UNIVERSITY OF THE PUNJAB
Session: BS (CS) Fall 2020 Morning
Course: Enterprise Application Development
Project II, Sessional Weightage: **10** out of 25

You are tasked to implement the following simple system. It is called **Student Interests System**. *(100 marks)*

Out of many screens of the system there is a screen to add a student in the system, and all of the fields in it are mandatory. Please check *figure 01* for the wireframe of the screen. You can change the design as per your desire.

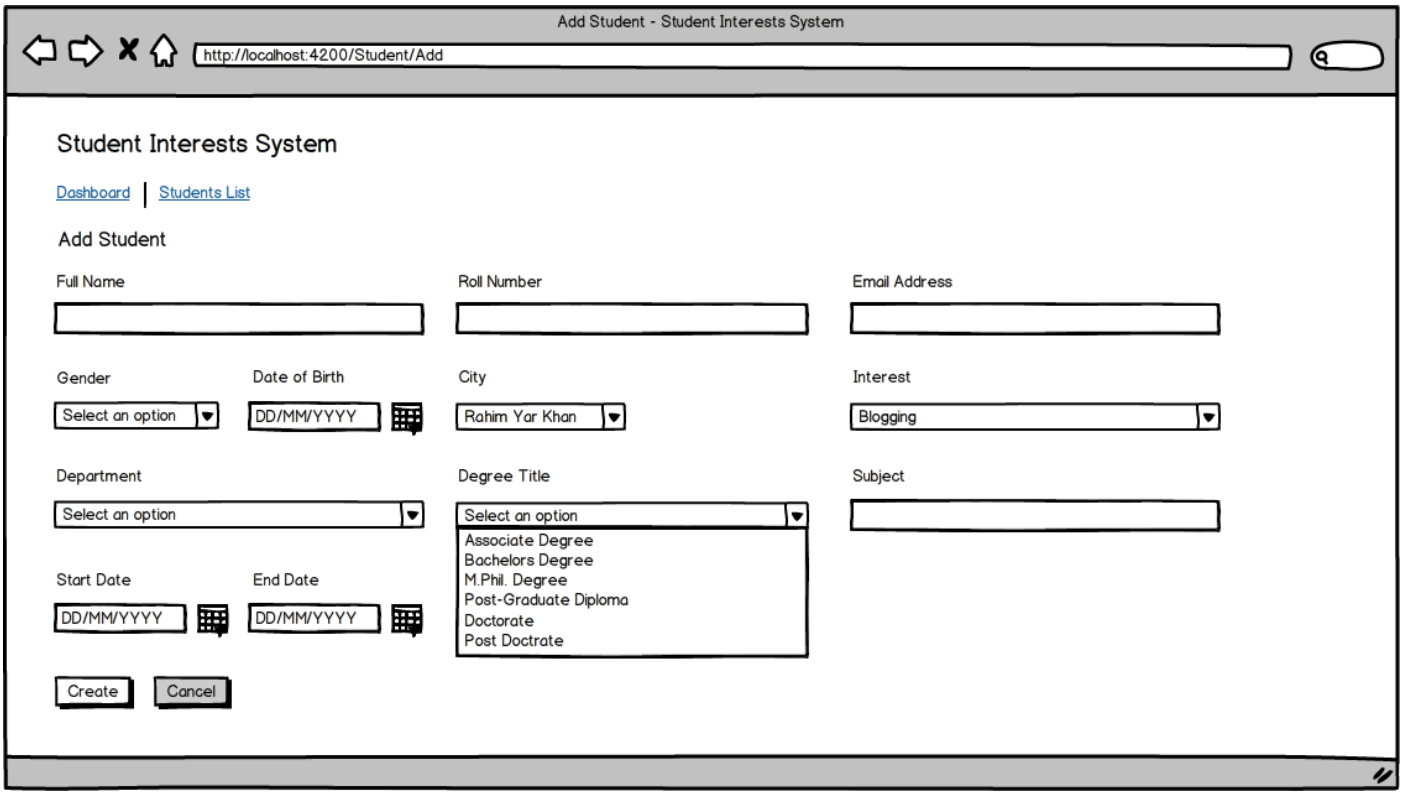

figure 01

In this screen, there is a *special* dropdown for entering the *Interest* of the student. The interest dropdown list will be populated from the data present in the database. If the desired interest is in the dropdown list then the user will simply select it from the dropdown and continue, else the user will type the new interest in the field, and on saving the student the new interest will also be saved into the database, such that on opening the screen to save a new student, the dropdown list will have the recently saved interest in it.

On clicking Cancel button, or on successful creation of student the system will redirect to Students List View. Please check *figure 02* to understand the wireframe.

On the list view, paginated data will be displayed, there would be a dropdown to select size of the page, the default size of page is 10, the page can be of size 5, 10, 20, 30 and 50, and on changing the size of the page by selecting a value from dropdown the page will be reloaded, and will display those many rows in the list.

Next to page size dropdown is page navigator, with 2 displays (current page number and total pages), and 4 actions

- << go to first page
- < go to previous page
- > go to next page
- >> go to last page

These actions will become active and inactive on the basis of total page numbers and current page number. Each row in the list will have link to View, Edit and Delete views.

Similarly, title of each column (except Actions column) of the list has upwards and downwards arrows. On clicking the upward arrow of a column the list will be reloaded and will be *ascendingly* sorted on the basis of that column, similarly on clicking the downwards arrow of a column the list will be reloaded and will be *descendingly* sorted on the basis of that column.

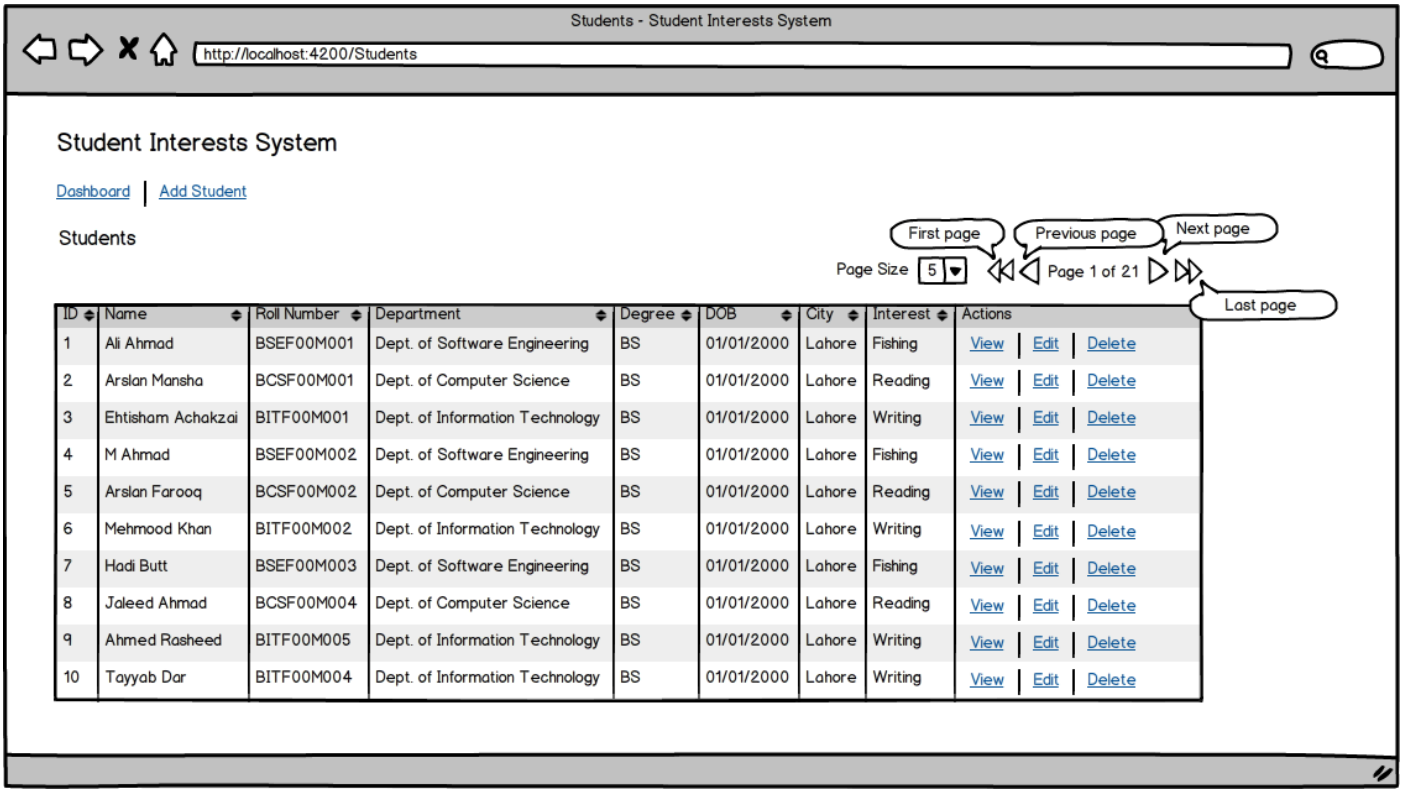

figure 02

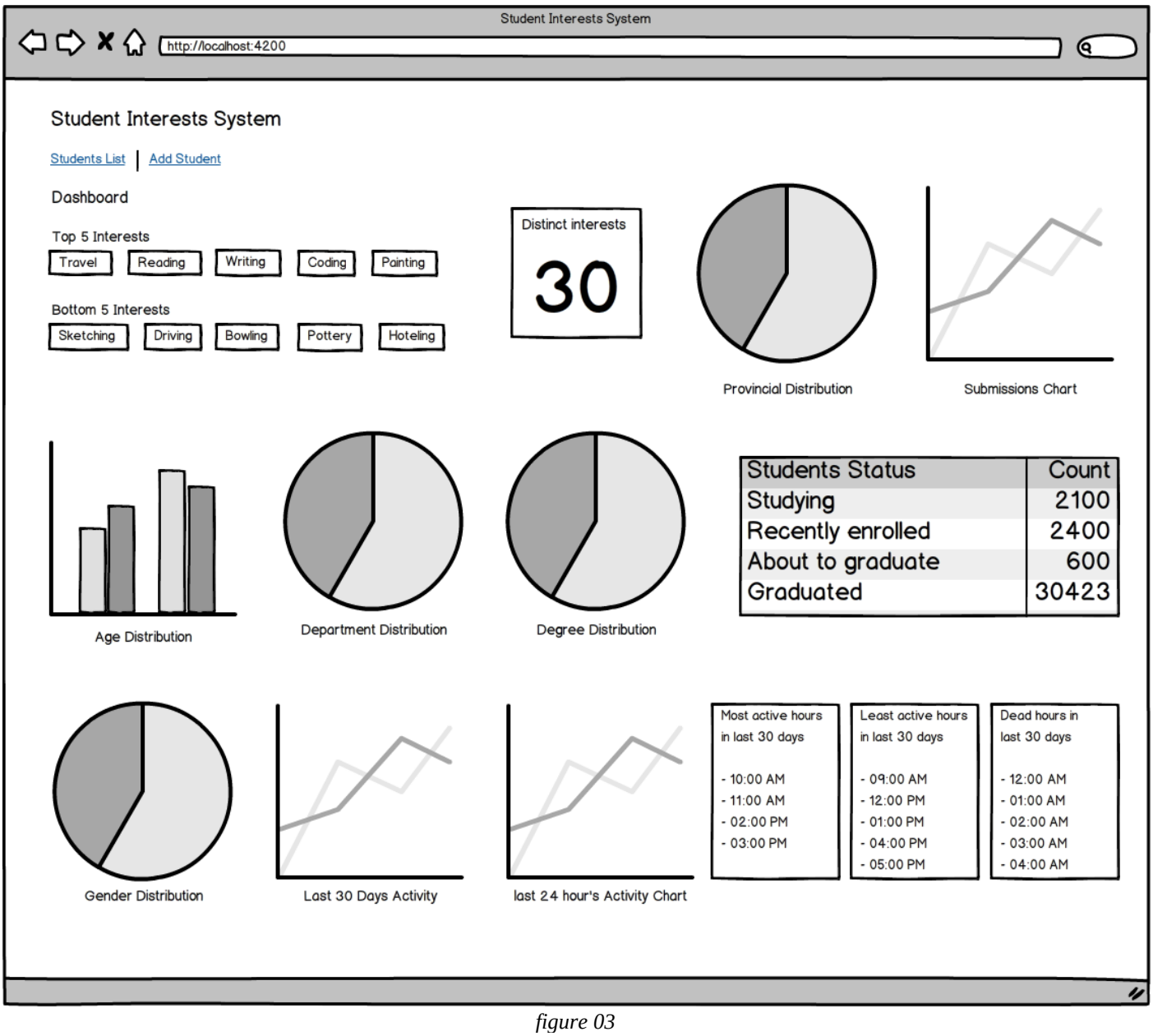

figure 03

The end result required from you is the dashboard screen, please check *figure 03* for the wireframe. The dashboard will have following widgets. You can group them as per your design.

- Top 5 interests: Green tiles each having one of the top 5 interests
- Bottom 5 interests: Red tiles each having one of the bottom 5 interests
- Distinct interests: Number of interests in the system
- Provincial distribution: A pie chart with distribution calculated among the students in the system
- Submission chart: Plotting of line chart on number of students created daily in last 30 days
- Age distribution: A bar chart plotted on the age of students
- Department distribution: A pie chart plotted on the department of the students in the system
- Degree distribution: A pie chart plotted on the degree title of the students in the system
- Gender distribution: A pie chart plotted on the gender of the students in the system
- Last 30 days activity: A line chart plotted on the number of actions performed daily for last 30 days
- Last 24 hours activity: A line chart plotted on the number of actions performed every 15 minutes for last 24 hours
- Students status grid
    - Number of students currently studying in university
    - Number of students recently enrolled
    - Number of students about to graduate
    - Number of students graduated
- Most active hours in last 30 days = List of hours in which there is most activity in the system
- Least active hours in last 30 days = List of hours in which there is least activity in the system
- Dead hours in last 30 days = List of hours in which there is next to zero activity in the system

Activity means everything that user does, such as

- logging in/out the system
- going to a screen
- changing page size
- clicking back/next button on page navigator
- adding/editing/viewing/deleting a record
- doing any type of operation on the system

To keep record of all of the activity done by a user, it should be logged somewhere.

These 3 are the primary screens of your project, other than these screens you will create screens for other CRUD operations for student entity. Similarly, you are expected to implement the user management in which users their roles and permissions are managed.

INSTRUCTIONS

- This project has to be done on individual level, every student is for him/herself. Blind cheating = 0 marks
- You can use any frontend technology, custom controls, or JS library
- You can use any backend technology
- You can use any database (except Service Based Database *.mdf* in .NET)
- Templating, UI/UX design, color schemes, and fonts of project is totally up to you
- You would have to use dummy data to display stats on the dashboard
- Projects having excellent implementation will be awarded with 15% - 30% curve in mid/final/sessional marks
- Each person will present his/her implementation in front of the class
- You will be expected to have prepared a PowerPoint presentation having the entity relation diagram, application flow, reasons for choosing the technologies, template, color schemes and fonts for his/her implementation
- You have to submit this project by Tuesday, December 26, 2023 23:59:59 PM
- Please name your private repository for this project **`RollNumber-EAD-PII`**, and add me as collaborator
- Please commit correct code and SQL files in the main branch, and mind the folder naming conventions, also upload the PPT in the repository. Submission link: https://forms.gle/PeqNvZhW8TgSegYt9

"Just take a deep breath and trust in your abilities. Good luck!"