ASSIGNMENT#5: INTERIM PROJECT REPORT (DESIGN DOCUMENT)

5% weight (A letter grade will be assigned)

Due in Class hardcopy– Thursday November 15

(Also upload on Canvas – <u>every team member uploads</u>)

Objective: This is a report to completely specify your project and to give us all related details. .
It should be a stand-alone complete report / design document of your project and reflect overall
scope, design, implementation strategy and current status. (Think of it as a document that can be
given to any team and they would know what exactly to do and to implement for this project).
Please reuse portions of the previous submission you made about the project proposal and
revised proposal etc. as you like. Make sure you take into account all comments/feedback we
gave you so far in writing and verbally.

At the time of your final demo meeting with you we will be consulting this interim report and
any feedback we write on it when we return it. You will be asked to bring that to the "demo" and
leave it with us. At the "demo" you will show us a ppt presentation spelling out the details of
what you tried to achieve, what you learnt etc. No other report will be required at the final "demo
meeting."

**The following format must be followed as much as possible. (Team Projects are considered
first. Individual ones are mentioned later.)**

Cover page: <u>Mention TEAM number</u> (important for us to keep track), latest project title, list of
project members and 3-4 line background of each member (alphabetically ordered by last names
with email of each member). Highlight the lead member of group.

- Keep the total size of report to 5-7 pages in addition to the title page. (If you need more
  pages, go ahead- we will not take that against you.)
- **Learning Goals/ Problem Statement** (depends on the nature of the project) – define
  what problem you are studying or what aspect you are trying to learn by doing this
  project. Describe what sort of application you are implementing and who will be a typical
  user of your application.
- **Approach to solve/ Functionality of the System:** the problem / overview of the
  functionality you are trying to achieve – this should include what you aim to accomplish
  in terms of the features and functionality of the end product of this project. State clearly
  <u>what is within</u> your scope and <u>what is not</u>. Each function that you will be able to show us
  must be described with some detail. <u>This is one area where the proposals have been
  weak. Be clear about what exactly you think you can implement given the time.</u>
- **Design of database** (if applicable) – as an ER diagram and as a set of tables and/or
  technical details of how you are going about the implementation. For applications of
  analytics nature, be clear to state all attributes you are capturing, your feature selection
  method (in case any machine learning procedure is applied), what data resides where,

where are the results stored etc. Every project in this course must have a database component. Pure machine learning algorithm implementation is NOT enough for this course. Note that if you are using a publicly available database (like the weather data, maps, or Kaggle data etc.), the ER diagram is very trivial- but still show it. In such cases your description of what you are doing with that data is very important.

- **Data**: What data you are using, where it is coming from. You must state this clearly and say how this data will be created (synthetically or from some website or just some mock-up samples you will create to demo your functionality). For public databases, give a description of what database (with URL) from which source, what are its overall contents, and what parts of the available data you are using. If data scraping is involved, describe it. This has been kept vague by many teams so far. Please be clear on this.

- **Interface** details: how does one use your application? Fancy interfaces are NOT expected, but how the app works and how the user interacts with the app must be clearly specified.

- **Architectural overview** (components and interfaces) as applicable, and implementation details of how the system/tool/application is supposed to work in terms of control flow or step by step operation. If your system has multiple components, describe the scope of each and its functional features.

- **Technologies:** If you are using tools and libraries, please list what they are. Environment and platform details go here and must be explicitly stated.

- **Algorithmic overview**- if applicable (if implementing some published work, explain). If you are using public libraries, published algorithms or tools, say what they are and how you will use them.

- **Demo Plan**: What you think you will be able to show as a demo (no need for fancy interface work) – you need to demonstrate and explain the capability and features you have tried to implement. Demo does not mean just showing how the application works but showing and explaining to us all the details of what you had to implement to do this project, what approach you took, what challenges you had to overcome , etc.

- **References if any**: (use the style as shown in my previous instructions about references.) Include these if you are drawing ideas from any papers in particular. All projects may not have references to list.

- Above guidelines are generic across projects. For your project, you may tailor them according to the nature of the project.
- Use above headings in your report so it will be easy to evaluate.
- Make sure to answer all questions we have raised in your proposal and discussed with you in the project team meeting.

We will evaluate on the basis of clarity of this report, completeness of the design and overall description of the approach taken and give you feedback.

TEAM PROJECT Contribution:

It would be good for you to state which member is responsible for which aspects of the project. We will assign the same grade to all members in team projects unless a concern is raised by a team member. In such cases, each member will be asked to email Prof Navathe and the TAs an "Individual Contribution Statement" saying what they did and their assessment of what others did on the project.

++

INDIVIDUAL STRAIGHTFORWARD APPLICATION PROJECTS:

- Note that individual projects which involve creating a database and showing some meaningful application in terms of transactions and queries must give details on the EER conceptual design, the data to be used, and the queries and reports to be implemented etc. They may not have much to add about architectural or algorithmic issues.
- For individual projects we expect sufficient complexity (with at least 4-5 relations involved), queries involving aggregation, grouping etc. and some report creation.

These reports should have the following sections:
- **Application description**. Include who is the user and justify why this is a meaningful application.
- **Learning Objectives**: Overall, what do you want to learn by doing this project.
- **Functionality of the System:** The queries and transactions your application will support. There must be some complex queries and reports included.
- **An EER diagram** for your conceptual schema. (must include at least 4-5 meaningful entity types)
- **An IFD diagram** showing the tasks you implemented (must have at least 4-5 queries, 2-3 aggregations and 1-2 reports)
- **Data Used**, possible sources of data, how data was scraped etc.
- **Interface Details:** What alternative user types can use the system and how. As stated above, fancy interface work is NOT expected. Focus should be on
- **DBMS and libraries** : DBMS used and any framework or libraries used

Add any additional details you feel are appropriate.