University Query Handling System

REQUIREMENTS SPECIFICATION

INTRODUCTION

A specific application catering to the need of all individuals in an institution such as universities. It will provide lodging/reporting, answering, forwarding of complaints from a complainant to the respective responder.

BACKGROUND

The most difficult part in an institution is either to get information or report issues with the management. We look forward to automating this system so that the whole process is streamlined. The concerned authorities get the specifics so that they can help students, teaching staff and non-teaching staff.

- **Why did we pick this ?**

A simple to use application that allows every user in an institute to voice his opinion, provide suggestions, report incidents or lodge complaints.

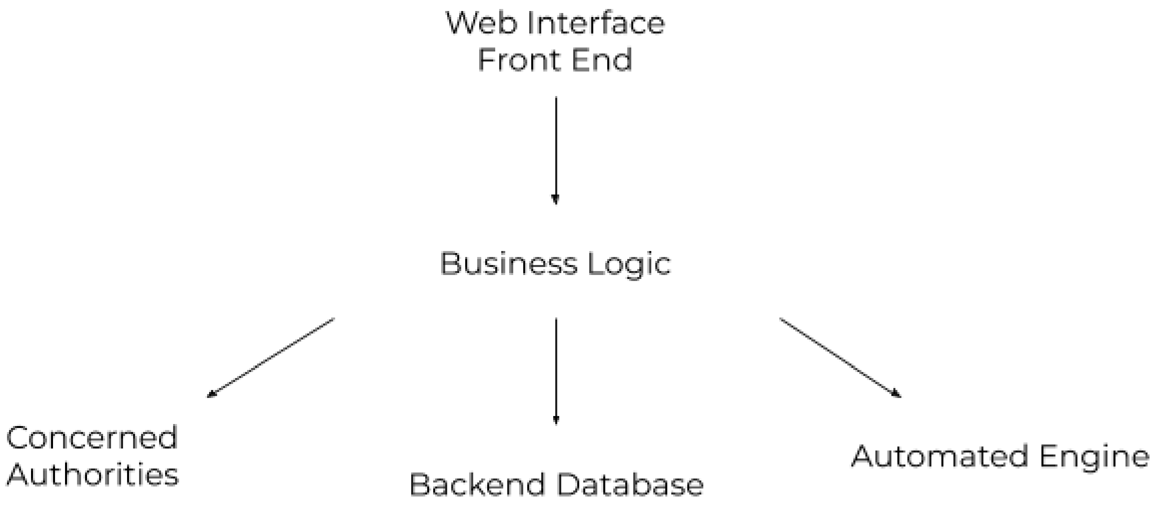
- **What problem is it solving ?**

Complaints in an educational institution often go unaddressed. Queries are a very slow process. Having a process in place will quicken the roundabout time. An interface for the user will make lodging queries much easier.

- **How are we solving this problem ?**

The application will have a few components.

A frontend web application to provide for easy use of lodging queries, business logic that will coordinate the queries along it's required route, and a backend to store, fetch and log data.



FUNCTIONAL REQUIREMENTS

General functionality, end user requirements(what are the end users and what they should be able to do, administrative user requirements, (mention details of what each user should do)

GENERAL

1. Role based login for users(students, teaching staff, etc) , Administrators and system administrator.
2. Queries can be handled in 2 ways: Manually or automatically from a database. We call the former general queries, and the latter specific queries.
3. General queries can be general questions regarding the University guidelines and processes, or complaints and suggestions.
4. All users will be able to make general queries. General queries can be sent anonymously too.

5. General queries will be looked into by Administrators, who will either respond to them, or forward it to the appropriate department/authority.
6. Users would be able to make specific queries too, as described below based on their roles:
    a. *Student specific functionalities:* will be provided with functionality to check details such as fees, courses, grades, etc.
    b. *Teachers specific functionalities:* Upload attendance for a class, upload grades, get timetable for their classes, etc.
    c. *Non- Teaching Staff specific functionalities:* will be able to see work schedule, salary details, student details, etc.
    d. Redressal officer are responsible for answering general queries and redressing complaints.

END USER REQUIREMENTS

1. The end users are basically those who want to send a query.
2. This may comprise as a teacher, student, or general staff.
3. Students should be able to :
    a. Ask for their fees status, receipt and structure.
    b. Make a general query and add appropriate tags (such as “Accounts” for fee related queries).
    c. Track general queries (queries that requires manual intervention).
    d. Lodge a complaint against the services of a particular department.
    e. Report any malicious behaviour that occurs (in the campus/ with a student).
    f. Request for a book in the library, check current book issue status, check for fines.
    g. Check placement status, companies registered, companies yet to come.
    h. Check their assessment scores, current attendance.
    i. Post suggestions.

ADMINISTRATIVE REQUIREMENTS

Listed below are the administrators:

1. Query manager
    a. In case a particular query has not been tagged to a particular department or it is a generic query
    b. It is then reviewed by this person and routed as found fit.
2. Suggestion reviewer
    a. Suggestions are reviewed by this person, if found to not be spam,it is then routed over to the respective individual.
3. Special officer
    a. Will look over reports that cite malicious behaviour in the campus.
    b. Will be given privilege over anonymous reports to trace user.
    c. Has functionality to block a person's college accounts and revoke status when needed.

NON – FUNCTIONAL REQUIREMENTS

User interface requirements

- Users should be able to send queries, check responses.
- Minimal use of the interface should lead to the required responses.
- Approximate waiting time must be shown for responses that require manual intervention.

Performance requirements

- Responses not requiring manual intervention must take less than 10s.
- End user must be provided with a seamless interface that eliminates quick interaction.
- Must support upto 1000 users at a time.

Security

- Each kind of user will have specific access control.

- Special Officer can change the access control of any user during emergency circumstances.
- Unsuccessful login attempts must be logged and an alert must be generated.

OTHER REQUIREMENTS

- Application is expected to run 24x7 365 with an availability of 99%.
- Minimize the effort the end user has to put to query.