Development of

A web based DBMS supporting development of Armenian Science Citation Index

Davit Abovyan

Supervisor: Arthur Petrosyan

Topics being covered

About the purpose of the project

Prototypes (screenshots)

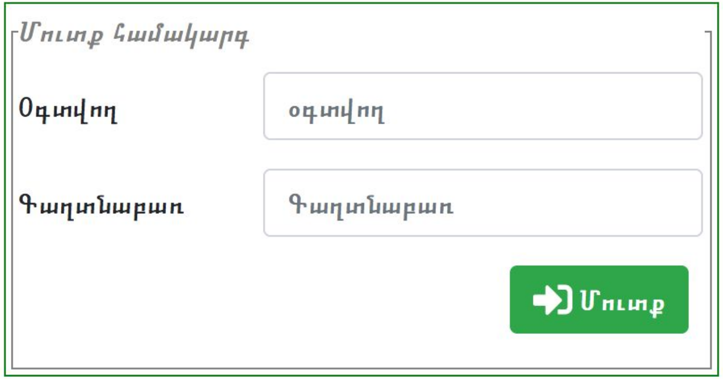
Tech approach

- High level tech approach
- Back-end tech approach
- Database tech approach
- Front-end tech approach

About the purpose of the project

In the scope of this project a web based data management system should be developed for Armenian Science Citation Index (ASCI) which can later grow into a web portal including complete and comprehensive bibliographical data on all scientific publications in Armenia, their citations in other publications worldwide, a centralized multi-functional information system that will incorporate 'bibliometric' data for Armenian researchers.

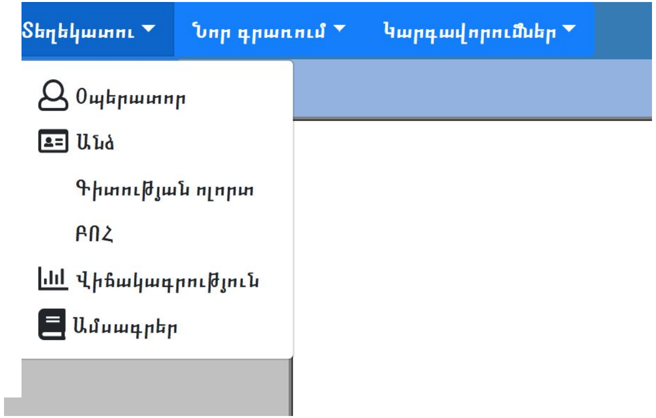
Prototypes (screenshots)



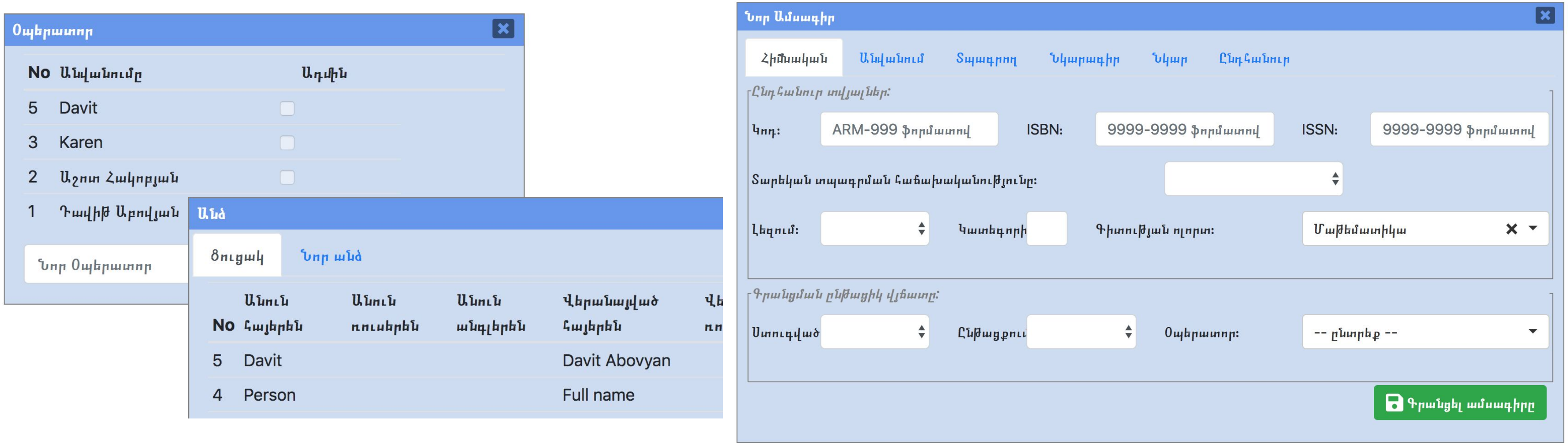
Մուտք համակարգ

Օգտվող — օգտվող

Գաղտնաբառ — Գաղտնաբառ



Prototypes (screenshots)



Օպերատոր

No	Անվանումը	Ադմին
5	Davit	□
3	Karen	□
2	Աշոտ Հակոբյան	□
1	Դավիթ Աբովյան	

Նոր Օպերատոր

Անձ

	Անուն	Անուն	Անուն	Վերանայված	Վե
No	հայերեն	ռուսերեն	անգլերեն	հայերեն	ռո
5	Davit			Davit Abovyan	
4	Person			Full name	

Նոր Ամսագիր

Ընդհանուր տվյալներ:

Կոդ: ARM-999 ֆորմատով

ISBN: 9999-9999 ֆորմատով

ISSN: 9999-9999 ֆորմատով

Տարեկան տպագրման հաճախականությունը:

Լեզում:

Կատեգորի

Գիտության ոլորտ: Մաթեմատիկա

Գրանցման ընթացիկ վիճակը:

Ստուգված

Ընթացքու

Օպերատոր: -- ընտրեք --

High level tech approach: RESTful application

Benefits are as follows:

- When extending application usage from browser to desktop and/or mobile no change will be made in back-end.
- External integrations with the system will be available,
- Full separation of front-end from back-end brings to more clear code which is easier to maintain.
- Further development will require development team instead of single developer in which case no full stack developer will be required, work will be easily divided between developers with more concrete specification eliminating lots of problems raised on teamwork.
- Later code refactoring due to change in technologies will affect only half of the code base in particular point of time.

High level tech approach: Methodology used

Agile is used as a software development methodology as it is expected that requirements will be changed during the development and the list of features to be implemented during this project can be extended or reduced depending on actual progress.

Two sprints is expected each with two month duration:

- First sprint for research on technologies, tools, libraries to be used and their testing,
- Second sprint for actual implementation.

High level tech approach: Version Control

GitHub is used as a version control tool, as it is already an industry standard, it is free (unless private repo is needed)

High level tech approach: Encoding

UTF-8 encoding is used to support Armenian and Russian unicode. It is considered in database, Spring servlet as well as in web view.

Back-end tech approach: Programming language

Java (version 8) is used as a programming language for back-end as it is considered as an efficient (time and memory) and effective (writing code) language for web applications, it is well known (easy replacement of developer, reach and stable libraries)

Back-end tech approach: Framework used

Spring framework is used, as being free and the most popular Java framework for web applications. In Spring for setups XML is used and for the rest annotations. Besides Spring main tools, Spring security is used for authorization and authentication, JDBCTemplate is used for work with database.

Back-end: Build/dependency control tool

Maven is used as a build and dependency control tool, it is effective, well known and popular tool.

Back-end: Integrated development environment (IDE)

Intellij is used as and IDE as it is well developed and have very powerful tool. It works fine with Spring, mySQL, Git, Maven. It is not free but education purposes it provide free license.

Back-end: Coding style used

- Naming - for variable and method name camelcase is used with lower case first character, name should include the description of the field with max limit of five words, constants are all in upper case with words divided by dash.
- Commenting - JavaDocs style is used for commenting methods signature, single line comments are used otherwise in any place where additional information will be useful.
- Code cleanup - “*Later means never*” is the slogan used, code is being cleaned during its initial writing process (at least tried to do so).

DESIGN APPROACHES: Application layering

- DAO (data access object) layer - all functionality related to work with database (CRUD operations) is done within this layer. Each DAO layer is implemented for a particular type of database (MySQL in our case). All the other layers works with database only through this layer.
- Service layer - all business logic is implemented here, it uses DAO layer to work with database, as well as service layer for other objects.
- Controller layer - this layer is the endpoint for all http request, it uses service layer to serve the requests.

DESIGN APPROACHES: Indirection in inheritance

Interfaces and several level of inheritance are used for better code reuse and fewer changes experience. In Service layer DAO layer objects are used indirectly through interfaces as in case of change or addition of other database type nothing should be changed except DAO layer.

DESIGN APPROACHES: REST API

Main requirements for RESTful services is be kept, CRUD is implemented by POST, GET, UPDATE and DELETE http methods, API support JSON object, API is versioned with no change to already provided contract, endpoints are in following the format

/api/vX.X/object_name/

DESIGN APPROACHES: Database access

For work with database Spring JDBC is used. Although it requires more code than JPA or Spring Data, but it gives more flexibility and more control. For effective use of database connects, pool is used.

DESIGN APPROACHES: Exception handling

Custom exceptions are developed in order to catch all main exceptions and rethrown in DAO and Service layers, so it can be properly handled in API responses. All try/catch blocks are eliminated from DAO and Service layers being catched in Controller.

DESIGN APPROACHES: Third party libraries

For effective manipulation of data in JSON format Gson library is used (Google implementation of JSON)

DESIGN APPROACHES: Logging

In order to track and analyze unexpected behavior of already deployed web application logs are being recorded using log4j library

DESIGN APPROACHES: Testing

To ensure that continuous integration of new features into application and bug fixed don’t introduce new bugs automation tests are developed. For unit and integration testing Junit library is used

DESIGN APPROACHES: Application deployment

Tomcat is used as an application server, it is free and feets Spring well. The back-end part of application is deployed in a hosted virtual machine together with MySQL server

Database tech approach

Database used:

MySQL database with InnoDB engine is used as it is free and fully satisfy all technical requirements for such web application, it is very popular and well developed.

Naming:

For field/table naming dash is used to separate words, index name ends with _idx, and foreign keys with _fk.

Other aspects:

Data integrity are assured in database level using NOT NULL and UNIQUE constraints and default values. Predicted limit of characters for VARCHAR fields is used, otherwise maximum.

Front-end tech approach: JavaScript framework

Angular 4 with TypeScript is used as a front-end framework, it is well developed and very popular and lots of libraries are developed for free use. For deployments it is compiled to relatively small minified files. Angular 4 has some limitations but is much easier to develop than ReactJS which is the main competitor with Angular.

Front-end tech approach: Front-end library

Bootstrap 4 is used for quick formatting of web view, although it supports responsiveness, but the way Angular 4 is used in web application, such responsiveness mainly loss its credits. HTML5 and CSS3 are used in Bootstrap 4

Front-end tech approach: Font instead of image

- It is light weight;
- It doesn’t need additional http requests to server (browser could make very limited number of request simultaneously);
- In scaling it doesn’t lose any accuracy;
- It can be placed in any place where text is allowed and the color change is as easy as for text.

Front-end tech approach: Cascading style sheets

LESS technology is used to generate css files. It has some kind of object oriented features such as inheritance, variables for code reuse, methods. It could be used for view customization.

Development code

1. ***Back-end part***

   Github: https://github.com/davit-abovyan/bibliographic/tree/master-REST

2. ***Front-end part***

   Github: https://github.com/davit-abovyan/bibliographic_front