Data Base

T-DAB-810

The Square

Request for proposal

Table of contents

[Introduction 3](#_Toc510448397)

[1. Project context 3](#_Toc510448398)

[2. Global architecture. 3](#_Toc510448399)

[3. Component description 3](#_Toc510448400)

[a. Component 1: Organization for the database. 3](#_Toc510448401)

[b. Component 2: Using the database. 4](#_Toc510448402)

[c. Component 3: Manipulate the database. 4](#_Toc510448403)

[4. Traceability matrix 4](#_Toc510448404)

# Introduction

The aim of this software architecture specification (SAS) is to present the technical elements necessary for the Square project.

# Project context

The provider will have to propose and create a data base for this new social professional network for the app. It must choice the best Database model in response to these requirements.

# Global architecture.

The software must implement the following requirements and are organized in four categories:

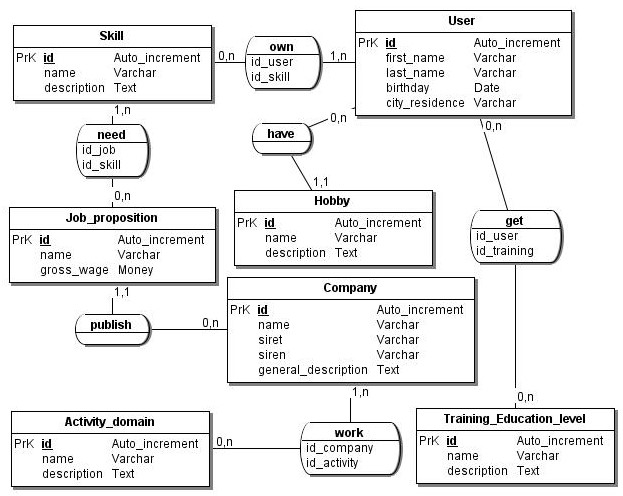
* REQ\_DESIGN\_XXX: design requirements show the relations and behaviour between each tables/entities of the database.
* REQ\_FUNC\_XXX: functional requirements must propose relation between two people with some constraints.
* REQ\_IHM\_XXX: IHM requirements ask for interface.
* REQ\_DATA\_XXX: data managing asks for some information for the storage in database.

So, the different components we need are tables and attributes to store data. Database use a relational database for the social professional network. It is better to manage the relationship for people. Update the database is easier with different relations. MySQL, the tools which will be used, offers interesting possibilities for easy updating of the database.

Import SQL scripts, view and graphically add entities and attributes, some useful features to manage a database. In addition, MySQL is often used for websites of small and medium size.

# Component description

## Component 1: Organization for the database.

To understand the subject, a mcd is useful to see clearly the attributes of tables and their relationship. The first component consists to create the SQL script which contains pieces of information about tables with data and their relations, assimilated for certain to tables.

## Component 2: Using the database.

Description of component 1: utility, main classes, key functions and actions, link to other components.

## Component 3: Manipulate the database.

We must have a user interface to use the database. For the moment, we will write some request in this tool to show some result we want. MySQL seems to be a very popular SGBD for the web.   
It is also widely used for websites. In addition, the corporate social network that we must create is likely to be realized in the form of websites such as Facebook or Linkedin. Although the latter certainly use NoSQL databases, our social network will initially be quite small.

# Traceability matrix

This matrix makes the correspondence between components, classes, functions and requirements developed in the request for proposal.

|  |  |  |  |
| --- | --- | --- | --- |
| Id requirement | ***Requirement description*** | ***Component*** | ***Function / action*** |
| REQ\_DESIGN\_XXX | | | |
| REQ\_DESIGN\_010 | You shall have a relation between skills and users | Component 1 | A user can have one or n skills.  A skill can be owned by one or n users. |
| REQ\_DESIGN\_020 | You shall have a relation between hobbies and users | Component 1 | A user can have zero or n hobbies.  A hobby can be used by one or n users. |
| REQ\_DESIGN\_030 | You shall have a relation between training/education level and users | Component 1 | A user can get one or n training.  Training can be obtained by 0 or n users |
| REQ\_DESIGN\_040 | You shall have a relation between activity domain and company | Component 1 | A company can work in one or n activity domain.  Activity domain can have 0 or n business. |
| REQ\_DESIGN\_050 | You shall have a relation between skills and job proposition | Component 1 | A job proposition need one or n skills.  A skill can be used by 0 or n job propositions. |
| REQ\_DESIGN\_060 | You shall have a relation between job proposition and company | Component 1 | A company can publish 0 or n job propositions.  A work proposition can only be published by a company. |
| REQ\_DESIGN\_070 | You shall use only one database | Component 1 | The name of database is dtb\_thesquare |
| REQ\_FUNC\_XXX | | | |
| REQ\_FUNC\_010 | You must propose relation between two people depending of:   * Hobbies * Skills * Trainings/education level | Component 2 | Create a data query using relations between hobbies, skills and trainings/education level.  The result shows a board with the data. |
| REQ\_FUNC\_020 | You must propose relation between two people function of own relationship. | Component 2 | Create a data query to propose adding people in contact based on the number of contacts in common. |
| REQ\_FUNC\_030 | You must propose relation between people and company function of:   * Skills * Skills needed * Activity domain | Component 2 | Create a data query |
| REQ\_IHM\_XXX | | | |
| REQ\_IHM\_010 | The interface is free. Simple web page. | Component 3 | Where we can test pieces of information from the database. MySQL will be used for this. |
| REQ\_DATA\_XXX | | | |
| REQ\_DATA\_010 | You must have users’ information:   * First name * Last name * Birthday * City of residence | Component 1 | Create table “*user*” with:   * id (auto\_increment) * first\_name (varchar) * last\_name (varchar) * birthday (date) * city\_residence (varchar) |
| REQ\_DATA\_020 | You must have company’s information:   * Name * Siret and siren * General Description | Component 1 | Create table “*company*” with:   * id (auto\_increment) * name (varchar) * siret (varchar) * siren (varchar) * general\_description (text) |
| REQ\_DATA\_030 | You must have skills | Component 1 | Create table “*skill*” with:   * id (auto\_increment) * name (varchar) * description (text) |
| REQ\_DATA\_040 | You must have hobbies | Component 1 | Create table “*hobbies*” with:   * id (auto\_increment) * name (varchar) * description (text) |
| REQ\_DATA\_050 | You must have training/education level | Component 1 | Create table “*Training\_Education\_level*” with:   * id (auto\_increment) * name (varchar) * description (text) |
| REQ\_DATA\_060 | You must have activity domain | Component 1 | Create table “*Activity\_domain*” with:   * id (auto\_increment) * name (varchar) * description (text) |
| REQ\_DATA\_070 | You must have job proposition with:   * Name of job * Description * Gross wage | Component 1 | Create table job\_proposition with:   * id (auto\_increment) * name (varchar) * gross\_wage (money) |
|  |  |  |  |