Advanced Management of Data
Project Task
Winter Semester 2020/2021

1. Introduction

Who doesn’t like to eat pizza? But these also need to be created and baked. There are so many options to compose pizzas with different ingredients and everybody has a different taste. So we need some kind of pizza manger to compose and order pizzas configurations to our needs and send them to the pizza baker.

2. Task Description
2.1 Preliminary remarks

The following practical task may be processed in groups of up to two participants. If more than one participant is present, it should be indicated for all parts (programming, term paper), who was responsible for which part.

The submission consists of:

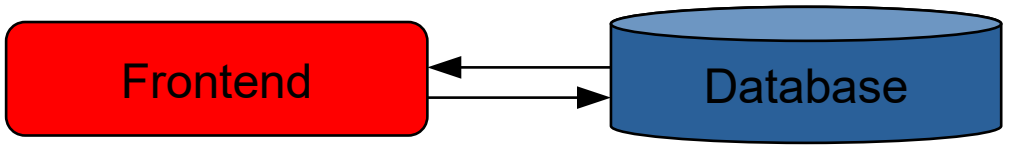
1. a PDF-file of the term paper in paper format A4 and
2. a separate ZIP-archive (with a maximum size of 10 MiB) that contains
   - the sources,
   - a script to initialize the database and
   - a small manual on how to use the sources to get a working program.

2. Task Description
2.2 Programming

The program consists of

1.a PostgreSQL-database with the program logic implemented in PL/pgSQL and

2.a frontend for interaction with the data from the database.



2. Task Description
2.2.1 Database

1. The database is provided by PostgreSQL.
2. Most of the program logic is implemented directly in the database using PL/pgSQL as main programming language.
3. All the pizza data (and related information) is stored in the database.

2. Task Description
2.2.2 Frontend

1. There are two different mutual exclusive views: pizza baker and customer
   a) it can be switched easily between both and there is no need for complex user management
2. The pizza baker view includes options to:
   a) list of all pizzas that were recently ordered
   b) manage ingredients (add, change, hide/show, remove and restock)
      - restocking performs by ordering from different available suppliers (hidden suppliers are not available)
        - it takes place immediately (suppliers have no delivery time)
      - base pizzas in all their different sizes are available in an infinite number, so there is no need to restock
   c) manage suppliers (add, change, hide/show and remove)

2. Task Description
2.2.2 Frontend

3. The customer view includes the options to:
   a) list all available ingredients along with their name, regional provenance and price
      - ingredients that are hidden or run out of stock must not be shown
   b) create a pizza composition by selecting a base pizza in a certain size and all available ingredients
   c) order a pizza in accordance with the previously created composition
4. For demonstration purposes there are available at least:
   a) 2 different recently ordered pizzas
   b) 3 different base pizza sizes
   c) 5 different suppliers with only partly overlapping assortment
   d) 10 different ingredients with various regional provenances and prices, whereof at least
      i. 3 have only 1 regional provenance and price
      ii. 5 have at least 3 different regional provenances and prices

2. Task Description
2.3 Term Paper

A term paper is to be written, that satisfies the following conditions:

1. There is an amount of about 12 pages of content.
2. On the cover page there is the name, study course and matriculation number of all students recorded.
3. An overview of all utilized technologies (excluding PostgreSQL and PL/pgSQL) is given and why they were chosen to solve a certain subtask.
4. The database tables are visualized using both Unified Modelling Language and Relational Schema.
5. The project is presented in such a way that, after reading, you know all parts of the program and their functionalities without having explicitly executed them.
6. It is discussed under which conditions a distributed database could be used with the project and how this distribution can be achieved in a good way. In case there are multiple options, no more than 3 different scenarios are described.
7. A good form and a balanced ratio of pictures and text is kept.
8. All used sources, libraries and technologies are referenced.

3. Examination

The examination consists of a 10-minute presentation, which should meet the same criteria as the content of the term paper, but the focus is on technologies and program functionality.

So the project and the used technologies should be presented, while UML, the Relational Schema and the distributed database discussion can be kept short (no more than 1 scenario here). The presentation should also include a live demonstration of the project or a demonstration video demonstrating all parts of the practical task. For group work, the presentation time should be divided equally between both students.

After that some questions (primary concerning the project and the term paper) are given. Finally there will be a short consultation and you will be informed about your mark.

4. Dates

- Handout of task description:
  - starting 2021-01-07 16:00 CET (UTC+1)
- Submission of project:
  - until 2021-02-11 18:00 CET (UTC+1)
    - late submissions will be accepted, but your final mark will be reduced by 1/3 per 8 hours or parts thereof (so your mark will reduce by 1 per day and you cannot pass, if you submit after 2021-02-14 18:00 CET)
  - via OPAL:
    ➔https://bildungsportal.sachsen.de/opal/auth/RepositoryEntry/20312915968/CourseNode/103028356652656
- Oral exam and presentation:
  - starting 2021-02-15 (until we are done)
  - modalities of appointment allocation will be announced after the project submission
    - appointment allocation most probably via OPAL, too
    - examination will take place via video conference as default, but we will try to offer personal meetings in a meeting room in the university as alternative