96%	6.96%
CM/QA	0.455857	0.920162	7.78198	4.942637	CM/QA	0.045173	0.074606	0.391028	0.371208	CM/QA	2.96%	2.46%	6.46%	7.96%
Manuals	0.769665	2.616861	6.021901	4.345777	Manuals	0.07627	0.212172	0.302588	0.326382	Manuals	5.00%	7.00%	5.00%	7.00%

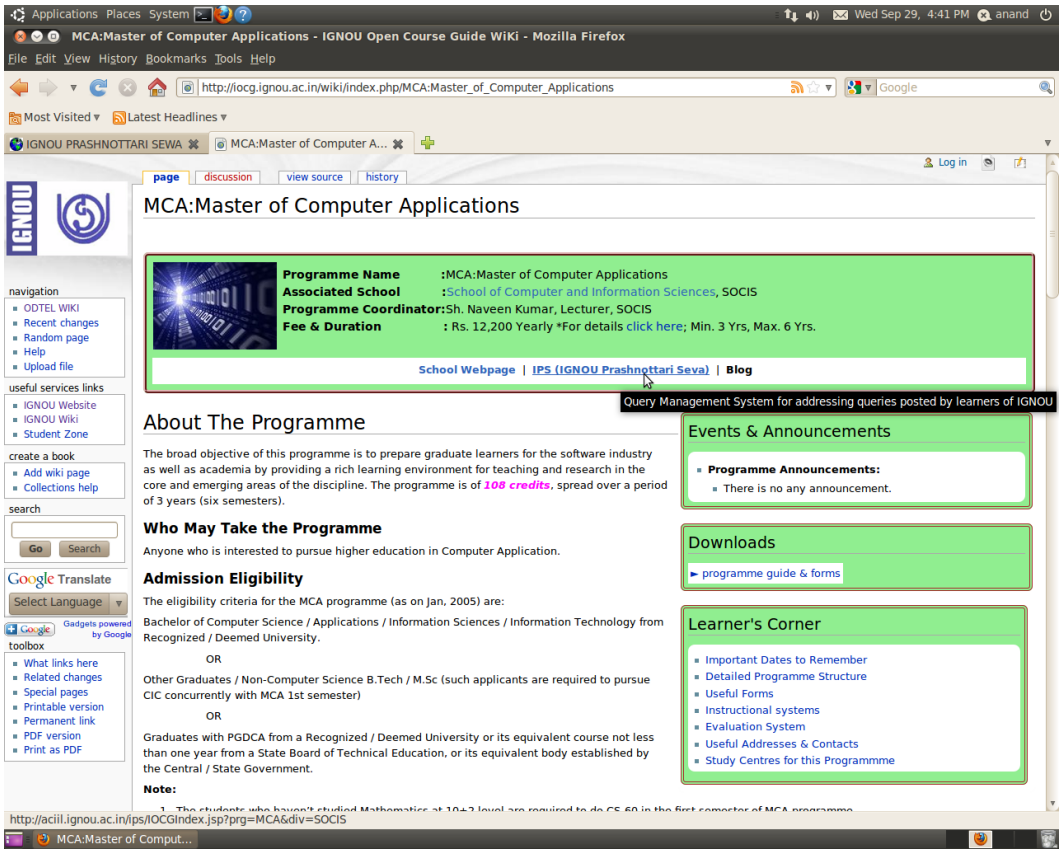
Module Name	Circular Module													
Total Size	12743													
Total Effort	69.672802													
Overall	Schedule (%)	Schedule (Months)	Effort (%)	Effort	Staff									
Plans And Requirements	22.15%	10.091333	7.00%	4.877096	0.483296									
Product Design	27.08%	12.333666	17.00%	11.844376	0.960329									
Programming	43.69%	19.901333	54.77%	38.15867	1.917393									
Integration and Test	29.23%	13.314999	28.23%	19.669756	1.477263									
EFFORT	Plans and Requirements	Product Design	Programming	Integration and Test	PERSONNEL	Plans and Requirements	Product Design	Programming	Integration and Test	EFFORT (%)	Plans and Requirements	Product Design	Programming	Integration and Test
Requirements Analysis	2.190928	1.480547	1.526347	0.491744	Requirements Analysis	0.21711	0.120041	0.076696	0.036932	Requirements Analysis	44.92%	12.50%	4.00%	2.50%
Product Design	0.855374	4.856194	3.052694	0.983488	Product Design	0.084763	0.393735	0.153391	0.073863	Product Design	17.54%	41.00%	8.00%	5.00%
Programming	0.272006	1.603563	21.559649	7.701576	Programming	0.026954	0.130015	1.083327	0.578414	Programming	5.58%	13.54%	56.50%	39.15%
Test Planning	0.196967	0.715235	2.113457	0.597686	Test Planning	0.019518	0.05799	0.106197	0.044888	Test Planning	4.04%	6.04%	5.54%	3.04%
Verification and Validation	0.367665	0.8929	3.258217	5.583102	Verification and Validation	0.036434	0.072395	0.163719	0.419309	Verification and Validation	7.54%	7.54%	8.54%	28.38%
Project Office	0.605872	1.175293	2.27479	1.36929	Project Office	0.060039	0.095291	0.114303	0.102838	Project Office	12.42%	9.92%	5.96%	6.96%
CM/QA	0.14443	0.291537	2.465584	1.565988	CM/QA	0.014312	0.023638	0.12389	0.117611	CM/QA	2.96%	2.46%	6.46%	7.96%
Manuals	0.243855	0.829106	1.907934	1.376883	Manuals	0.024165	0.067223	0.09587	0.103408	Manuals	5.00%	7.00%	5.00%	7.00%

9. Screen Layouts of IPS

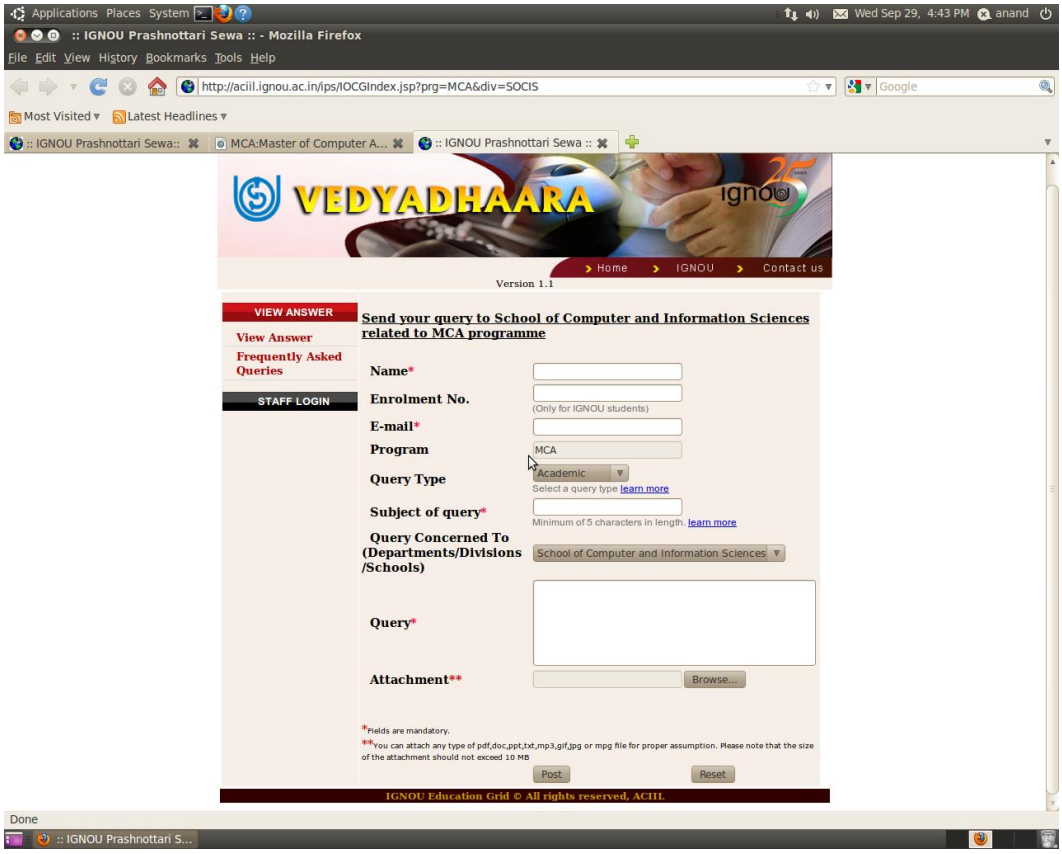
IPS Home Page



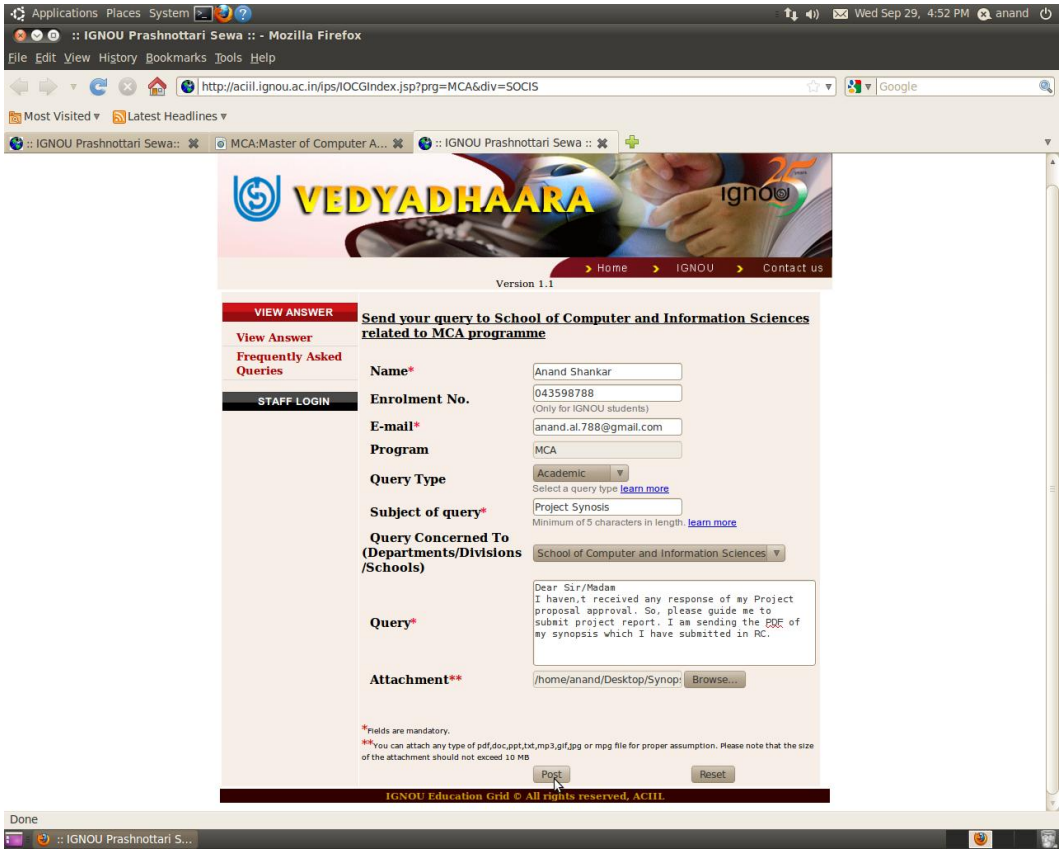
IPS navigation from Programme page of ODTEL wiki



IPS interface to the send query

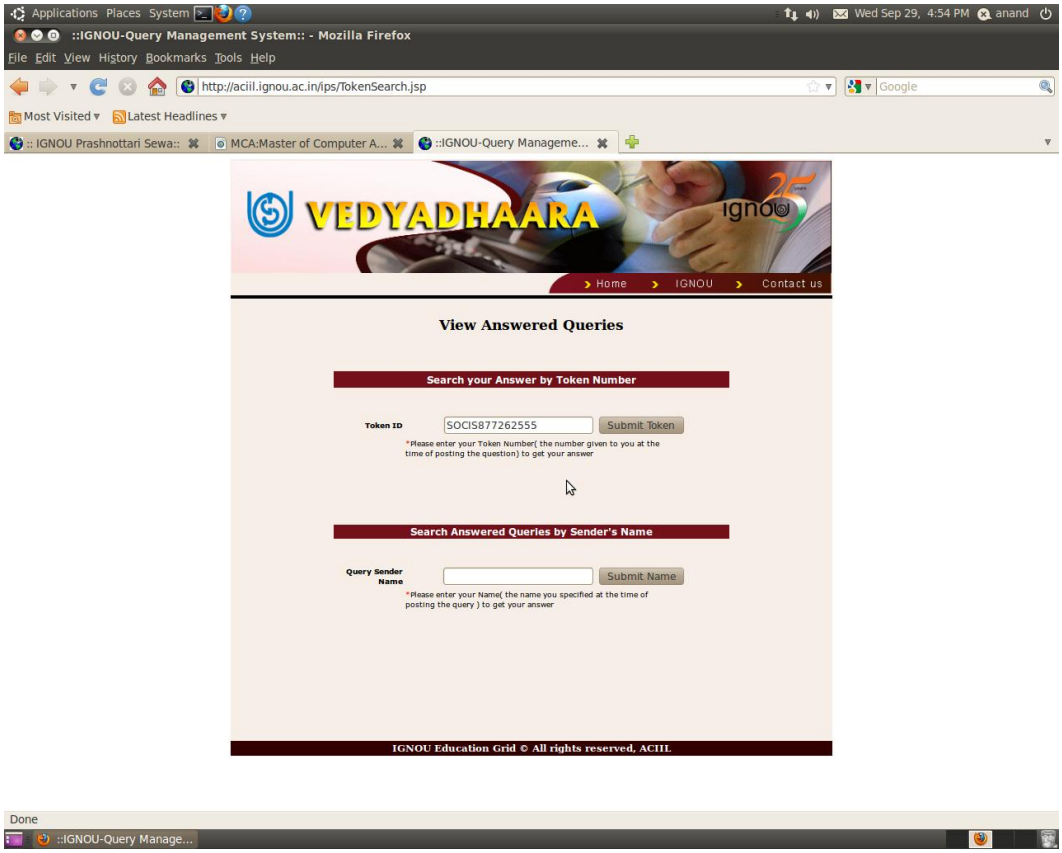


IPS Sample for sending a query

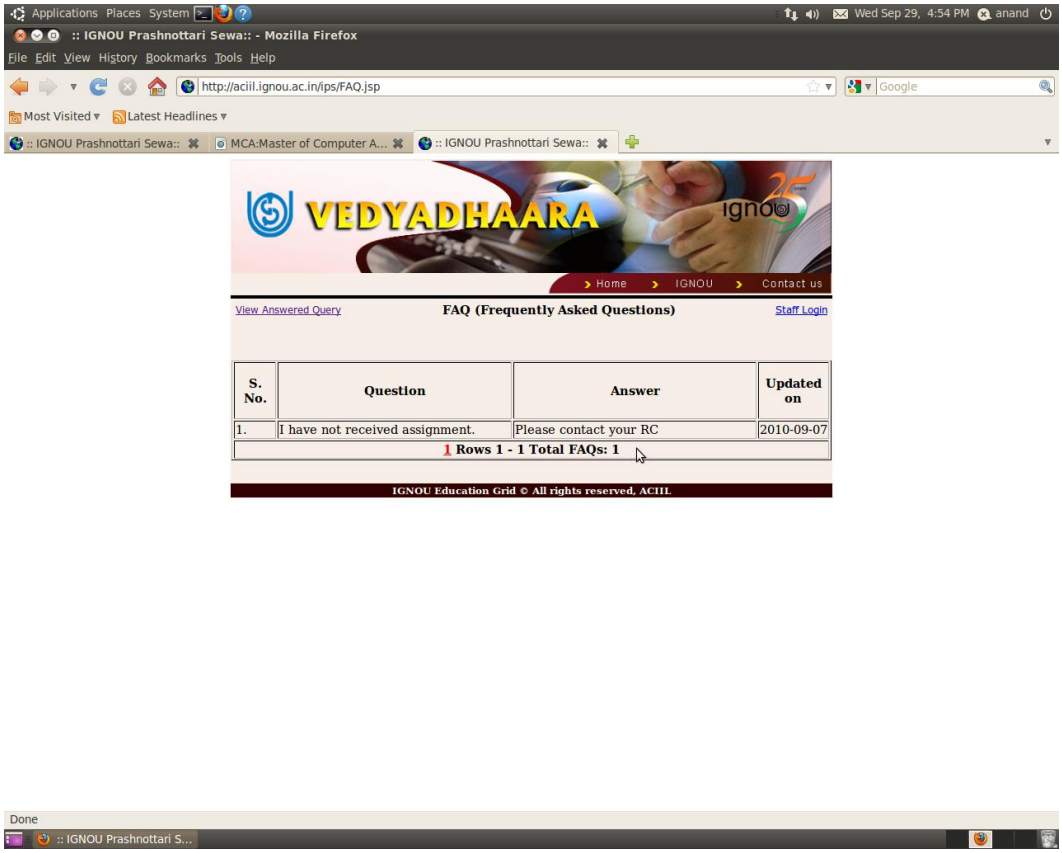


IPS acknowledgement after sending the query

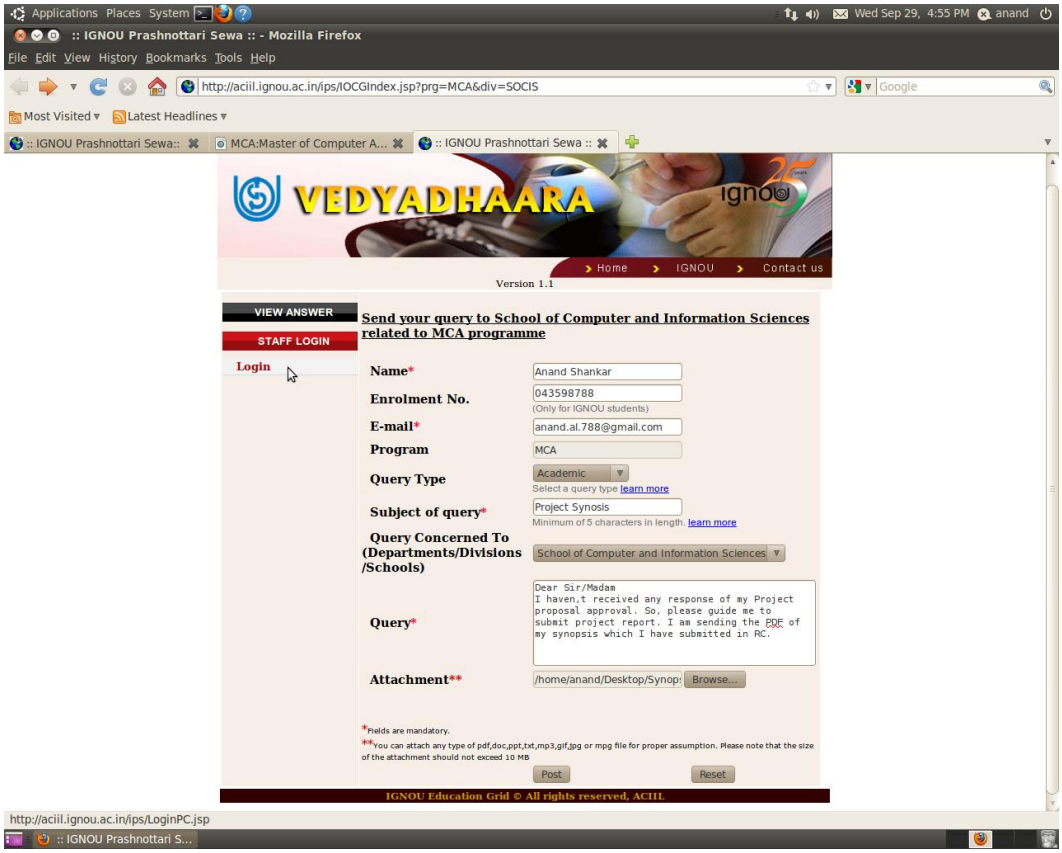
IPS interface for tracking status of the query



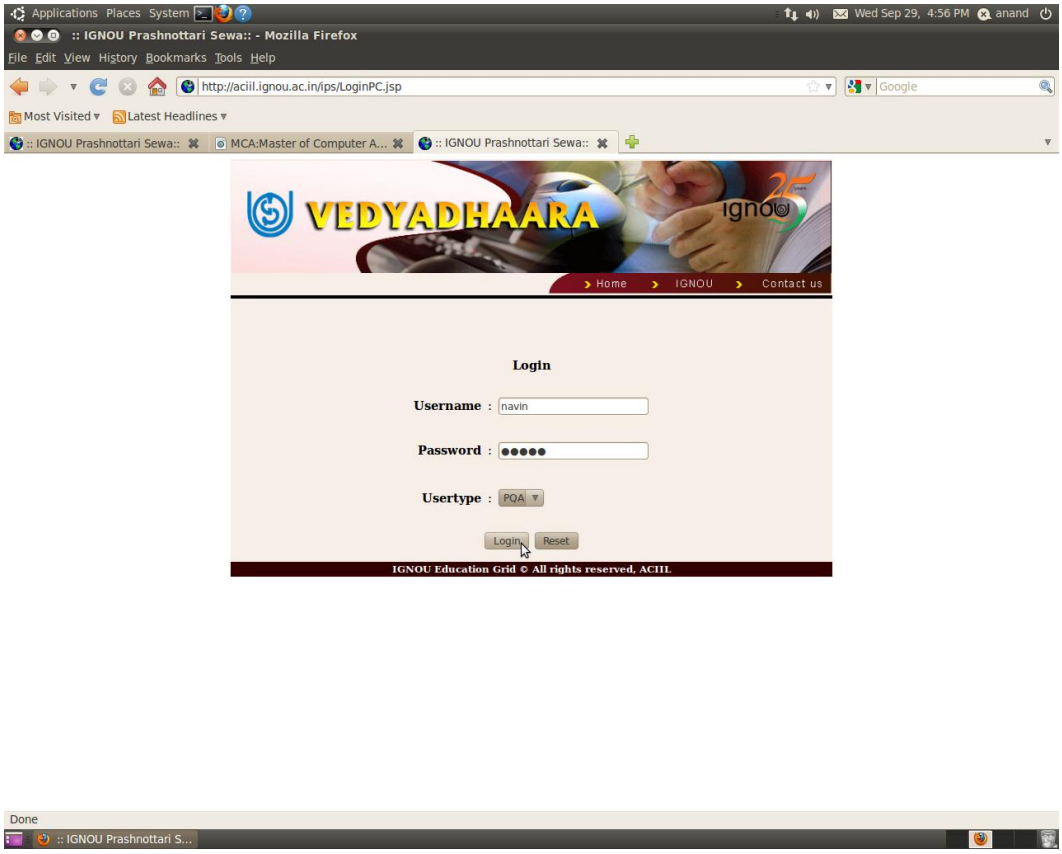
IPS FAQ


Applications Places System
Wed Sep 29, 4:54 PM anand
:: IGNOU Prashnottari Sewa:: - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://aciil.ignou.ac.in/ips/FAQ.jsp
Most Visited
Latest Headlines
:: IGNOU Prashnottari Sewa::
MCA:Master of Computer A...
:: IGNOU Prashnottari Sewa::
VEDYADHAARA
ignou
Home
IGNOU
Contact us
View Answered Query
FAQ (Frequently Asked Questions)
Staff Login
S. No.
Question
Answer
Updated on
1.
I have not received assignment.
Please contact your RC
2010-09-07
1 Rows 1 - 1 Total FAQs: 1
IGNOU Education Grid © All rights reserved, ACIIL
Done
:: IGNOU Prashnottari S...

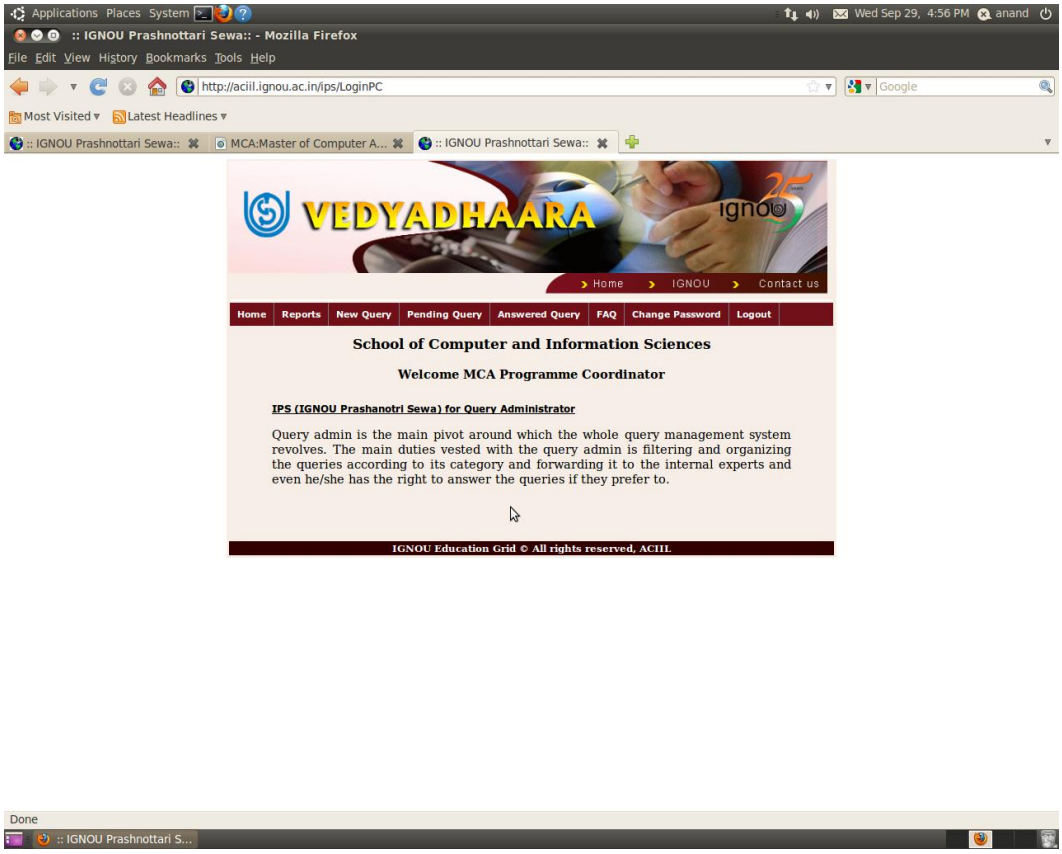
IPS Login Link for IGNOU staff



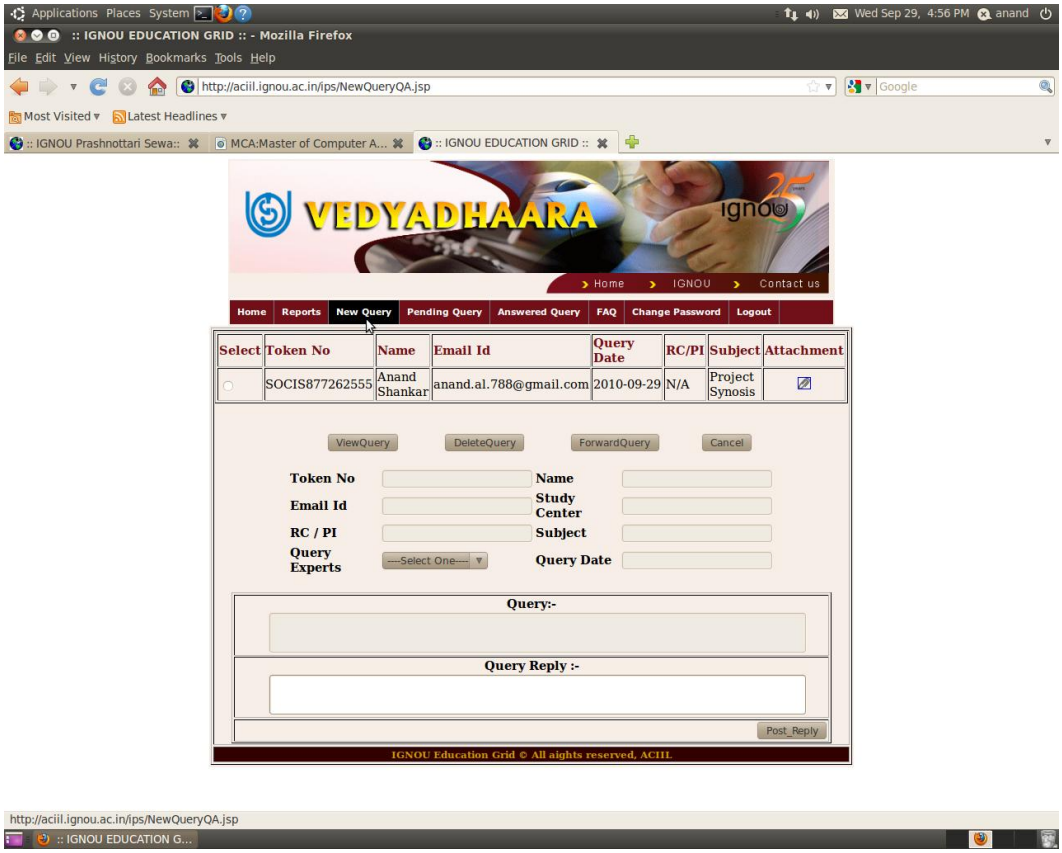
IPS Login Page for IGNOU staff



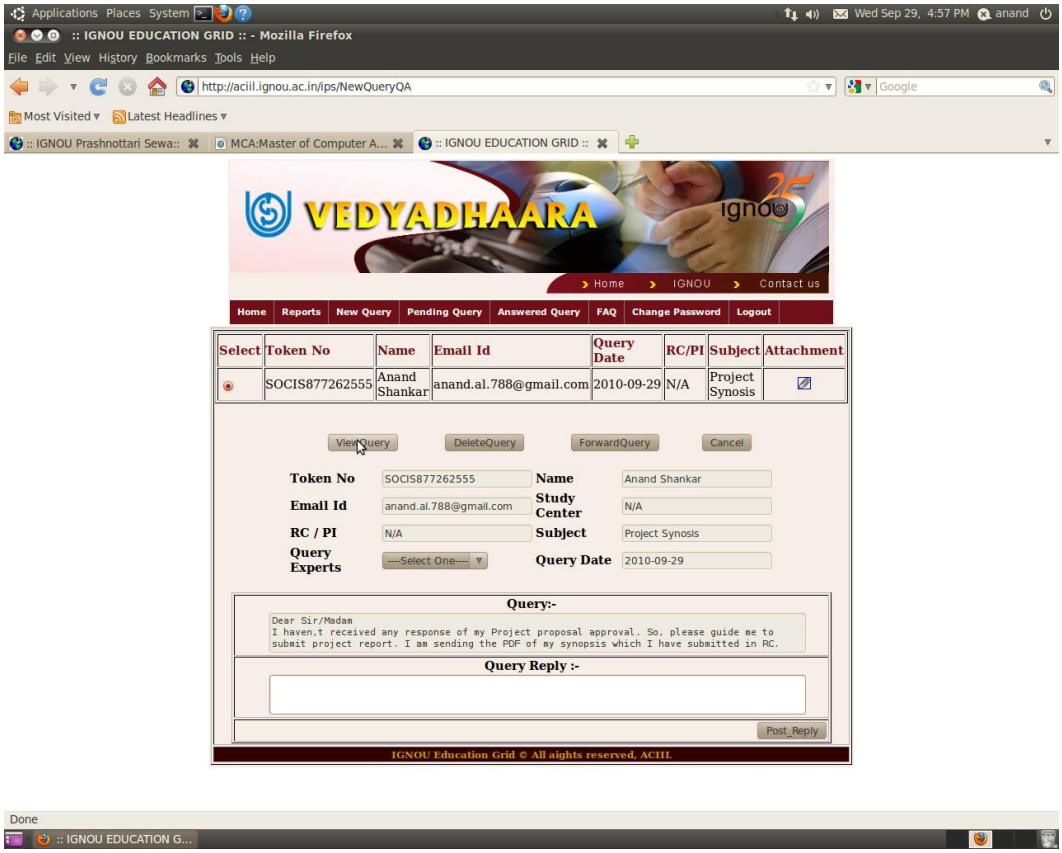
Dashboard for Programme Coordinator



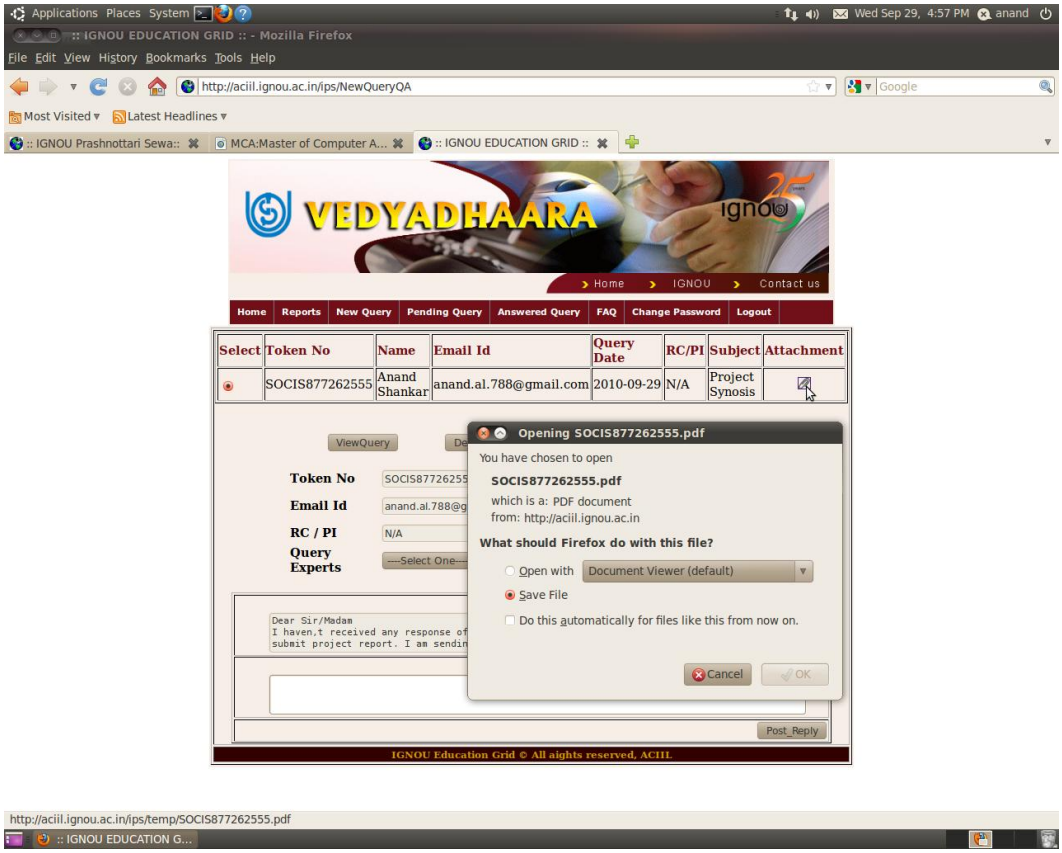
Viewing New Queries



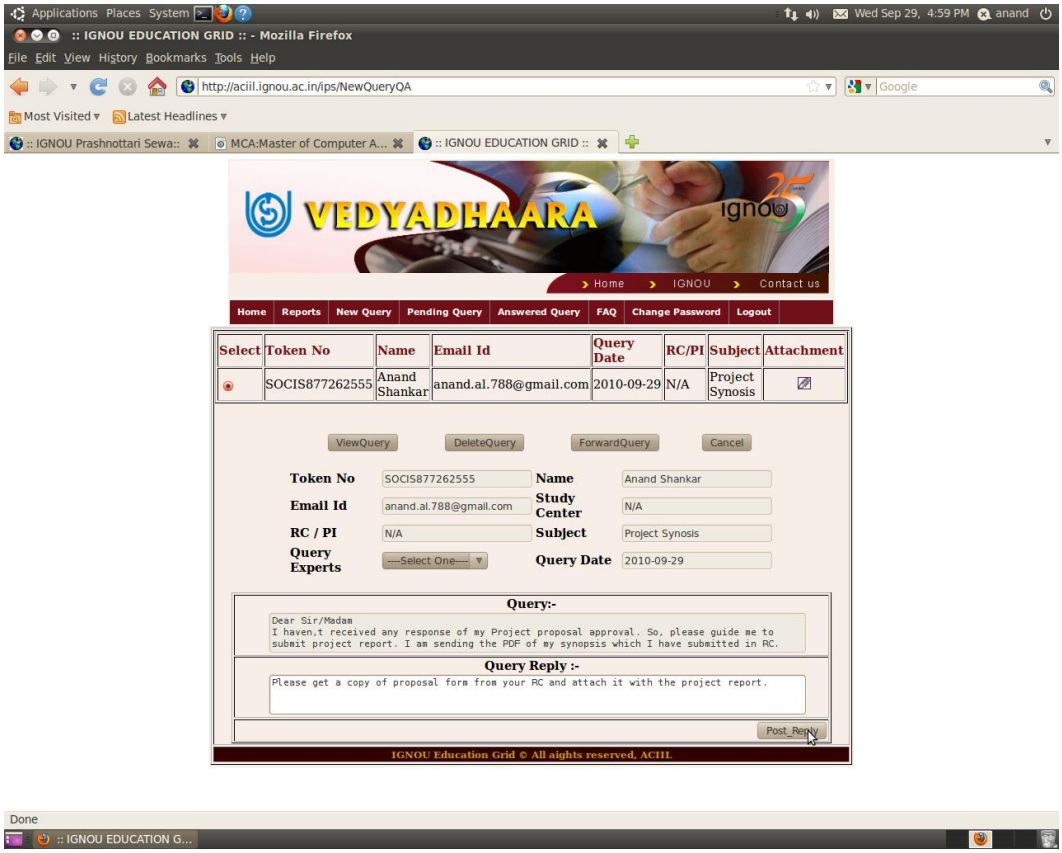
Selecting details for a query



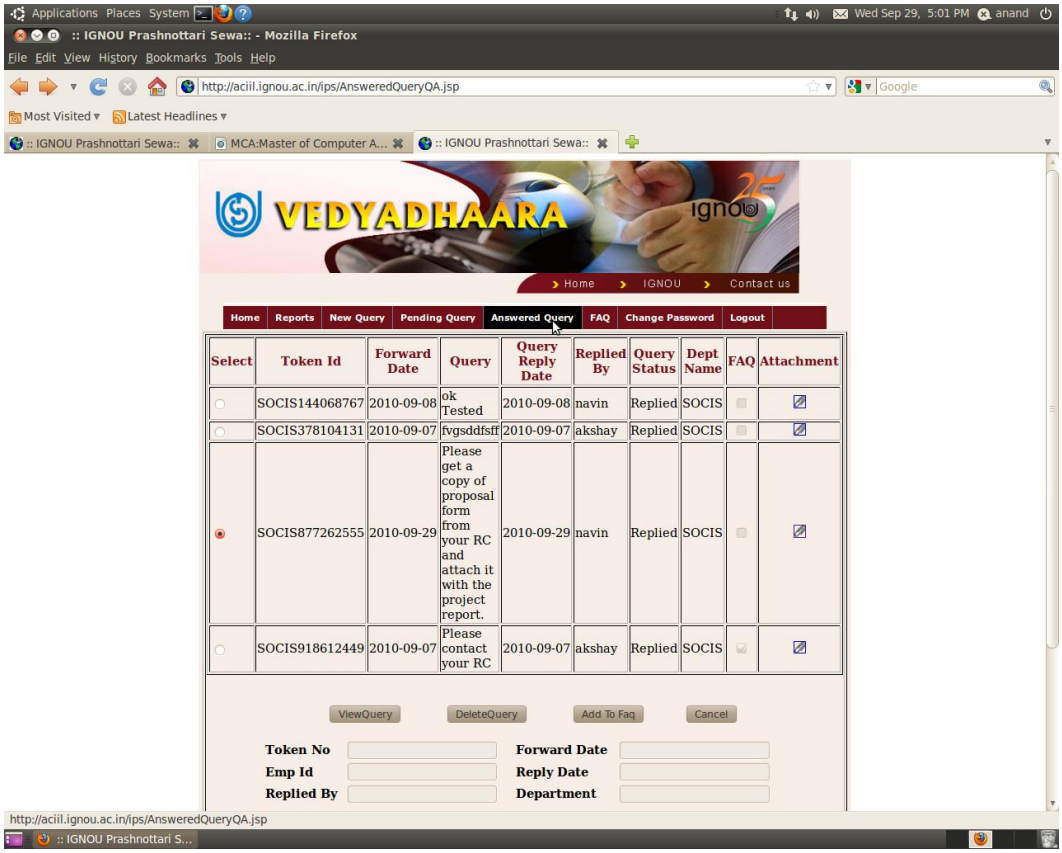
Downloading attachment



Replying a query

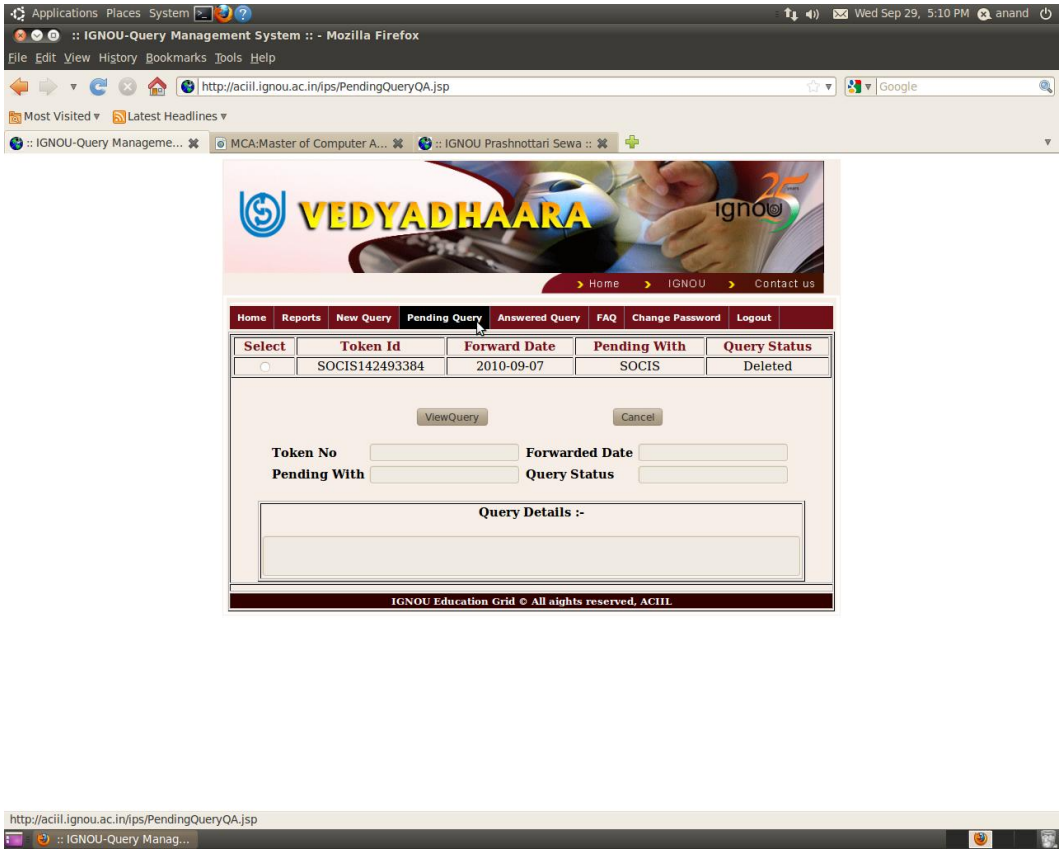


Viewing Answered Queries



Select	Token Id	Forward Date	Query	Query Reply Date	Replied By	Query Status	Dept Name	FAQ	Attachment
	SOCIS144068767	2010-09-08	ok Tested	2010-09-08	navin	Replied	SOCIS		
	SOCIS378104131	2010-09-07	fvgsddfsff	2010-09-07	akshay	Replied	SOCIS		
	SOCIS877262555	2010-09-29	Please get a copy of proposal form from your RC and attach it with the project report.	2010-09-29	navin	Replied	SOCIS		
	SOCIS918612449	2010-09-07	Please contact your RC	2010-09-07	akshay	Replied	SOCIS		

Viewing Pending Queries



10. Future Scope and Further Enhancement

- IPS interface for all the Schools, Divisions, Centres, Units, Cells, RCs, SCs, PIs, Student and General public.
- Enhancing the IPS with following features
    - Queries to be managed through SMS also
    - Person to Person chat facility
    - Call centre based IPS.
- Training for IGNOU staff on Groupware and discussion forum.
- Development of Online Library and multimedia publishing
- Developing a support system for Regional Centers.
- Improving the Online Student Registration System for IGNOU.
- Development of IGNOU Education Grid Portal.
- Integrate IPS with ODTEL framework of IGNOU.

11. References

Book Title	Author	Publication
Swing	Matthew Robinson, Povel Voroiev	Dreamtech
Core Java	Cay S. Horstmann, Gary Cornell	Pearson
Head First Java	Kathy Sierra, Bert Bates	O'reilly
SQL & PL/SQL using Oracle	Ivan Bayross	BPB
Java Server Programming	Kogent Solutions Inc.	Dreamtech
Oracle SQL * Security	Theriault	SPD
Oracle Utilities Pocket Reference	Mishra	SPD
NetBeans: The Definitive Guide	Boudreau	SPD
Java Database Object	Jordan	SPD

www.google.com

www.sun.com

www.mysql.com

www.wikipedea.com

12. Glossary

Vedyadhara:- Vedyadhara is an Open Online Learning Guide which was introduced by Prof. K. R. Srivathsan, PVC, IGNOU.

FAQ:- FAQ stands for Frequently Asked Question. In FAQ that question and answers will be added which is asked by students mostly.

Alert:- Here Alert means; some automatic tools which remind again and again for something.

Academic:- IGNOU has Schools for handling Programme and Course related materials and contents. Subject expert faculties are there to handle these things. Such as SOCIS, SOH, etc.

Administrative:- All the Registration, Examination or Marks related things are handled by some divisions which comes under Administrative things such as SED , SRD, etc.

P-P:- P-P means Pear to Pear.

Archives:- Archives means storing records somewhere like database or in file so that in future It can be retrieved as needed.

Super Admin:- Here Super Admin means a person how has a special Login ID and Password to login in the IPS and can see the status of all the users.

General:- Who is not the student and staff of IGNOU.

Token id:- Token ID is a king of reference number by which anyone can track their query.

C11_2000:- This is a tool which is based on COCOMO.

Dia:- This ios a tool which is used to design a system or architecture..

Department:- Here Department means various Schools, Divisions, Centres, Units and Cells of IGNOU.

***** End of the Document *****