ISSN: 0975-8585

Research Journal of Pharmaceutical, Biological and Chemical Sciences

Social Time Management Web Service Development.

Aleksey Andreevich Kuznetsov*, Valeriy Zalmanovich Magergut, Inessa Andreevna Kochetkova, and Marina Aleksandrovna Kosonogova.

Belgorod State Technological University named after V.G. Shukhov, Russia, 308012, Belgorod, Kostyukova Street, 46.

ABSTRACT

We consider the urgency of a social time management web service development. We consider the problems arising from the choice of technology for this project implementation. The stages of service development and testing are developed including its commissioning. Also the problem of time shortage and time management was considered as the main way to solve these problems. The concept of social time management was revealed and the possibilities of information technology in the field of time management and human resources organization was explored. The result of this analysis is the decision to develop a new web service. The implementation is divided into several key points: preparation, choice of technologies, the development of each part of the project, integration, testing and commissioning. Each of these points is analyzed and described separately. The particular attention is paid to the choice of technology and architecture for the project. The result of the project implementation is a full-fledged social time management web service, allowing users to solve the problems of organization and time management with each other.

Key words: web service, time management, social network, development, design, databases, information technologies, web technologies, real-time systems

**Corresponding author*

INTRODUCTION

The tempo of human life and social activity increase, the increase of time spent together with the community, not only lead to positive results in various areas of life, but also create some problems, one of which is the lack of time - one of the main non-renewable resources [1,2]. In this regard, there is a need to organize and manage time which the time management [3], the technology that allows the efficient use of time resource [4]. However, apart from personal time planning, the person also needs to the organization of social time i.e. the time that he spends in the community or in dependence on it, resulting in the need to use social time management. For the realization of social time management the communication between its members is needed primarily. And the information technologies here are manifested itself in all its glory [5]. A pretty vast part of communication and people relations occurs at the use of such technologies. Social network is a good example of such technologies.

Since the modern information technologies [6] do not have some ready-made solutions fore the social time management issue, limiting only by organizers as a means for time management and social networks, email and instant messaging as a means of communication, then the necessity of new service design, implementation and commissioning appears, which allows to realize the functions of social time management fully, combining the organizers and social networks functionality (see Fig. 1).

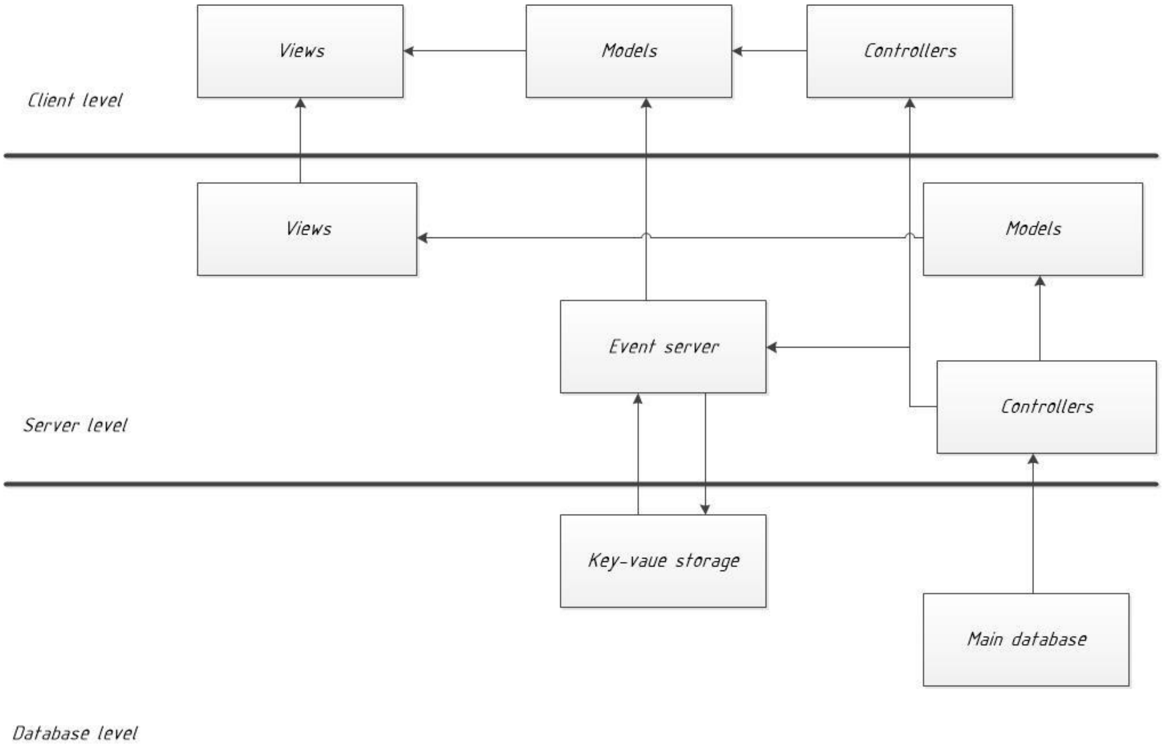

Figure 1: Project architecture

MAIN PART

The project implementation is conditioned by the following actions:

- Work environment preparation, necessary software installation,
- Choice of the project technology,
- Database development,
- Web framework modelling,
- Development of web framework controllers,
- The creation of web framework representations,
- Event server development,
- Client framework controller development,
- The creation of client framework models,

- The creation of views for the client framework,
- The system components integration into a single unit,
- Alpha-testing,
- Beta-testing,
- Commissioning.

Preparation of the working environment is the selection of the tools and technologies necessary for the further development. Let's choose the modern version of Ubuntu 13.04 as the operating system as UNIX-like systems are the most preferred ones for web development. The main application means necessary in the course of development, will be the browsers that are in the system repositories, the terminal preinstalled in it, text editor Vim. Let's install additionally the following utilities, most of which are present in the standard repositories of Ubuntu 13.04 operating system:

- Git - to provide the work with the control system Git. Let's use github or bitbucket to host the project.
- Wget - the utility for files downloading within the network,
- Ssh - the utility for SSH protocol use,
- Build-essential - a set of programs and utilities to compile and build the necessary technologies in the future.

The selection of technologies is one of the most important points for the project implementation. As follows from the architecture, it is necessary to select:

- Web framework, corresponding to MVC model,
- The technology to implement an asynchronous event server,
- Client framework, corresponding to MVC model,
- A database with the replication property,
- Key-value storage type.

It is necessary to choose a modern and functional framework as an MVC web framework with a set of tools necessary to implement a complete Web application that works with a large number of web-based technologies large variety (see Fig. 2).

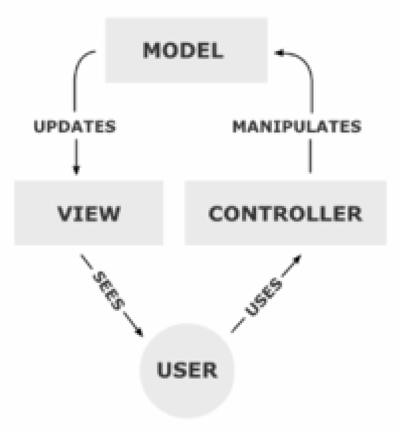

Figure 2: MVC model

One of these frameworks is RubyonRails [7], that is gaining popularity now due to its architecture, openness, and a large number of modules - gems, simplifying the process of Web applications development and allow to implement the most complex functions in the project. The main advantages are RubyonRails is the strict adherence to the Model, View, Controller architecture [8], which allows to organize the business logic for various projects in the object-oriented view, a great community and a developed ecosystem, providing the majority of modern web technologies and tools use, the opportunity for a clear integration in a web application of a wide variety of other information technologies, which make this framework one of the leaders in its field.

As a technique for asynchronous behavior implementation let's choose Node.js - a high-performance server implementation of the Javascript language based on V8 engine [6] to generate the distributed scalable network applications. The advantages of this technology is an asynchronous operation, the speed provided by V8 engine, one of the most popular programming languages in the world - Javascript - as the core one. The disadvantages of this technology are the low stability of the software interface that is based on the rapid development cycle of the project, and certain weaknesses in the implementation of error processing. Let's use the library socket.io, which provides an asynchronous connection with the server through various types of sockets to implement the server communication with a client, a Web application and a key-value storage.

AngularJS - Javascript-framework will serve as a client framework with an open-source from Google developers, extending the browser applications based on MVC pattern. The main difference of AngularJS is the use of imperative programming for the logic description (as opposed to the declarative approach in JQuery), which makes the application much easier to read and reduces the server load. The framework adapts and extends the traditional HTML to provide a two-way binding for dynamic content, which automatically synchronizes the model and the presentation and may be used as updates in real time. It is also worth noting the framework possibility to work with REST-backend that should be used in the project being developed.

By selecting the database one should be based on the earlier designed data models. These data may show that a large number of data within the developed service are nested into each other and are not linked with dependent relationships. In this situation, it is reasonable to suggest the use of non-relational document oriented databases, which would have coped well with complex structured data (see Fig.3).

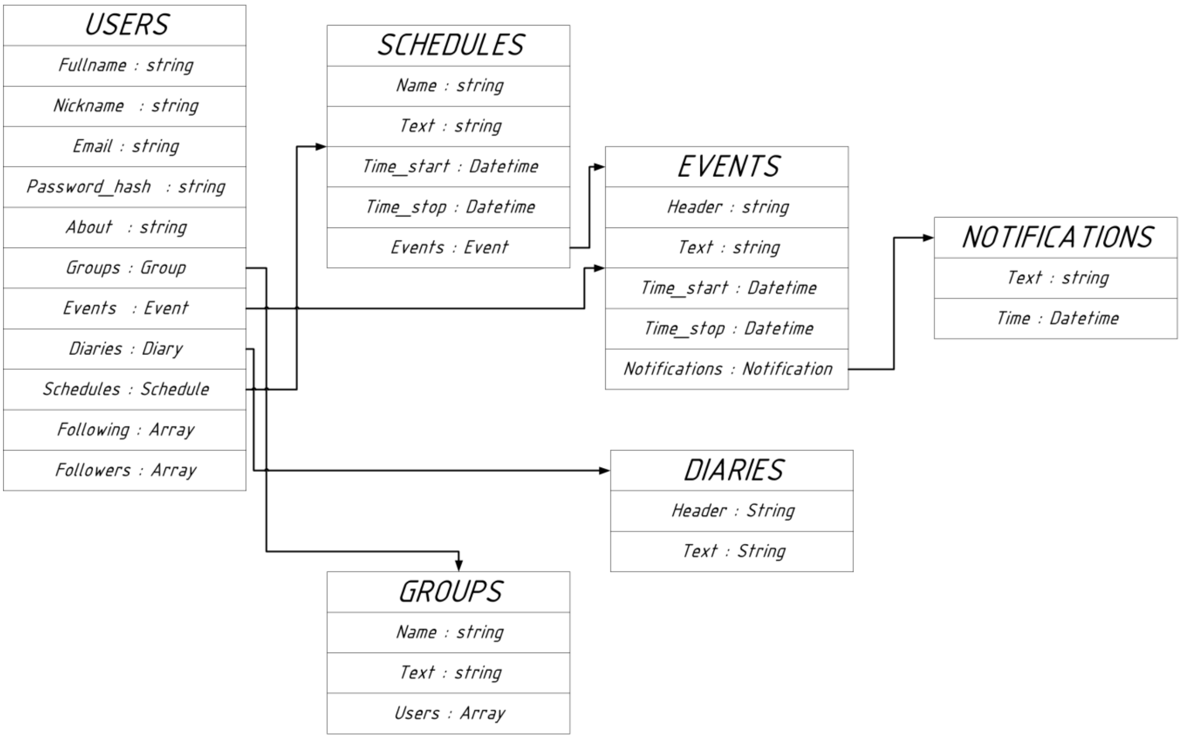

Figure 3: Nested data management by MongoDB DBMS

One of the best non-relational NoSQL is a document-oriented database management system MongoDB, which presents an opportunity for the organization of the most complex nested data structures, the tools for replication, archiving and scaling to distribute the increased load between different databases, and also offers a large number of drivers for DBMS operation for quite a large number of web-based technologies, including the selected earlier RubyonRailsi Node.js [9].

The most preferred storage variant of a key-value type Redis database (Fig. 4), keeping the data into RAM and recording them on a disc after some periods of time [10], thus providing the speedy operation with the data, which is comparable to the speed of RAM memory that allows you to use Redis in the cases when the performance is critical one. Another advantage is the availability of client modules in most programming languages to work with this database.

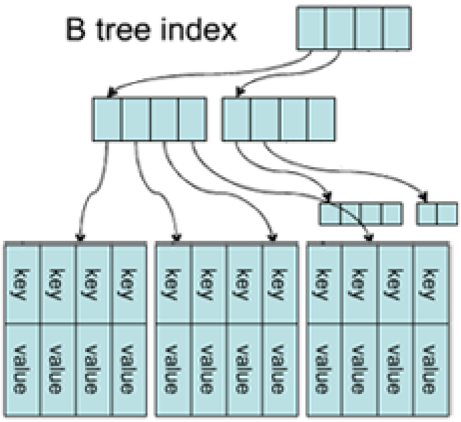

Figure 4: Scheme of key-value stores

The creation of databases is performed to install MongoDB and Redis, and then its filling in, the setting of access rights, the authentication and authorization of users, the installation of backup databases.

The model development in RubyonRails involves the generation and editing of abstractions from the database corresponding to the previously designed database models. The main models of web application are:

-user – user model,
-event – event model,
- Notification - notice
- Diary - diary entry,
- Group - the uniting of members in a group,
- Schedule - timetable.

Besides the direct design of the data structure model one must also provide the input data validation, the filtering, and to use the model lifecycle objects (callbacks and observers). RubyonRails web framework involves the placing of most part of the application logic in the models rather than in the controllers.

When you create a controller it is necessary to develop CRUD (Create, Read, Update, Delete) operations for each of them and provide for the possibility of these operations through JSON and design RESTAPI in such a way. The creation of views involves the creation of templates and data mappings for a Web application. The representations will reflect the data from the controllers and models in the user's browser directly. Besides, you need to create presentation Pages to display the static pages with constant information.
The development of the event server is used to create a new NodeJS application and launch the socket server socket.io. After that, any client application may communicate with the server via sockets. It is necessary to provide user authentication using the session data (cookies) and to store them in a database of key-value type if necessary. The basic function of the event server is the data update in real time. To do this, you must send a request to a server each time when they are changed. Then the server determines the user's subscribers and sends updates or changes. Thus it turns out that all the events for the subscribers are added, removed, modified in real time. Also, the event server should arrange the delivery of notifications to customers. To do this, a client must send a request to a server that calculates the time-to-event and up to this moment stores the data in the key-value notification database at each notification action. Once it is time to make an alert, it is passed to all client devices of a user, whether they are browsers, mobile or desktop applications.

The possibility to alert users shall be provided not only by the means of browsers and client applications, but also be by e-mail, SMS-messages at the user's choice. When notifications are implemented in the browser one should pay attention to HTML5 Notifications. The development of models, controllers and views of the customer service in general is similar to the development of a server part with the difference that a server level abstraction occurs and not the database level abstraction. The communication between the client and server level is due to JSON requests to the already developed RESTAPI. The system integration into a single unit is the overall design of various project parts of and the linkages between these parts. The main points of contact between the project parts should be a programming interface of the web application (RESTAPI), allowing obtain server data for the client web framework and for other client applications and server events, whose task is to update the data in real time and organize the operation of alerts. It is necessary to conduct thorough testing of individual components after this point as well as a web service as a whole, then the latter must be ready for commissioning.

CONCLUSION

The result of this work shall be a full-fledged functional web-based service that allows the user to take full advantage of social time management for his own purposes: to create and organize events, to tell others about them, to receive alerts about events, to keep a dairy, to look for interesting information, to make schedules, to filter the information by date and users quickly. The design and the selected technologies allow to perform the project scaling and data clustering for a large number of servers in the case of load increase on the service.

SUMMARY

The project development and its implementation allows all Internet users to manage their own time together according to the plans and events of other users to carry out a social time management.

REFERENCES

[1] Yudin D.A., Magergut V.Z. 2013. Firing process image segmentation with the use of texture analysis based on self-organizing maps. Information Technologies. 5: 65-70.

[2] Kosonogova, M., Ivanov, I.,2013. Method of as sessing the level of student's straining for the purposes of adaptive electronic training. In the 2ndInternational Conference of Informatics and Management Sciences of the 2013, CD-R.

[3] Morozova O.F. 2010. Time management of the social management system. Bulletin of the Tomsk State University. 334: 35-39.

[4] I.A. Kochetkova, Dovgal V.M. 2011 Tools of multidimensional images recognition, diagnosis and prognosis. Innovative materials and technologies: The collection of Intern. scientific-practical conf., pages 287-290.

[5] Shaptala V.G., Radoutskiy V.Y., Shaptala V.V. 2011 Application of neural network technologies for the prediction of emergency situations. Information tehnologies. 2: 60-65.

[6] Ledeneva T.M. 2011 About the calculation of the similarity functions for fuzzy numbers. Bulletin of Belgorod State Technological University named after V.G. Shukhov, 4: 35-37.

[7] D. Flanagan, Y. Matzumoto, 2008. The Ruby Programming Language. O`Reilly Media, pp: 81.

[8] D. H. Hansson, D. Thomas, L. Breedt, M. Clark, J. D. Davidson, J. Gehtland, 2013. Agile Web Development with Rails. Pragmatic Programmers, pp: 19.

[9] K. Chodorow, M. Dirolf, 2010. MongoDB: The Definitive Guide . O`Reilly Media, pp: 25.

[10] J. L. Carlson, 2013. Redis in Action. Manning Publications, pp: 78.