1. NAME: PRAJAPATI CHETAN RAMESHBHAI
REG NO.: 16CO132
MOBILE NO.: 9611591403
EMAIL ID: crprajapati111@gmail.com

2. NAME: BOBBY GURUDAS PATIL
REG NO.: 16CO130
MOBILE NO.: 7248900445
EMAIL ID: bobbypt05@gmail.com

TITLE OF THE PROBLEM: CAMPUS RECRUITMENT SYSTEM

BRIEF NARRATION OF THE PROBLEM:

This project is to facilitate students in college, company to register and communicate with Placement Office. The users can access easily to this and the data can be retrieved easily in no time. In the main page there are options for a new register, a registered student to directly login using username and password, submit resume. In the student registration form, we can give personal details, educational qualifications, and professional skills and upload resume. The job details of the placed students will be provided by the administrator. The administrator plays an important role in our project. They provide approval of student registration and updating. In this project we create a search engine for administrator, who can search everything about the student and company.

BRIEF WORK ENVISAGED(IN THE SW ENGG. LAB) FOR PROPOSED PROBLEM:

Develop a domain model ,use of context diagram (in multiple levels), DFDs and ER diagrams in the traditional approach.Follow requirements engineering practices that can be selected with suitable justification. (i.e., elicitation, analysis,specification and verification..) . Prepare the SRS and FRS in a standard form proposed in the process model. Development of High-level design, detailed design, logical design, physical design and architecture of the proposed / to-be-implemented system; using suitable tools and implementation or prototyping and then demonstrating. Perform testing on software.

Part1 - SoftwareCrisis

Q1. lmagine that your software has failed at the customer's site. List out the possible reasons for failure and write possible solutions.

#1 REASON

INCOMPATIBILITY OF SOFTWARE WITH USER'S SYSTEM

- Software Incompatibility means that the application software could not be compiled on the user's system.
- If there is different version of operating system at user's site during installation it leads to software incompatibility.

#1 SOLUTION

- We do not need to modify whole software but we need to install & update new software libraries to solve issue of software incompatibility.

#2 REASON

LOW DISK SPACE ISSUE

- If user's system does not have enough space to create the databases to store the information of different number of students getting recruited & which companies recruited how many students. Suppose there is 32GB memory than now around 128GB was required.

#2 SOLUTION

- So basically what we need is to expand the memory of a device to store more database.
- So the solution of this problem is to purchase additional memory from the market to support additional databases.

#3 REASON

CHANGE IN THE USER REQUIREMENTS

- Suppose software is built for only campus recruitment system of NITK only and now user wants the campus recruitment system for additional colleges like MIT,IISc,IITB,IIITH,etc.

#3 SOLUTION

- So basically to have new requirements in software (like campus recruitment software as mentioned above) either additional programs need to be developed or the existing logic in the programs have to be revised to meet new requirements
- This essentially means revisiting all the phases i.e. Requirements, design, construction and unit testing and system Testing.

#4 REASON

NON FUNCTIONAL REQUIREMENTS FAILURE

- When a user tests the software and at a time lots of user starts using same software and software failed to scale-to desired number of users or the response times became unacceptable

#4 SOLUTION

- So basically we can use different optimised algorithms to store database like different indexing techniques.
- In some cases even additional hardware may need to be procured to provide adequate load balancing.
- Lastly we need to retest to ensure that the changes made are correct both from a functional and non-fucntional perspective.

#5 REASON

LACK OF QUALITY TESTING

- In most projects, the importance is given to coding isn't given to testing. Honestly speaking testing calls for a greater integrity and role in entire software development lifecycle. Casual testing, testing under non-real time environments contribute to testing failures.

#5 SOLUTION

- So basically we need to test our software under live production environments.

Q2.Imagine that your software is not delivered to the customer on the given time and customer is willing to cancel the project. List out the possible reasons for delay in development and write some possible eflorts that you would make to convince the customer in order to retain the project.

REASONS FOR DELAY

- There can be many reasons why a project got delayed during development and it is suggested that a detailed root cause analysis should be performed to ascertain the exact reason

a) **Because of Team developing the software**

- Clients are always eager to have their projects rolled-out on time and even before the stipulated time at throw away prices. In most cases, this keenness of the client leads to developers agreeing to a rather shorter or unrealistic and non-negotiable time frame for the software delivery at meager rates. As a result, programmers are not able to deliver the project on time.
- Design team misunderstood the requirement and thus made incorrect design. And thus more time was required to correct it.
- At the end of System Design, it was realized that the size and complexity was far more than what was estimated earlier.
- Suppose during Construction Phase, member of the project team left or felt sick and could'nt work on project for a few weeks or some new meber arrived.

- During System and Integration Testing a large number of bugs were found – this meant that code had to be revisited and corrected.

b) **Because of Lack of Communication**

- Another key aspect is the failure to set up effective communication channels and participative environment. Due to this, ideas or process flows get adrift and leads to lack of previews and interactions between the active project promoters and developers. At times standardized assumptions may lead to misunderstanding as standards may vary leading to business software failure.

c) **Due to Unfocused Executive sponsors**

- Inactive leadership through non-effective sponsors is why IT projects fails. They are the people who keep the process ignited and are primarily responsible for the success or failure of project. So in most cases lack of time spent by these executive sponsors or the complete lack of it, leads to project failures.

d) **Due to Lack of Periodic Assessment**

- Lack of client induced and developer mandated assessments, and failure in smartly establishing milestone points leads to improper assessments, which leads to irrepairable or catastrophic failures.

Efforts to convince and retain the project:

a)

- **If the delay is due to the company that developed software – then the followng options can be considered:**
- Articulate to Business sponsor with data the reasons for delay and provide at least 2-3 revised timeline plan options for making the software live – also offer a plan to ramp-up the resources to speed-up development and testing .
- As an option, we can use divide and conquer approach – the software can be deployed with reduced size and scope i.e. Salvage the efforts put in so far by leveraging software developed and ready to go-live.

b)

- **If the delay is due to lack of communication then folowing efforts can be made:**
- Team members should communicate with each other frequently and should be co-operative and should co-ordinate with each other.

- There should be a proper communication between user and team and even sponsors.

c)

- **If the delay is due to unfocused executive sponsors then following efforts can be made:**
- Sponsors can be made focussed by challenging them in their field of interest and can be made to concentrate on a particular project.

-> In all the scenarios, an attempt should be made to salvage the project and protect the investments made for future return of investments.

-> In the process

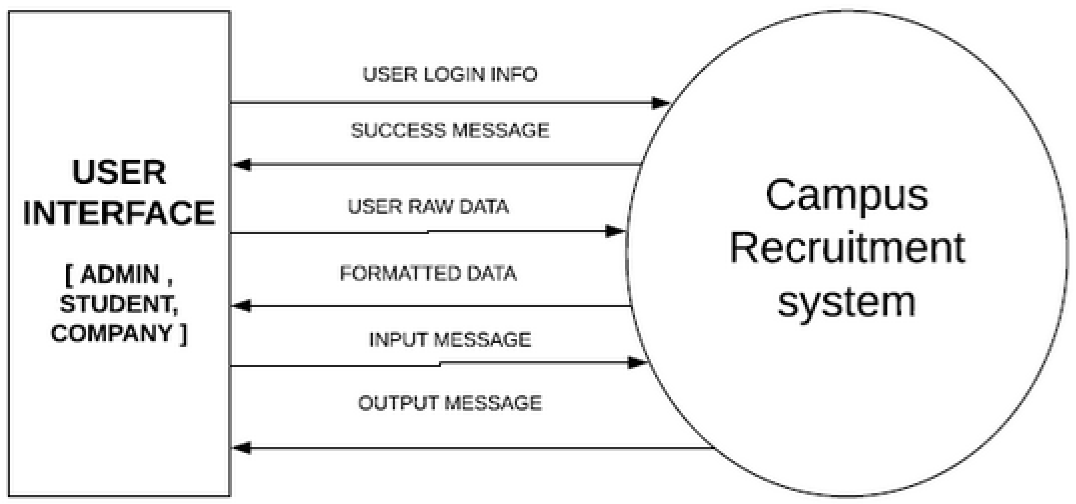
Reduced Scope (Software size, complexity)
Going live incrementally
An interim solution

can be considered.

Part2 - Context Diagram, Data Flow Diagram and ER-Diagram

Q-1. Draw a context diagram for your problem statement.

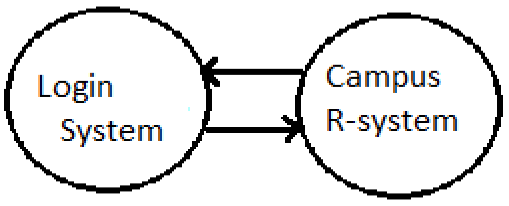
Ans.



Q-2. Imagine that you have recruited a fresh graduate to draw Data flow diagram for your problem statement. List out minimum five possible errors he / she may make while drawing the data flow diagram and explain it with a clear pictorial representations.

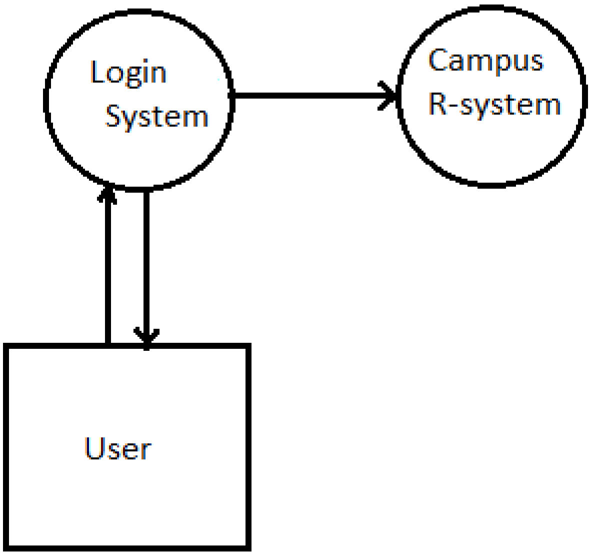
Ans.

- A fresh graduate may commit the mistake of drawing more than one bubble in the context diagram. But a context diagram should depict the system as a single bubble. Example is shown below.



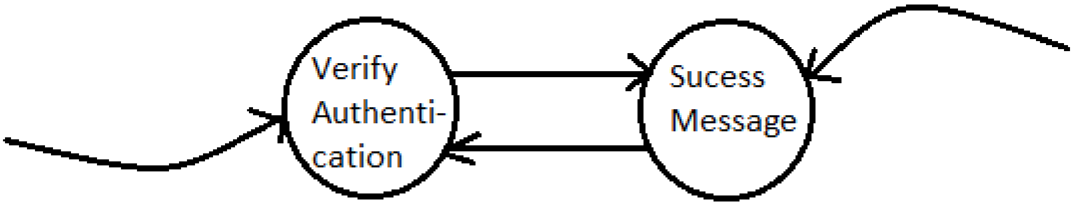
Wrong

- A fresh graduate may show external entities at all levels of DFDs. All external entities interacting with the system should be represented only in the context diagram. The external entities should not appear at other levels of the DFD. Example is shown below.



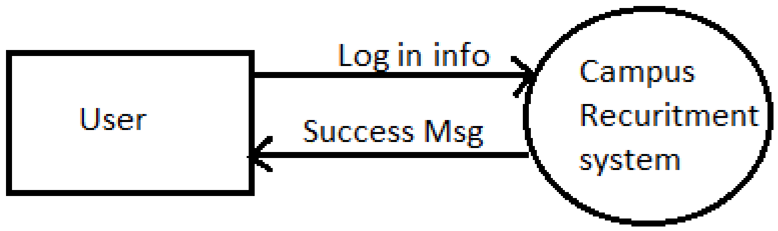
Wrong

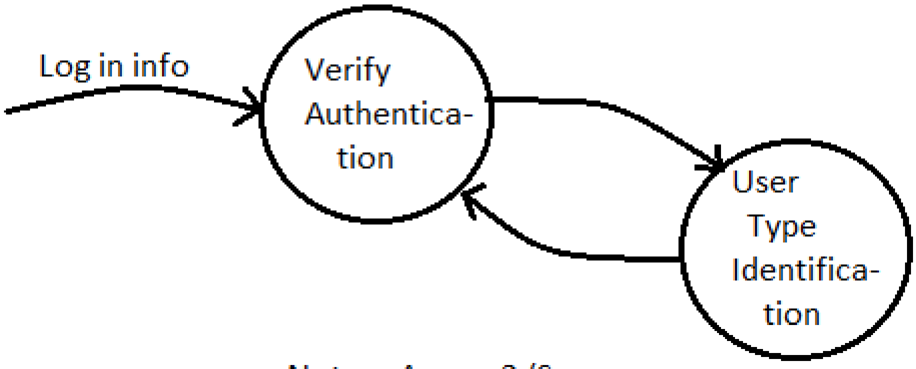
- A fresh graduate may include few or more bubbles at each level of DFD. Only 3 to 7 bubbles per level of DFD should be allowed, i.e. each bubble should be decomposed to between 3 and 7 bubbles. Example is shown below.



Wrong : It should have bubbles between 3 to 7 .

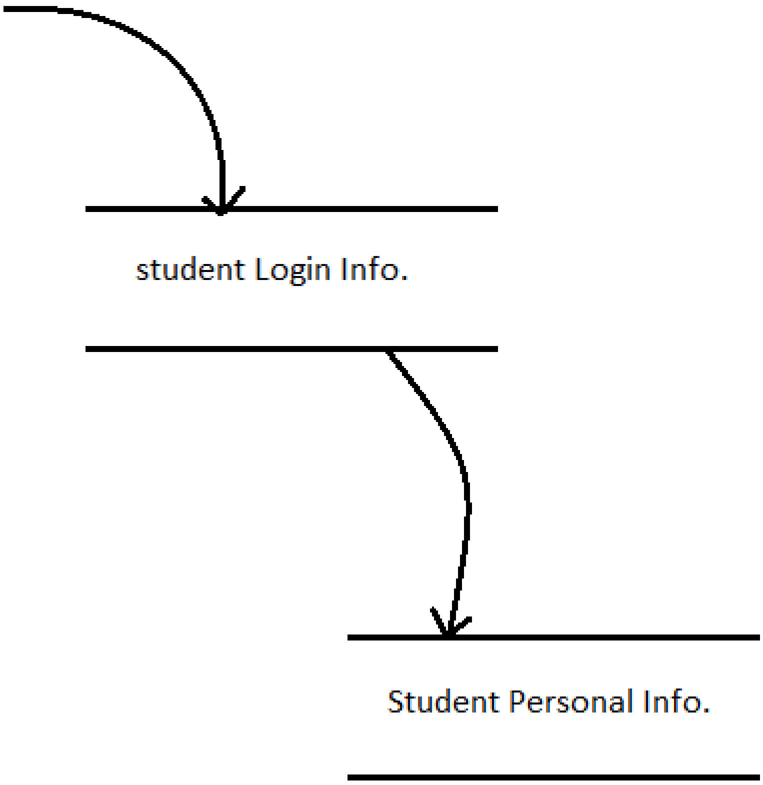
- A fresh graduate may leave different levels of DFD unbalanced. Example is shown below. Here arrow-2 is missing in level-1 of below diagram.





Note :- Arrow-2 (Sucess Msg) is missing which makes DFD unbalanced.

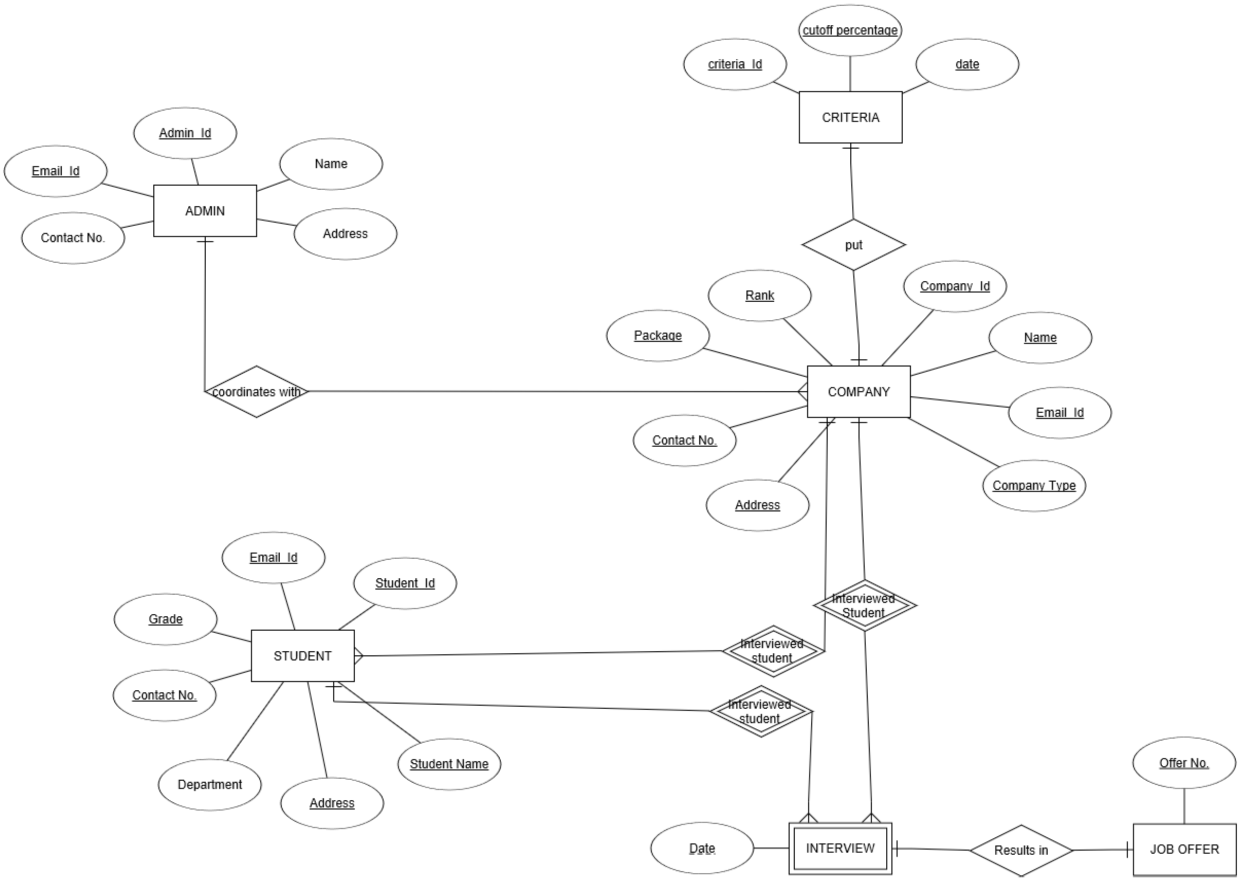
- A data store should be connected only to bubbles through data arrows. A data store cannot be connected to another data store or to an external entity. Example is shown below.



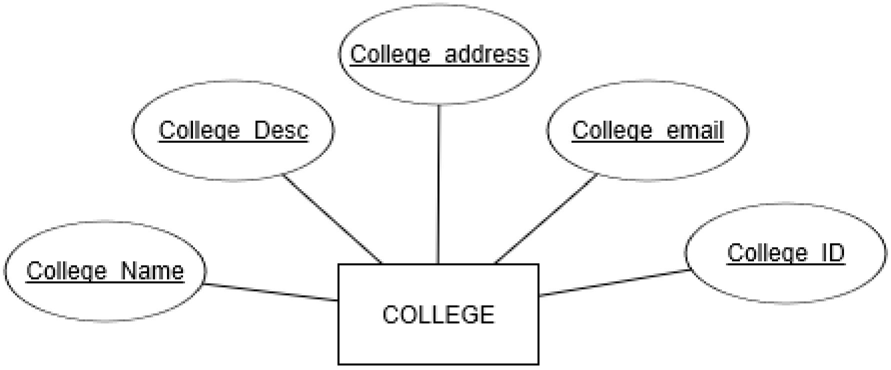
Note : Data store should not be connected

3. Construct an ER-diagram for your problem statement. lmagine that customer is changing his / her requirements after two or three days which may result in adding new entities or updating the existing entities. At this situation, you are instructed to draw a new ER-diagram. List out the possible change in requirements and corresponding changes to be made in the ER diagram (entities and relationships).

Ans.



Suppose Customer is changing his / her requirements after two or three days to add many colleges in recruitment system instead of only NITK ,then the changes to be made in ER diagram is to add one more entity college with it's attributes shown in below figure. There will be 1:N Relationship between admin & college, N:N Relationship between college & company and 1:N relationship between college & student respectively.


College_address
College_Desc
College_email
College_Name
College_ID
COLLEGE

END