and charts are usually a sign of a restricted user interface and static SQL queries.

4 Project Deliverables and Final Project Demonstration

The database group project consists of five phases. Each of the first three phases leads to a project deliverable that summarizes the group's result of each such phase in a PDF document that has to be submitted at a given deadline for grading. The fourth phase results in a web-based database application software. The fifth phase is the project software demonstration that will also be graded.

4.1 Phase I: Requirements Analysis

In the first phase, the group's task as application developer and database designer is to propose and understand an appropriate project topic, identify its main data management needs, explore and motivate its potential for interesting queries, and analyze the needed user functionality. The group members should ask themselves questions such as

- What are the main functions that the web-based user interface should provide?
- How do the different functions work together? Sometimes there are dependencies between different functions.
- Which real-world data are needed to support the functions identified before?
- Can such real-world data be found in the Internet?
- What (colloquial) queries are important for the application?
- Which public domain and/or proprietary software is needed to perform the task? (The database system used must be CISE Oracle.)

The first project deliverable is supposed to be a detailed document (PDF file) that presents a clear and structured description and motivation of the selected project topic and its requirements that the group thinks the software solution should later fulfil. This means that a group has to carefully deliberate on the requirements and functions and precisely describe them in their document.

The focus of this project is supposed to be on the database part and not so much on the application part. This means that a group should not design and implement highly sophisticated main memory algorithms but focus on database queries that evaluate large volumes of stored data. Of course, the application part must be highly functional, and the different user functions must cooperate nicely together. However, a fancy layout design of the user interface is not required but appreciated.

It is important that each group demonstrates in their deliverable that their application would really benefit from database support and that *new information* (such as *trends*) *can be derived from the stored data*. A simple retrieval of data from the database (that is, search) or the pure connection of different tables (that is, joins) are not sufficient. As an example, let us assume that a group selects a sales application as their project topic and stores many *daily* sales numbers in their database. Of course, one can search for sales data of interest in the database and display them in the user interface. But searching only identifies an interesting subset of all data stored in the database. DBMS are specialized for search tasks, and the respective SQL queries are relatively simply structured. This project aims at more interesting queries that, first, derive new information which is not explicitly stored in the database but can be derived from the data in the database by computations and, second, represent trends (see Section 3). In the sales application, examples of more interesting trend queries are:

- What were the total *monthly* sales from 2012 to 2017? Can we observe a seasonal trend?
- How have the total weekly, monthly, or quarterly sales (in general, of product X, of products X, Y, and Z) developed in the last *n* months? Can a trend be recognized? For example, it could be that the sales are low in the summer months so that advertising efforts could be put in place in these months.
- When were the most successful or most lossy *m* months in the last *n* months?
- How has the benefit-cost ratio developed for all products, selected product categories, or individual products in a given time period? Can trend patterns be detected?

The answers to all these trend queries are not directly stored in the sales data. But they can be derived, that is, computed by (complex) database queries. Therefore, *each group has to list a number of (at least five) complex database queries in their deliverable in order to show that their application has the potential to have such interesting database queries*.

In summary, the triangle in Figure 4 shows the three main components that must fit together for a successful project. Having achieved less than these three components will not lead to success.

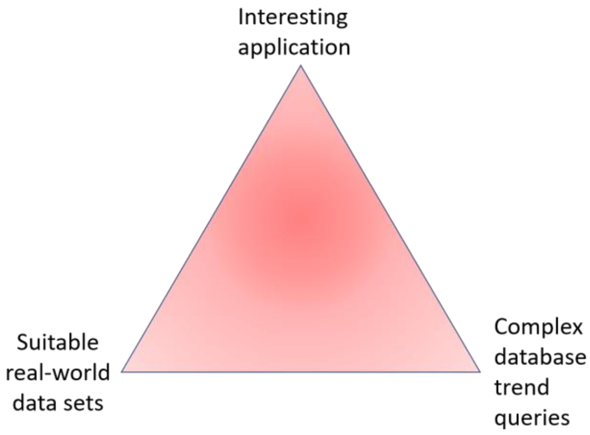
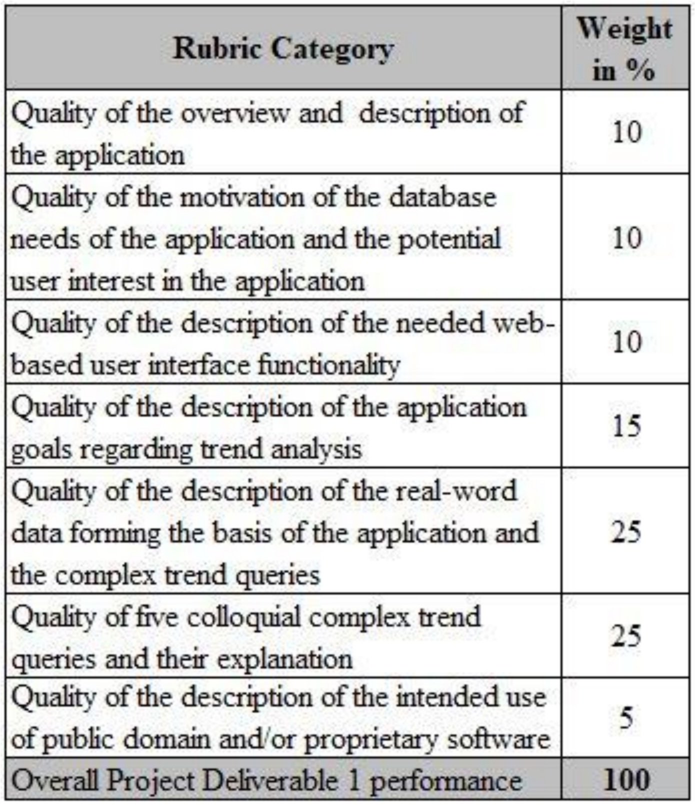


Figure 4

The rubric for grading a group’s Project Deliverable 1 submission is presented in Figure 5. The first column lists the rubric categories detailed below. The second column lists the percentage with which each rubric category will be weighted. For each rubric category a group will get a performance value between 0 and 100 according to the grading table in the syllabus in Section 4.5. The weighted rubric category performances will be added up and result in the overall group’s Project Deliverable 1 performance. By omitting the prefix “Quality of the”, each rubric category must be a section heading in the deliverable according to the order in Figure 5. This will give each group a clear indication for the contents of the deliverable and the deliverable a clear structure. Further, a group’s project must provide an application-specific title, the list of group members who have contributed to the writing of the deliverable (group members who have not contributed are not listed), and a table of contents. Each page of the deliverable must be fully filled with text and figures, and normal line spacing must be used (that is, double line spacing is not allowed). A signature page is not required.



Rubric Category	Weight in %
Quality of the overview and description of the application	10
Quality of the motivation of the database needs of the application and the potential user interest in the application	10
Quality of the description of the needed web-based user interface functionality	10
Quality of the description of the application goals regarding trend analysis	15
Quality of the description of the real-word data forming the basis of the application and the complex trend queries	25
Quality of five colloquial complex trend queries and their explanation	25
Quality of the description of the intended use of public domain and/or proprietary software	5
Overall Project Deliverable 1 performance	**100**

Figure 5

The meaning of the rubric categories is as follows:

1. *Quality of the overview and description of the application*. A reader of Project Deliverable 1 is not familiar with the application a group has in mind. Therefore, it is the group's task to provide a detailed description of the application so that the reader has a chance to understand it. Therefore, this description should be written from a reader's perspective and not from the group's perspective that already has full knowledge of the project. The group members should ask themselves what should be described so that the reader has a chance to understand the application.

2. *Quality of the motivation of the database needs of the application and the potential user interest in the application*. A first question a group should answer to themselves and to the reader is why and how their application would benefit from database support and complex trend queries. Not every application requires database support. A second question is who the users of the application are and what their interests in the group's application are.

3. *Quality of the description of the needed web-based user interface functionality*. The web-based user interface provides support for input and output. Input refers to the possible user interactions and data input. Output refers to the graphical presentation of trend query results. The question is what the requirements of user input and graphical output are regarding the group's application.

4. *Quality of the description of the application goals regarding trend analysis*. Trend analysis by means of complex trend queries is the overall topic of the database project. The question is what the group's goals and planned achievements regarding trend analysis are.

5. *Quality of the description of the real-word data forming the basis of the application and the complex trend queries*. Finding real-world data that support the group's application and enable the formulation and execution of complex trend queries is one of the main steps of the project. The real-world data sources have to be described in a manner that both the group and a reader of Project Deliverable 1 can understand their nature, properties, and weaknesses and make an assessment about their suitability for trend analysis.
6. *Quality of five colloquial complex trend queries and their explanation*. Five different colloquial queries (not copies!) that represent trend queries must be listed (at least) and described in detail. A trend query can be recognized by the fact that (i) the graphical visualization of its result is a diagram with a time unit at its *x*-axis and a numerical unit at its *y*-axis, (ii) the trend is represented as a smooth curve, and (iii) each (time, number)-pair of the trend curve is a computed value.
7. *Quality of the description of the intended use of public domain and/or proprietary software*. Each group should already begin to determine the software to be used in the implementation phase of the project. This choice can be changed at any time later if the group decides this. But this enables the group to learn the missing software components (e.g., a web-based programming language) until the actual implementation phase begins.

Several submissions of a group's first deliverable are allowed before the deadline. The most recent submission will be graded. If the most recent submission should be so dissatisfactory that the group cannot move on with the next phase, a revision will be requested from the group that will not be considered for (re)grading. This means, the more effort a group puts into this submission, the lower the probability is that a revision is needed, and the more time the group will have for their next deliverable. It is important to note that it is *not* the task of this phase to determine and describe solutions to the requirements.

4.2 Phase II: Entity-Relationship Diagram Design and User Interface Design

Based on the requirements analysis of Phase I, the goal of the second phase is to describe the overall conceptual design of and solution approach to a group's application. This incorporates the two aspects of user interface design and conceptual database design.

As for the *user interface design*, the task is to devise the set of web pages that is needed to convey the promised functionality of the application to the user at the user interface. This requires a clear description about the flows of action and web pages the user can expect. It starts with the entry or welcome web page and spreads out into a number of successive web pages that are appropriately linked with each other. The whole web site can be regarded as a graph where the nodes are the web pages and the edges are the URLs connecting the web pages. This is shown in Figure 6.