Data modelling and databases project. *PublicLib* web user interface

[Extended Abstract]

Boris Makaev
Innopolis University
Russia, Innopolis
sandman96@mail.ru

ABSTRACT

This paper describes project I've made for Innopolis University DMD course. First to say, my team is just only me and I've done this work by myself. Of course, this was one of my most challenging issues, but it was a great experience I've had. Before this project I have never had a chance to meet with Back-End of Web sites. And this was hard and tough work for me. This paper contains full understanding of how my system works and how it has been designed. This project itself is composed from two phases, which are implemented in this system. There will be screen shots of code and web UI so you can see the result of my work. And, of course, they will help you to understand my project more deeper.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; D.2.8 [**Software Engineering**]: Metrics—*complexity measures, performance measures*

General Terms

Web UI, DBMS, Databases

Keywords

Innopolis University, LaTeX, DMD project, WUI, PostgreSQL

1. INTRODUCTION

So let's start our journey of developing my DMD project - PublicLib. From the start all students knew their task and what to do exactly on this project. The main idea of this project is to create an Web UI that can search through publications in DB. Also there needed to be more than million publication in DB. Describing in few steps, we should have implemented in such order:

- Designing ER-model that would represented our data in DBMS later. It is more easier to understand what to do, when you have finished ER-model.
- Based on ER-model create Relational model of our future Database.
- When we develop our Relational model, we also might consider Normalize our relations and decompose them in more efficient relations, so it would be more efficient when we implement them in DBMS. There are few normal forms, and each one set Functional Dependencies on our relations, so some attributes are dependent from others.
- When we finished our work with relations we must implement them in DBMS. I have chosen PostgreSQL 9.4 for this project, because it was tight bounded with course where we were studying PostgreSQL as main DBMS. And also it is very good supported, open-sourced project that has great community. It was very interesting to read documentation.
- When our database "skeleton" is ready, we need to fill it with data. I have chosen DBLP for data supply. It have already .XML file to download with all publications. So I parsed it through Java SAX parser in my DB tables.
- We have filled DB with articles that ready to be used. Now where then fun and interesting part begins - Designing and Developing WebGUI. Back-en, Front-end, this will be discussed later in sections

I will talk more closely with examples in next sections. Citation of Einstein paper [**?**].

2. ER-MODEL DESIGN

This was first step in developing this project. Starting point for all DB design. When I chose resource when I can get data about publications, I started to think how ER-model could be represented. The main goal of my ER-model was to separate Conferences and Journal Papers. I thought it would be great to create sort of two fractions that will be display differently in WUI.

We can see this ER-model in figure 1. There are 4 main entities. Author, Paper, Conference and Journal. Author entity keeps all info about author we need. Name of author

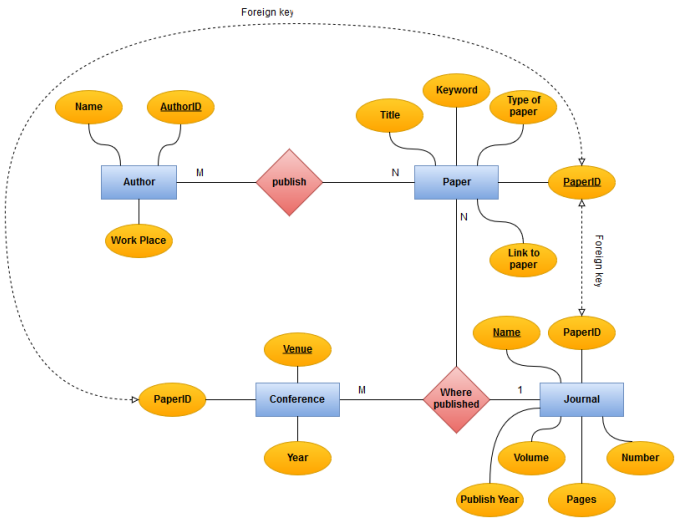

Figure 1: ER-model

and Workplace(Institute). Paper is a common entity for all publications. They all have these attributes no matter is it Conference or is it Journal. And as we can see Journal and Conference entities keep info about specific type of paper. All of them bounded with relations and some of them with foreign key. So this is it for ER-model, let's keep moving toward next steps.

3. RELATIONAL MODEL

This is quite important step when we create relations from ER-model. When it comes to normal forms, my relations represent 2NF when we have a key that define all attributes. But I added some new ID values in Conference and Journal further, because my model was not fully correct at start. So here my first version of relations that I've made:

- Author(AuthorID : integer, Name : string, WorkPlace : string)
- Paper(PaperID : integer, Type : string, Keyword : string, Title : string, Link : string)
- Conference(Venue : string, Year : integer, PaperID : integer)
- Journal(Name : integer, Volume : integer, Number : integer, Pages : string, Year : integer, PaperID : integer)
- publishedPaper(AuthorID : integer, PaperID : integer)

As we can see that Venue of Conference and Name of Journal are keys of that tables, but I added auto increment primary key further in my DB implementation.

4. DBMS CHOICE AND DB IMPLEMENTATION

So here we are. We have a scratch design that we want to implement. First we need to choose DBMS sytem. As I have already said, I chose PostgreSQL. It has many pros and cons, but main reason I have chosen this system, that we were studying it and practiced in it and I thought it would be great idea to improve my skills here. So we have DBMS system. Let's implement empty(for now) tables. Let's see code:

```
--Table with author's name and work place--
 CREATE TABLE Author(
AuthorID SERIAL PRIMARY KEY,
Name VARCHAR(200) UNIQUE,
WorkPlace VARCHAR(200)
);

--Table about paper itself
CREATE TABLE Paper(
PaperID INTEGER PRIMARY KEY,
Type VARCHAR(200),
KeyWord VARCHAR(200),
Title VARCHAR(200),
Link VARCHAR(200),
Year VARCHAR(200)

);

--Relation that bound first two tables
CREATE TABLE published_paper(
AuthorID INTEGER,
PaperID INTEGER,
FOREIGN KEY (AuthorID) REFERENCES Author(AuthorID),
FOREIGN KEY (PaperID) REFERENCES Paper(PaperID)
);

--Two tables below describe where our paper is been publish
--Conference
CREATE TABLE Conference(
  ConfID SERIAL PRIMARY KEY,
  Venue VARCHAR(200),
  Year VARCHAR(200),
  PaperID INTEGER
);
--Journal
CREATE TABLE Journal(
  JournalID SERIAL PRIMARY KEY,
  Name VARCHAR(200),
  Volume VARCHAR(200),
  Number VARCHAR(200),
  Pages VARCHAR(200),
  Year VARCHAR(200),
  PaperID INTEGER REFERENCES Paper(PaperID)
);
```

. We can see all tables with their constraints. Some of attributes doesn't have constraints, because of I reached some problems with my parser. As we can see that Name of author is unique, that is why if we had two different authors with the same name it would count as if it was One author after all.

5. PARSING DATA INTO TABLES

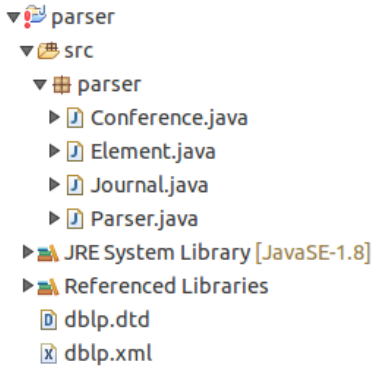

Figure 2: Java SAX Parser structure

. We have implementation of tables and now we want to add data into them. I have chosen DBLP resource where I could get data. In the beginning of my work I didn't know about XML parser at all and it was hard work for me how things work. After some days I've started to getting a clue that I am about to find out how exactly this Parser works. In figure 2 we can see files that Parser had in the end. As Parser I've chosen Java SAX parser template and configured, or "tried to configured" regarding my tables. After a week I started parsing data. There was some mistakes in my Parser configuration and, for example, titles of publications with special symbols were parsed incorrectly, but still there not many of those.

6. WEB USER INTERFACE. A LITTLE INTRODUCTION

Now the second big part of my project starts. In first part we ended up with ready-to-work-with tables, that has over 1 million publications (1138263 to be precise) and now we need to create web user interface to interact with this data. Let's point out issues I've met when I just started working on it. What stack of technologies to use? This was the first question I've asked myself. For front-end there was no doubt about HTML/CSS(a bit bootstrap). The hardest thing was to decide what back-end technologies I will choose. So let's point out my final choices:

1. PHP 5.3
2. Apache Server
3. HTML/CSS
4. jQuery (a little bit just for better design)

Firstly I wanted to choose Python to work at back-end like my friend did. But other said it would be easier to develop on PHP. Actually, I was interested in learning both of them, because I have never code on them and this was a good challenge for me. So PHP became my core back-end language. At this point I want to tell that of course I've met some problems. Not with learning PHP but in back-end coding techniques I had in the beginning. Code redundancy with searching algorithm made it impossible to maintain(2000 code lines...How? Ask myself two month ago). I knew there was something I needed to change, so I rewritten code(not completely, of course, algorithm still same as before, just used more proper functions to provide searching) and reduced my code. And in the end I still had some problems with my code, while learning more about web development I noticed that my implementation is far away from ideal, but I know now exactly how I should start working on such projects now, where to begin.

. If we talk about Apache server then it is common popular server. When I've chosen it I didn't thought about it's pros and cons, just wanted to base my site somewhere.
Talking about HTML/CSS. I wanted to create good looking site, without using Bootstrap framework, I think it's kind of soulless, but of course good scalable. But I was to interested in creating this site from scratch that I forgot about good Bootstrap positioning solutions. That is why my WUI is not scalable at all, but works fine on my screen resolution. Bootstrap in my WUI is used in inputs and some submit buttons. So let's go to the WUI itself.

7. WUI IDEA AND FUNCTIONALITY

Idea I've had from start that it is publication system that gives you chance to interact with DB data. Let's look at all functionality that I have implemented in my WUI:

- Only registered people can have access to publications. There is no role differentiation so everyone who registered can delete or update all publication they want.

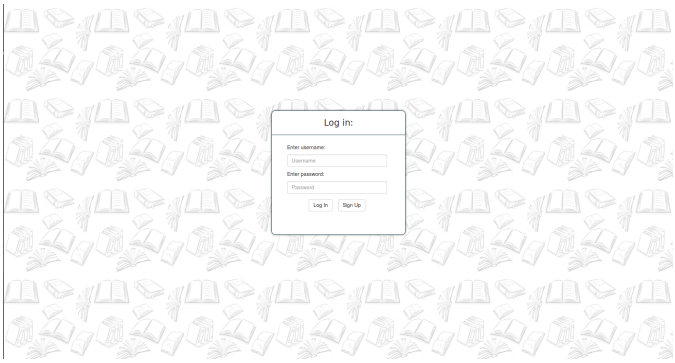

Figure 3: Login

- Search for publications by some criteria(To be precise title, year of publish, author name, keywords, conference name, journal name) We can see search bar that has criteria for searching through papers and authors Down below we see restult of searching through authors by name and publications by title. As we can see we can sort results by title, year asc and desc, and result are paginated there is a limit for each kind of result(author, conference, journal) of 10 blocks by page.

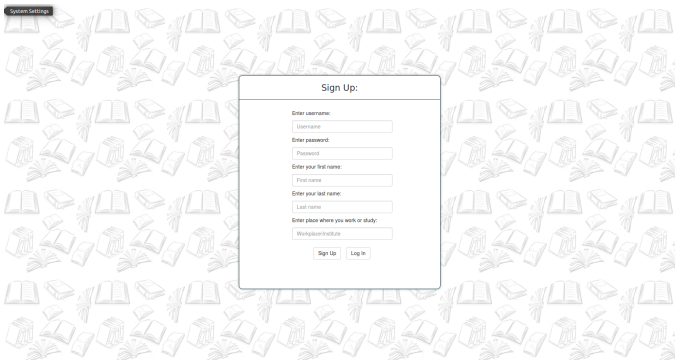

Figure 4: Registration

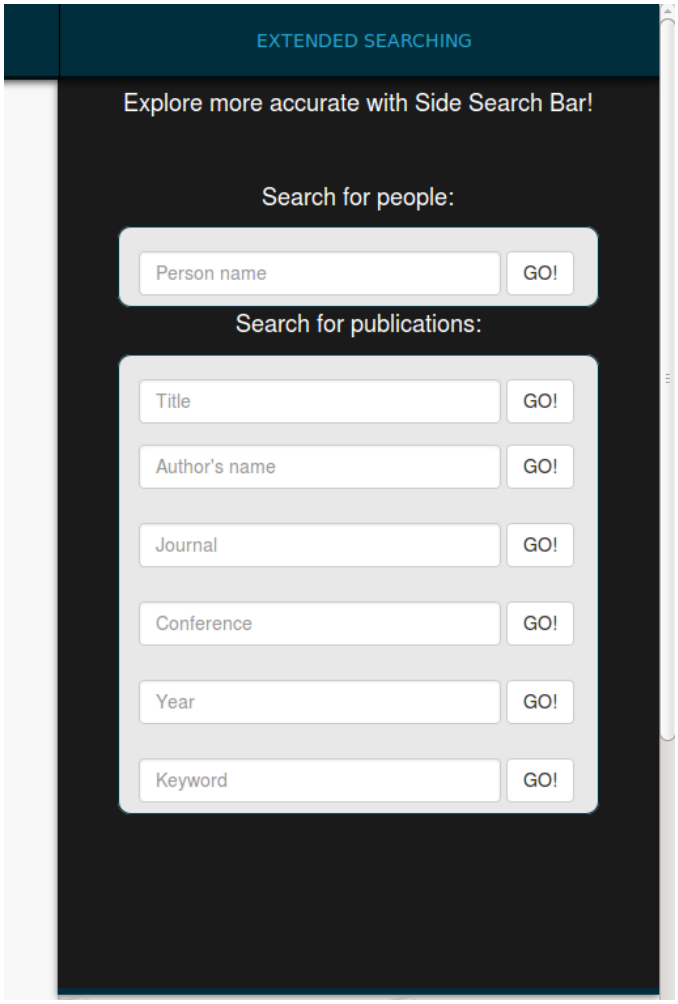

Figure 5: Search bar

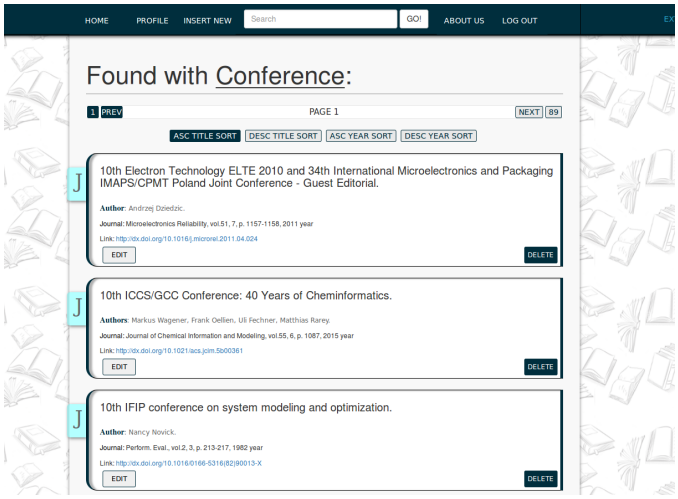

Figure 6: Publication search(searching style same for all criteria)

- Edit, delete, insert publications. But!!! There is a big weakness in my system. Edit doesn't work correctly. I don't know why, it was working fine for a while, but then I changed something and It updates only Authors.

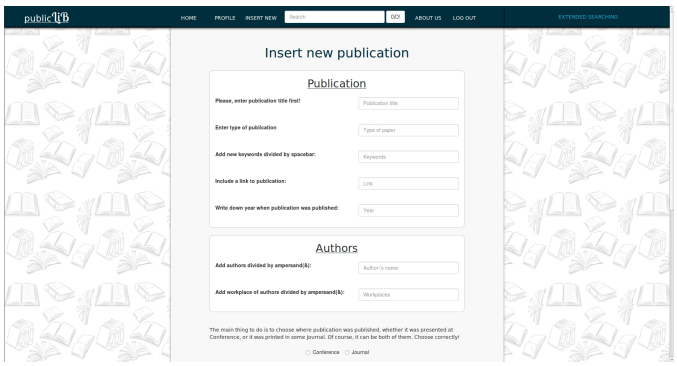

Figure 7: Insert publication(Edit style the same)

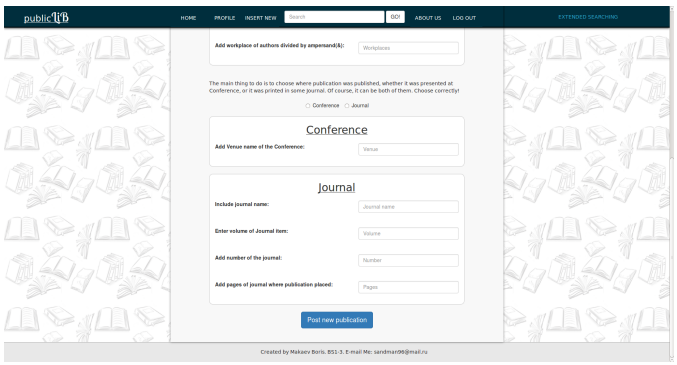

Figure 8: Insert publication(Edit style the same)

- Search for authors by their name. And we can subscribe on authors

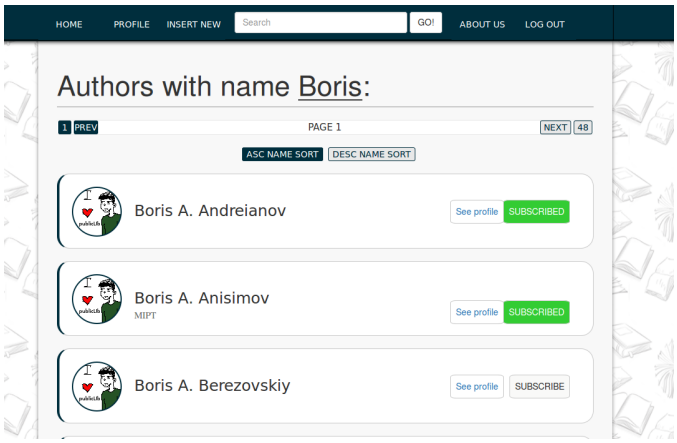

Figure 9: Author Search

- I added feature to this WUI that I think is interesting idea. I added Profile for users where you can see your subscriptions

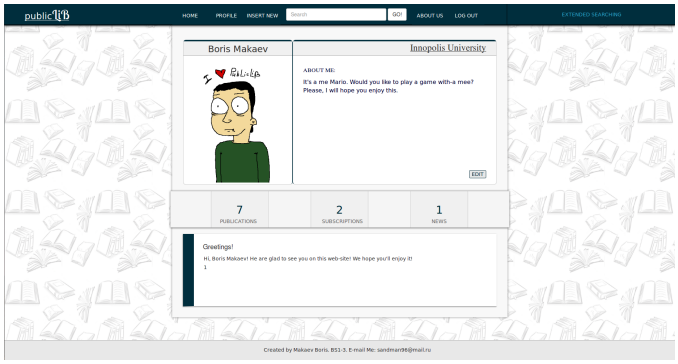

Figure 10: Profile with opened tab News

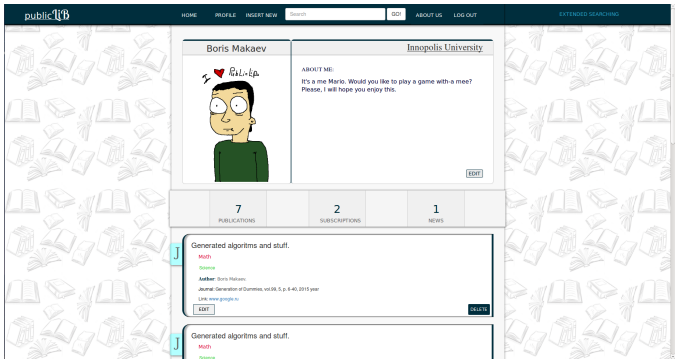

Figure 11: Look at publications of user

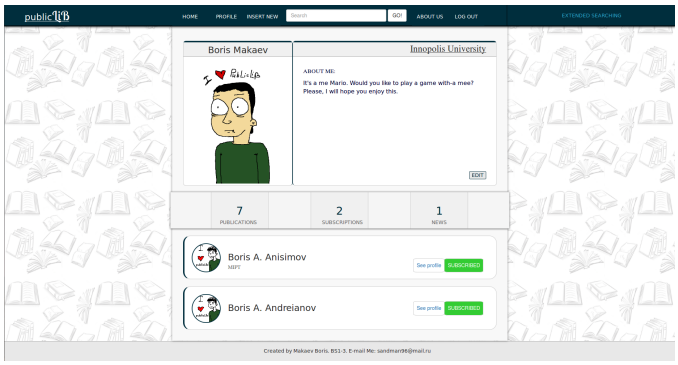

Figure 12: Look at subscriptions

Also we can go to the page of followed person and see what papers did he publish. As we can see design is the same as user profile but only with publication printing

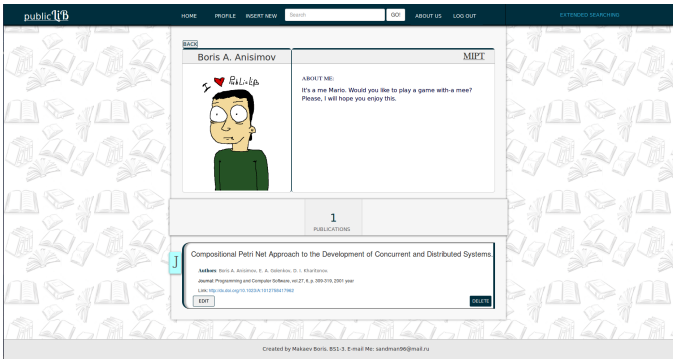

Figure 13: Look up publications of followed author, or just an author.

So here it is. That is all for my project

8. CONCLUSIONS

So I have finished my project and I hope you will like it. I got experience in web development, this was challenging task for me, but very interesting! I learned new technologies to create good projects, it is a good starting point to learn new things.