TeraCode

Developer Hiring Exam

Exercise One

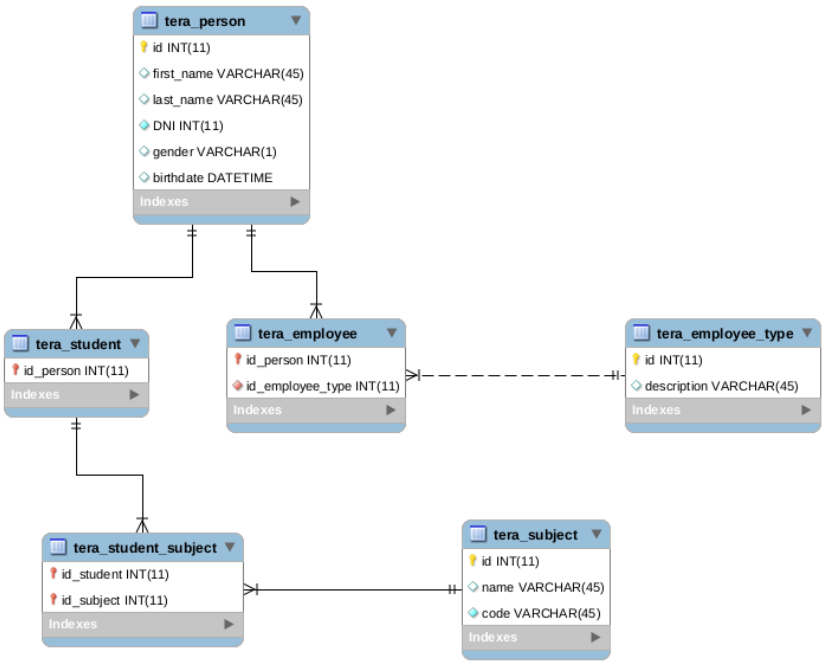
A) answered in the code:

- URL: https://github.com/juanma1982/teracode/tree/master/src/com/teracode
- FILE: School.java
- Method: listStudentsGroupedByFirstLastNameLetter

B) answered in the code:

- URL: https://github.com/juanma1982/teracode/tree/master/src/com/teracode
- FILE: School.java
- Method: getStudentsInSubject

C) The diagram is shown below, but the PNG original file is in the repository under “docs” with the name “entity relationship diagram.png”



Inheritance was implemented in two ways: first as a table whose main ID is a foreign ID of the person table. This way you can recover all the information from the “father” through a JOIN and

add new attributes of the “son” table. The second way, used by tables which inherit from employee, use employee_type column to discriminate the class. This refers to different types of employees.

The many to many relation between subjet and student was resolved as a relation-table: tera_student_subject. This table has as unique ID both foreign IDS, so there is no possibility to have a same student for same subject.

Finally, the fields DNI (in student) and Code (in subject) were restricted as UNIQUE to avoid possible duplicates. They were not chosen as table ids, because it’s not a good practice, because an unexpected modification of these fields will impact in all indexes and foreign keys, so autoincremental integers were created for record identification.

D)

Depending on the Database engine, I will use different tools to check which is the most expensive part of the query. For example MySQL Performance Schema for Mysql, etc. There are several things to do in order to improve a query, but depends on the problem:

1. If there is a plain "select *" without joins, the inmediatilt solution is to paginate the query in the client, maybe using ajax. Or changing the "*" for a short list of columns. (Some columns blobs, text, etc are very heavy to laod)

2. If the query has joins, maybe changing JOINS for nested queries

3. If the query has filters, maybe you can create an index.

the last solution is improve hardware. (also check the LAN latency)

E)

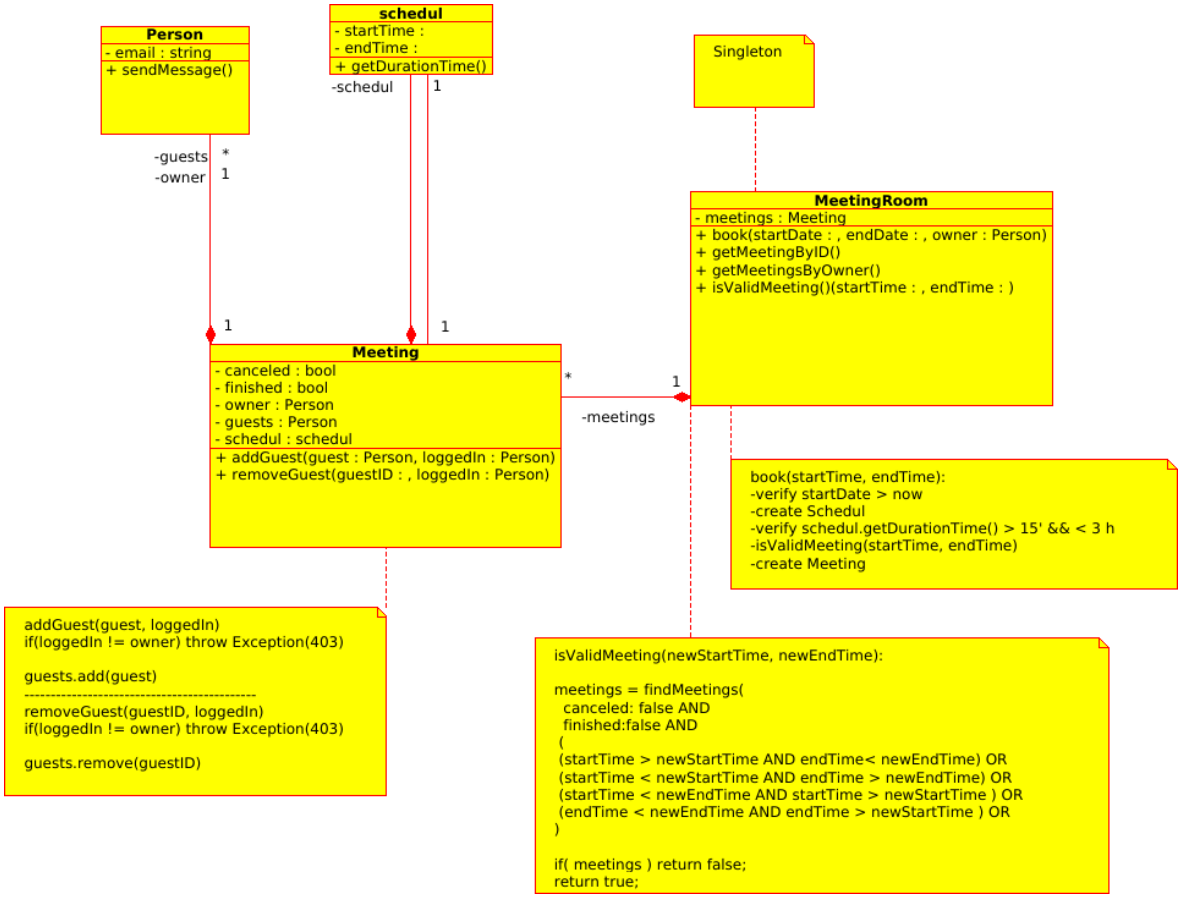
First thing to do is to change "age" to DATE, because is not a good practice to storage age but storage birthdate. So I will have two different dates, “startDate” and “endDate”, and the query will have the following shape: “select * from student where birthdate > startDate and birthDate < endDate”.

To improve this query I will create an index for column birthDate

F) To delegate all the saving logic to Database i thing the best tool is to use stored Procedures.

Exercise Two

Domain Model:



User interface and booking flow

The flow is the following:

1. The user select year and month, using top controls. By default current month and year are selected.

2. Select a day in the calendar. By default today is selected

3. In the bottom of the screen, a control shows a time-line with meetings and empty blocks of times. If user click there a new screen is shown (see image 2) in order to create or modify a meeting.

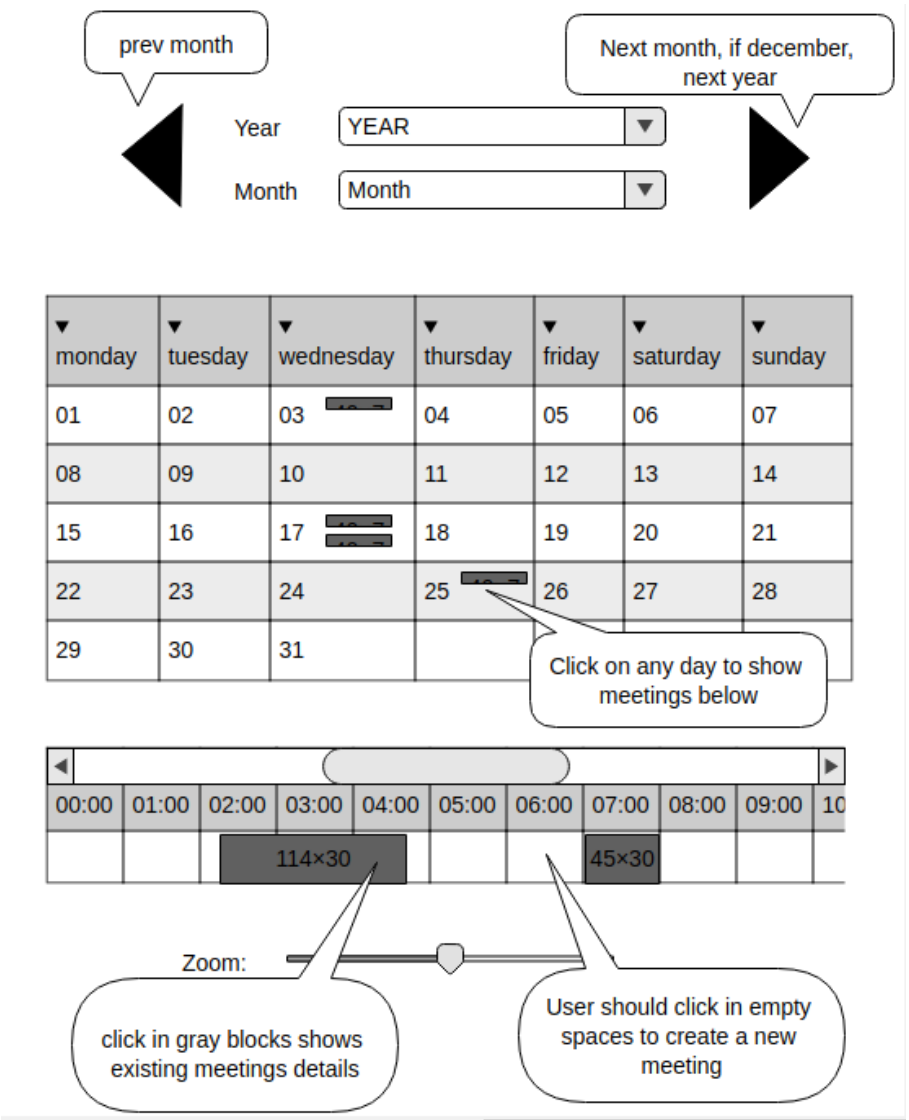

prev month
Next month, if december, next year
Year
YEAR
Month
Month
monday
tuesday
wednesday
thursday
friday
saturday
sunday
01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
Click on any day to show meetings below
00:00
01:00
02:00
03:00
04:00
05:00
06:00
07:00
08:00
09:00
114×30
45×30
Zoom:
click in gray blocks shows existing meetings details
User should click in empty spaces to create a new meeting

Image 1

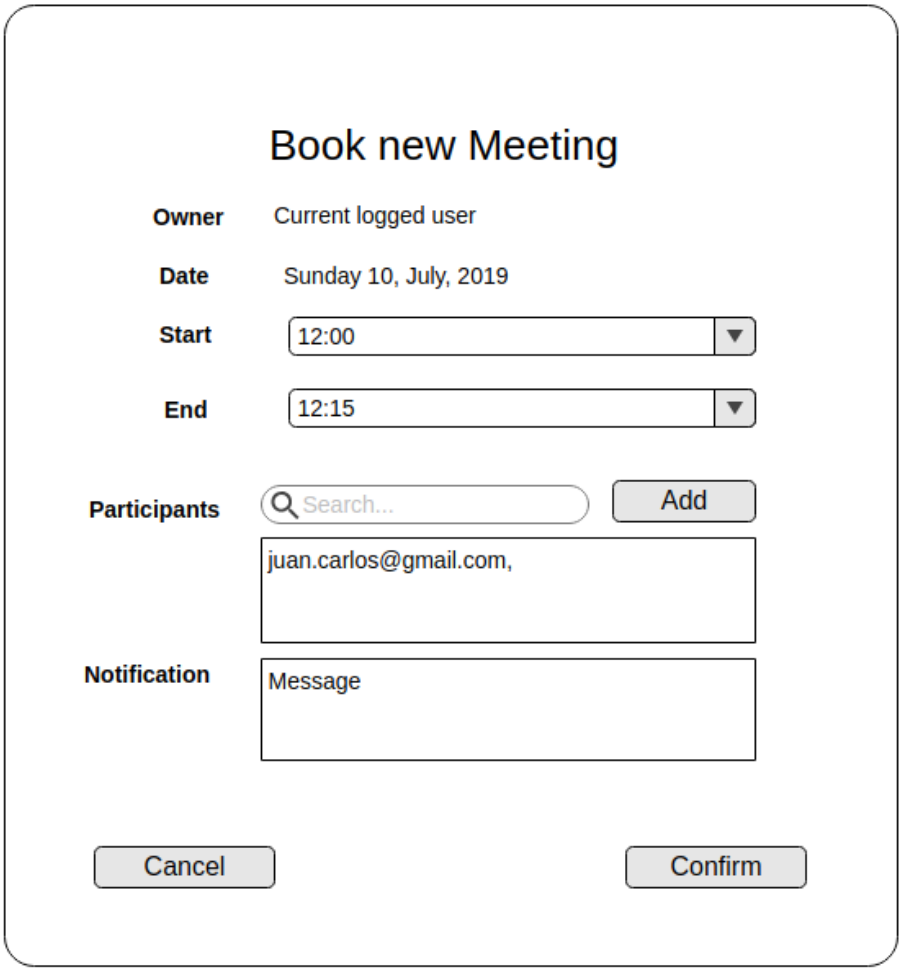

Book new Meeting
Owner
Current logged user
Date
Sunday 10, July, 2019
Start
12:00
End
12:15
Participants
Search...
Add
juan.carlos@gmail.com,
Notification
Message
Cancel
Confirm

Image 2