

# System Supporting Planning and Management of Time and Cost of Projects Based on Java EE Platform

Szymon Kubicz, Przemysław Nowak, Michał Wojtera, Jarosław Komorowski, Bartosz Sakowicz

**Abstract** – The aim of this article is to present a management supporting system which follows unique project management methodology. The article shows how to integrate different frameworks, libraries and technologies working on Java Enterprise Edition platform in order to create fully operable and very useful internet application.

**Keywords** – Java EE, JAVA, JSF, SPRING

## I. INTRODUCTION

In the era of advanced technologies and continuous rapid development of civilization, few things became extremely important: hard work and also good planning and coordination of that work.

Therefore over the past years began to appear more and more project management tools similar to the system described in this article [7,12]. The implementation methodology uses technologies based on Java EE platform, such as JSF or Spring. Thanks to them it was created functional website which is helpful for everyone involved in the project life cycle. With the interactive Gantt's graph, system allows intuitive scheduling and comfortable project management.

## II. PROJECT MANAGEMENT

Project is a unique sequence of tasks undertaken with the aim achieves unique objectives within a specific timeframe [4]. The key features of the project are:

- Aim
- Finite duration
- Uniqueness
- Element of uncertainty and risk
- Distinct structural

Project management is striving to achieve specific aim, remaining within the prescribed time, cost and reaching at the

---

Manuscript received November 09, 2011.  
Katedra Mikroelektroniki i Technik Informatycznych  
ul. Wolczanska 221/223 budynek B18, 90-924 Lodz, POLSKA  
al. Politechniki 11, 90-924 Lodz, POLSKA  
NIP 727-002-18-95  
tel. +48 (42) 631 26 45 faks +48 (42) 636 03 27

beginning a complex of final product quality. The measure of success in project management is: range – means how many objectives succeeded, quality – are customers pleased, resources – did appear losses in team or deterioration in team relations. Relations between individual successes meters can be presented in graphical form using the so-called project management triangle (Fig.1) [5].

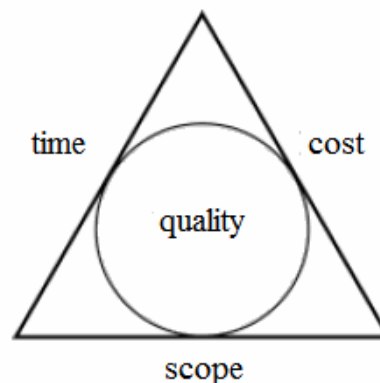


Fig. 1. Project Management Triangle [4]

## III. TECHNOLOGIES USED IN APPLICATION

Applications based on Java Enterprise Editions, although they require more work and attention, they offer number of interesting features such as standard formation, scalability, portability. These features have convinced many companies which produce web-based software.

Presentation layer in described system was made with usage of Java ServerFaces Framework. JSF has many values, which meaningfully facilitates implementation of user interface, few of them are introduced below:

- predefined interface components,
- event driven programming model,
- model components, through developers can create their own components and reuse them in many projects,
- usage of MVC (Model View Controller) design pattern.

To implement a persistent layer Hibernate framework has been used, which is the one of the most popular tools served to object relational mapping. Additionally Spring framework was used, which provides support in all stages of application design. Moreover it allows for easy integration with specialized frameworks such as JSF and Hibernate.

### III. SPRING SECURITY

Spring Security requires creation of database tables according to scheme introduced on Fig.2 [9-11].

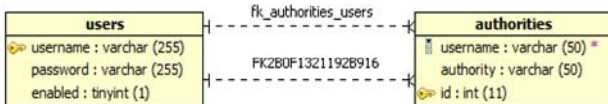


Fig. 2. Spring Security – database tables

In configuration file must be defined object responsible for access to the database. Then it is necessary to configure authentication-manager, which is responsible for authorization. It needs to be transmitted references to object authentication-provider, which is responsible for delivering to users, including their roles. In addition authentication-provider can be determined about the way of encoding passwords. Configuration of authentication-manager presents Fig.3.

```
<authentication-manager alias =
"authenticationManager">
  <authentication-provider>
    <password-encoder hash="md5"/>
    <jdbc-user-service data-source-ref =
"dataSourceSecurity"
  users-by-username-query="SELECT username,
password, enabled FROM users where username=?"/>
  </authentication-provider>
</authentication-manager>
```

Fig. 3. Spring Security – authentication-manager

Subsequently must be determined login page and access rights to specific resources.

### IV. SPRING - HIBERNATE

Configuration Spring – Hibernate is limited to one configuration file. In described application the file is spring-context.xml. Other part of configuration has been moved to Java classes using annotation mechanism. Setting up the annotation was made possible by adding an entry in configuration file (Fig.4):

```
<context:annotation-config />
```

Fig. 4. Spring – Hibernate – enabling annotations mechanism

and determining package, in which Spring will search objects which contains annotations (Fig.5).

```
<context:component-scan base-package="com.thesis">
  <context:include-filter type="annotation"
    expression= "org.springframework.stereotype.
Repository"/>
</context:component-scan>
```

Fig. 5. Spring – Hibernate – component-scan.

Next step is to configure object containing bearings of database and determining how to access it (Fig. 6):

```
<bean id="hibernateDataSource"
class="org.apache.commons.dbcp.BasicDataSource"
destroy-method="close">
  <property name="driverClassName"
value="com.mysql.jdbc.Driver"/>
  <property name="password" value="szym"/>
  <property name="url"
value="jdbc:mysql://localhost/temp?characterEncoding
=UTF-8"/>
  <property name="username" value="root"/>
</bean>
```

Fig. 6. Connection with data base.

If the application uses Spring framework, and the role of deliverer of the Java Persistence API belongs to Hibernate then must be determine so-called vendor adapter object (Fig.7).

```
<bean id="vendorAdapter"
class="org.springframework.orm.jpa.vendor.
HibernateJpaVendorAdapter">
  <property name="showSql" value="false"/>
</bean>
```

Fig. 7. Spring – Hibernate – vendorAdapter

The last step of configuration Hibernate using Spring is to determine object of session factory (SessionFactory). It is Hibernate object serving creation of session (objects of Session type), which in turn manages connection data.

Then it is necessary to create DAO (Data Access Object), which will deliver uniform interface for communication between Java objects and database. In implementation of interface it is necessary to provide annotation which will give information about DAO class for framework (Fig.8).

```
@Repository("projectDAO")
```

Fig. 8. Spring – Hibernate – annotation Repository

DAO class should also inherit from HibernateDAOSupport class. It is an object which makes available the whole set of methods for handling data access. Usage of HibernateDAOSupport allows programmer to ignore all problem related to transactions and sessions management. However, in order to work properly, to the HibernateDAOSupport has to be delivered session factory,

which was set up earlier. For this purpose it can be used **Autowired** annotation, which causes that **SessionFactory** object is 'injected' automatically (Fig.9).

```
@Autowired
public void init(SessionFactory sessionFactory){
    setSessionFactory(sessionFactory);
}
```

Fig. 9. Spring – Hibernate – injection of session factory to DAO class

With DAO object programmed it is possible to access the application in the easy way (Fig.10).

```
@ManagedBean("principalBean")
@Scope("session")
public class PrincipalBean {
    private ProjectDAO projectDAO = null;
    @Autowired
    public PrincipalBean(@Qualifier("projectDAO")
    ProjectDAO project) {
        projectDAO = project;
        getPass();
        init();
        currentStock =
stockList.get(currentStockIndex);
        getProjectByPrincipal();
    }
}
```

Fig. 10. Spring – Hibernate – DAO object in use

This sample source code presents exemplary 'inject' of an object implementing interface **ProjectDAO**. Usage of reverse control design pattern, application can operate on the same interface in complete separation from specific implementation [6]. Object implementing **ProjectDAO** interface provides methods which allow for database operation. The database schema used in application is shown on Fig.11.

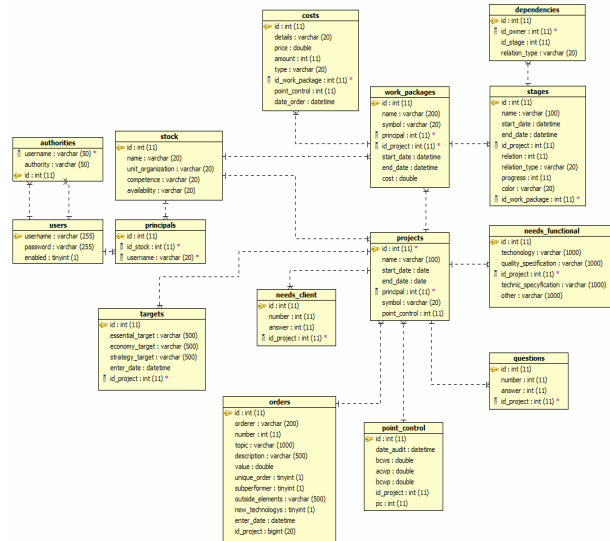


Fig. 11. Database schema

## V. ANNOTATIONS

Programming annotations gives a lot of possibilities of including additional information directly in the source code. So far, it was not possible. Until annotations became available for Java programmers, all configuration has to be done in external XML files. For example, configuration of JSF beans have been done in the **faces-config.xml** and it looked as follows [1,8]:

```
<managed-bean>
    <managed-bean-name>principalBean</managed-bean-name>
    <managed-bean-class>com.thesis.principal.PrincipalBean</managed-bean-class>
    <managed-bean-scope>session</managed-bean-scope>
</managed-bean>
```

Fig. 12. Traditional bean initialization in JSF

Due to annotations programmer can configure exactly the same features in application by adding designation directly in interesting class of source code (Fig.13).

```
@ManagedBean("principalBean")
@Scope("session")
public class PrincipalBean {
    ...
}
```

Fig. 13. Initialization JSF bean with annotation

Another example of convenient usage of annotations are **POJO's** classes used by **Hibernate** for object-relational mapping. It allows to get rid of enormous number of **.hbm** configuration files, and to locate directly all of configuration in proper classes.

## VI. GANTT CHART

One of the key tools which described applications offers is interactive Gantt chart (Fig. 14).

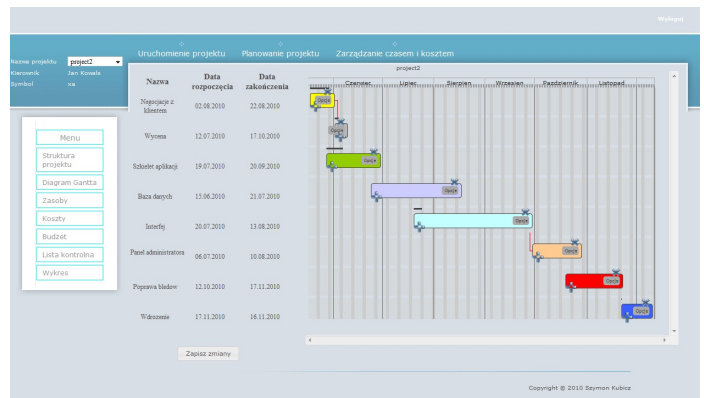


Fig. 14. Gantt Chart

## REFERENCES

The graph's engine is programmed due to jQuery library. The information needed to generate it are taken from database. The whole object, which represents the project, is transformed to a JSON format. Followed by corresponding field, announcement is retrieved by browser part of application.

## VII. FUNCTIONALITY OF THE SYSTEM

The presented web-based system certainly can be useful for managers who want to organize and sort the projects which were undertaken by a company. It provides support for three main stages of project management:

- starting – classification of orders, establishment of a manager, setting objectives, customer needs and functional needs,
- planning – structuring and scheduling the project, creating budget, graphical representation of project costs
- management of time and cost – creation of control points, controlling the time and expenditures by using earned value method.

Additionally, due to security restrictions mechanism, all information are available only for authorized persons.

## VIII. CONCLUSION

The described application was created in order to support project managers in their work. The proposed system is ready for use during common work with the project. Furthermore, it is easy and convenient to use. Gantt chart is an attractive graphical tool for manipulation of the project schedule.

The fact that the application was made by agile application frameworks such as Spring makes its implementation ease readable and ready for future enhancements. As a deployment environment, any Java EE compliant application server can be used (e.g. Tomcat).

The system can be also easy extended both in terms of its functionality (to ensure compliance with all agreed methodology) as well as in technical features

## ACKNOWLEDGEMENTS

The authors are a scholarship holders of project entitled "Innovative education ..." supported by European Social Fund.

- [1] Eckel Bruce, Thinking In Java, Helion S.A., Gliwice 2006
- [2] Geary David, Horstmann Cay S., Core JavaServer Faces, wydanie II, Helion S.A., Gliwice 2008
- [3] Johnson Rod, Hoeller Huerger, Arendsen, Risberg Thomas, Sampaleanu Colin, Spring Framework – Profesjonalne tworzenie oprogramowania w JAVIE, Helion S.A., Gliwice 2006
- [4] Mingus Nancy, Zarządzanie projektami, Helion S.A., Gliwice 2002
- [5] William R. Duncan, A Guide To The Project Management Body Of Knowledge, PMI Standards Committee, Project Management Institute, PA 19082 USA.
- [6] M. Zywno, B. Sakowicz, K. Dura, A. Napieralski "J2EE Design Patterns Applications" 12 th International Conference MIXDES 2005, Kraków, Poland, 23-25 June, pp. 627 - 630, vol. 1, ISBN 83-919289-9-3
- [7] Wojtera M., Sakowicz B.: „Web Application for Project Management Based on Open Source Solutions”, 13th International Conference Mixed Design of Integrated Circuits and Systems MIXDES 2006, 22-24 czerwca 2006, Gdynia, wyd. KMiTI, str. 797-800, ISBN 83-922632-9-1
- [8] Karolina Czekalska, Bartosz Sakowicz, Jan Murlewski, Andrzej Napieralski: "Hotel Reservation System Based on the JavaServer Faces Technology", 9th International Conference Modern Problems of Radio Engineering, Telecommunications and Computer Science, TCSET'2008, 19-23 February 2008, Lviv-Slavsko, Ukraine, ISBN 978-966-553-678-9
- [9] Szymon Gradka, Bartosz Sakowicz, Piotr Mazur, Andrzej Napieralski: "CRM system with behavioral scenarios based on Spring framework", 9th International Conference Modern Problems of Radio Engineering, Telecommunications and Computer Science, TCSET'2008, 19-23 February 2008, Lviv-Slavsko, Ukraine, ISBN 978-966-553-678-9
- [10] Marcin Mela, Bartosz Sakowicz, Jakub Chlapinski: "Advertising Service Based on Spring Framework", 9th International Conference Modern Problems of Radio Engineering, Telecommunications and Computer Science, TCSET'2008, 19-23 February 2008, Lviv-Slavsko, Ukraine, ISBN 978-966-553-678-9
- [11] PILICHOWSKI M., SAKOWICZ B., CHLAPIŃSKI J.; "Real-time Auction Service Application Based on Frameworks Available for J2EE Platform", pp. 166-169, Proceedings of the Xth International Conference TCSET'2010, "Modern Problems of Radio Engineering, Telecommunications and Computer Science", Lviv-Slavsko, Ukraina, 23-27 February 2010, s.380, A4, wyd. Publishing House of Lviv Polytechnic National University 2010, ISBN 978-966-553-875-2
- [12] Jamroz, A.; Zabierowski, W.; Napieralski, A.: "Work time management in a small company as example of usage the web technologies" CAD Systems in Microelectronics, 2009. CADSM 2009. 10th International Conference - The Experience of Designing and Application of; 2009, ISBN 978-966-2191-05-9