Online Project Management for Dynamic e-Collaboration

Lucia RUSU, Vasile RUSU
Babeş-Bolyai University, Cluj Napoca
lucia.rusu@econ.ubbcluj.ro, vasi2587@yahoo.com

Today's collaborative projects demand efficient and productive software application tools for the workplace that will bring remote teams together to get the work done. Dynamic e-collaboration is a necessity for virtual relations and business agreements. It depends on two distinct factors: trust and need. This paper presents a way to manage remote teams using a web application developed with ColMap model of project management in an IT company. The information exposed and shared applications with partners in collaborative projects are based on RBAC. Group collaboration and management software has been proven to successfully manage and coordinate projects.
Keywords: Dynamic E-collaboration, Collaboration Model, Web Application

1 Introduction

Nowadays, in crises context a many companies to survive or maintain market position and customers based on suitable management and proper solution, which involves short term collaboration. Collaboration involves flexible relationships between business partners and offer joint work, usually based projects are carried out on the short or medium term. New relationships are created almost ad hoc, and are abandoned as quickly as organizations that are distributed after the purposes and responsibilities involved in business operations have disappeared.

The term *eCollaboration* is necessary to develop a new IT framework for collaborative Web applications that supports this kind of dynamism that address opt-in and opt-out collaboration. This concept occurred in close connection with the use of meaning-based Internet technologies to enable continuous exchange, automated information between suppliers, consumers and intermediaries [1]. *Dynamic e-collaboration* became not an IT fashion, merely a necessity for virtual relations and business agreements. It depends on two distinct factors: *trust* and *need* has a great influence in the information exposed and shared applications in collaborative projects [2]. These factors established an appropriate IT infrastructure is provided to facilitate current sharing process point of view because of the IT support of traditional eCollaboration when short application borne by Dynamic eCollaboration.

Major change is that while short-term projects and participation in them is almost ad hoc, when quantum of time is compressed and therefore the impact of fluctuations in the level of confidence and needs (and Need Trust) is placed exaggerated. In this case appear new requirements and orders which require IT infrastructure to support dynamic collaboration efforts [3].

This paper is structured as follows: after an introduction about dynamic e-collaboration in virtual organizations, section 2 focused on project management in virtual organization. We concern on Collaborative project management and related works in online Project Management tools and dynamic e-collaboration model. In section 3 we present EasyProject –a dynamic collaborative prototype using ColMap approach and three layer architecture. We start with framework model and functionalities, than we point Authentication and Authorization mechanism, Data Base and Data Access Layer, the Business Logic and Presentation Layer, User Interface and Gridview Control. Section 4 contains conclusions about main benefits of EasyProject and the future work.

2 Project management in virtual organization

2.1 Collaborative project management

It has now been widely proven that modern project management can help an organization become more efficient, more productive and more competitive in the global economy. It therefore seems logical that if more organizations within specific industries embrace PM best practices, then those industries will become more productive and competitive. If organizations and industries within regions, states and countries become more efficient and competitive, then local economies will benefit. [5]

By using project management software, project managers can view their project status in

seconds. No longer do they have to create complex reports and use spreadsheets to determine where they are at. By simply clicking on the project dashboard, a user can see if the project is on time and resourced accordingly. Web-based software can surprisingly increase performance, productivity and efficiency within an organization [6].

Since web-based applications can be accessed through any web browser, no desktop installation or updates are required. Web-based applications require to be installed on a server, which is most of the time hosted by the software developer. Moreover, certain providers even offer Intranet solutions, which can be installed on the own server [7]. By using a web-based project management application, the members of a project can work together, access resources and complete their tasks from anywhere in the world.

Project management is a standard by ANSI standard by ANSI / PMI 99-001-2000, proposed by the Project Management Institute (PMI). Project addressing in terms of lifecycle, distinguish following phases: initiation, project management and its completion. Initiating the project is in finding an idea or goal to be achieved, initiation is usually done based on the idea offered by a customer who is called the language specialist or the project sponsor.

Documents from which it can initiate a project: the project proposal and feasibility report. The project proposal converts the idea into the details of a potential project and feasibility report is the result of feasibility study to determine if the initiative is worth continuing. These two documents are held by the project sponsor. Project management is productive phase involving the implementation of parts and involves: resources, labor, quality and risks.

Key elements of a project may be:

- *Planning*: whatever the size of the project must is clearly defined, established area of action, purpose and impact. The aim of the project should include planning and the final result and resources involved both financial and human ones.
- *Risk Management* consists in identifying, analyzing and strategy to respond to possible hazards and involves identifying, quantifying and addressing risks. Management involves monitoring results, review the results that occur during project development.
- *Resource Management* is responsive with the planning of human resources, financial and physical required for the execution of design

activities that are vital regardless of size or complexity of the project. An important feature is the monitoring and reporting budget during its development project.

- *Quality management* is the methods and procedures for project management and aims to increase safety and success of reducing the risk of failure. This enables a continuous improvement, which is provided by the project manager using quantifiable or un-quantifiable specific indicators [4].

2.2 Related works

Good Project Management is one of the keys to a projects success. There are many online project management applications on the web. Here is a brief overview of some well known online Project Management tools:

- **Basecamp** is a solution which takes a fresh, novel approach to project collaboration. Basecamp solves this problem by providing tools tailored to improve the communication between people working together on a project, as a collaborative approach.
- **activeCollab** is a project management and collaboration tool that can be set up on the own website. It have an area where project manager can collaborate with the team, clients and contractors and keep projects on track while retaining full control over access permissions and your data.
- **Zoho Projects** has same functionalities and offer features as: Plan, Track, Collaborate and Manage Projects Online.
- **Go Plan** can manage multiple projects with ease user interface. Track project progress through task management, calendaring, note sharing and issue tracking are some of the problems solved with it.
- **Teamwork Project Manager** is an easy-to-use online teamwork and project management software application that helps managers, staff and clients work together more productively online [4]

Some of the reasons why we do not use this kind of applications are: they are not free or open source, and even they usually have free versions, these are limited in terms of projects number and users number, they do not have the specific features that we need and most important reason is we cannot add new custom functionalities and modules.

2.3 Dynamic e-collaboration model

The term *eCollaboration* occurred in close

connection with the use of meaning-based Internet technologies to enable continuous exchange, automated information between suppliers, consumers and intermediaries [1].

It is necessary to develop a new IT framework for collaborative Web applications that supports this kind of dynamism that address opt-in and opt-out collaboration. Marmaridis and Ginige propose a solution of the graft peer to peer architecture for Dynamic eCollaboration (DeC), a framework that allows web applications to be exposed to business partners, offers secure solution and providing selective information sharing on the business logic. This information will be shared by end-users are easy to work and provide security at every level [2], [3].

A new type of cooperation converge in recent years to Dynamic eCollaboration that unlike e-Collaboration involves flexible relationships between business partners and offer joint work, usually based projects are carried out on the short or medium term. New relationships are created almost ad hoc, and are abandoned as quickly as organizations that are distributed after the purposes and responsibilities involved in business operations have disappeared. Dynamic eCollaboration depends on two distinct factors: *trust* and *need* has a great influence in the information exposed and shared applications in collaborative projects [2]. These factors established an appropriate IT infrastructure is provided to facilitate current sharing process point of view because of the IT support of traditional eCollaboration when short application borne by Dynamic eCollaboration.

Major change is that while short-term projects and participation in them is almost ad hoc, when quantum of time is compressed and therefore the impact of fluctuations in the level of confidence and needs (and Need Trust) is placed exaggerated. In this case appear new requirements and orders which require IT infrastructure to support dynamic collaboration efforts.

This framework was developed as a result of working with a group of small businesses willing to achieve Dynamic eCollaboration, identifying the need for an ICT framework that will allow other similar collaborative efforts [4]. It allows the exchange of information and business logic and is designed on an architecture pure peer to peer (P2P) [2], [5], for Dynamic eCollaboration, offering end-guiding user development, security and simplicity in operation. In identifying characteristics of this framework we studied how the trust and the need (and Need Trust) influence these features.

Marmaridis and Ginige propose a solution of the graft peer to peer architecture for Dynamic eCollaboration (DeC), a framework that allows web applications to be exposed to business partners, offers secure solution and providing selective information sharing on the business logic. This information will be shared by end-users are easy to work and provide security at every level [3], [4]. This model has four parts: Dynamic eCollaboration Project Orchestration, as an interface between business users and set collaborative projects, Network Connectivity and Synchronization, Unified Security and Remote Access and Information Sharing and Applications.
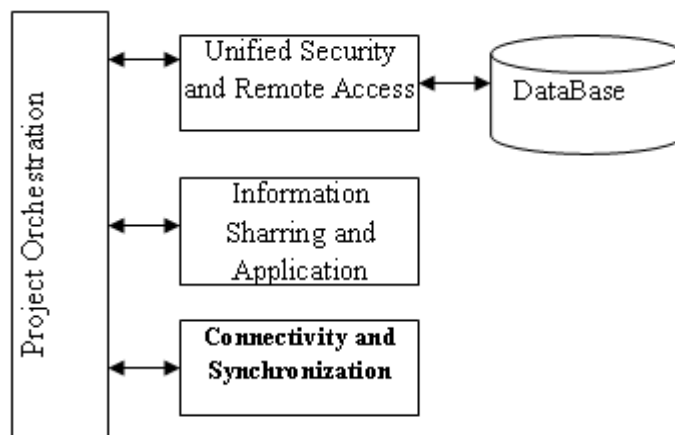

Fig. 1. Dynamic e-collaboration model

Dynamic eCollaboration *Project Orchestration* is a subsystem responsible for the interface between business users and they initiate, define and set

collaborative projects. *Network Connectivity and Synchronization* is a subsystem that allows IT systems to those establish a secure channel of

communication protocols over standard communication network. It can provide intelligence work for a number of protocols like SOAP, XML_RPC, HTTP and HTTPS or integrated to Enterprise Messaging Bus (EMB). *Unified Security and Remote Access* is a subsystem that provides a unified view of the parties and individuals involved in a collaborative project for particular time security solution, providing access over the systems and management information flow. *Information Sharing and Applications* subsystem is able to provide different views of applications that can be shared by the project while supports ad-hoc changes to the views shared throughout the project, at any moment of time.

According with Schuster and Donnan, steps that can scroll through them, identified as being necessary for a project that uses Web typical e-collaboration dynamics can be summarized as follows:

- Setting project where the stakeholders are the collaboration and use that to DeCS established secure communication channel computer-to-computer and a virtual team of users for a particular project.
- Sharing tasks between different Small and Medium Enterprise (SME) - stage where the parties are exposed parts of tasks (functions) of the web based management plan application. SMEs have relationships with customers, are able to facilitate the division's job to small parts. Once the parties agreed, decisions are recorded in the job management is shared, becoming available to all parties

involved.
- Auction for each part of the job - by a group or individual task that requires collaboration, through this mechanism ensures that the bid price is right then layout is global and final pricing [1], [4].

3 EasyProject – a dynamic collaborative prototype
3.1 Framework model and functionalities
EasyProject is an online project and task management tool which provides a virtual workspace to facilitate effective communication between team members. It helps you to share information and work jointly on projects and efficiently use all available resources. Every project manager need to manage team members and their activities to complete the project on time, and can easily assign tasks to team members with priorities and due dates. EasyProject will automatically notify team members of new tasks, inform of completed tasks and even send out reminders of past-due tasks. A complete reporting function allows monitor all phases of the project.

We propose an framework support project management in an IT company as *ColMap* (Collaborative Project Management) which seeks to cover the needs of informing team members and offers the possibility of project manager to coordinate activity in distributed manner, by allocation of tasks with different priorities and gives designers the opportunity to remotely access the working plans and to track progress of tasks and deadlines for solving them (figure 2).
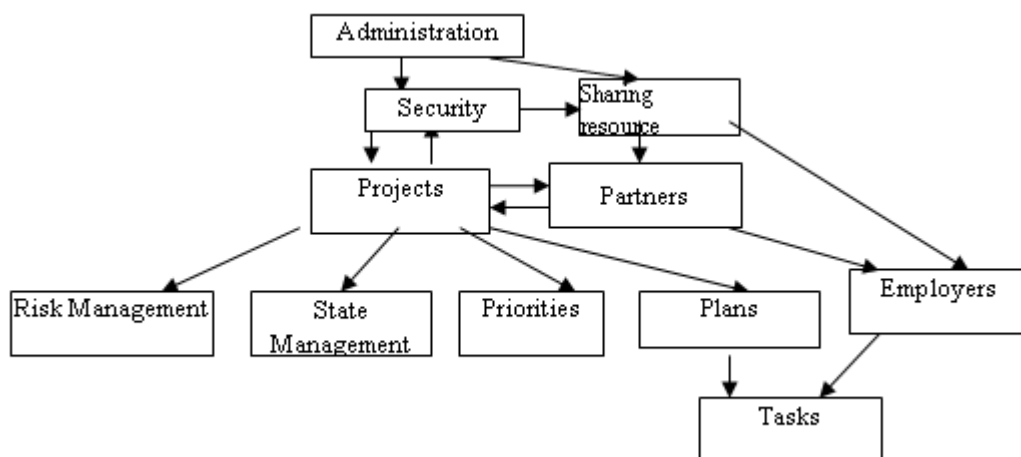

Fig. 2. Collaborative modular structure

This model offers opportunity to manage dynamic e-collaboration during the several phase of the project by *Management Partners* module, following key elements of project. We develop a

modular structure, based on DeC functional model. This module is linked with *Project Management* module and with *Employers Management*. Based on trust and need, we divide

collaborative partners and assign to several projects. A partner can be a free lancer or a SBE which can join at project execution. Project management is based on several modules with proper functionalities, started with *Plans Management, Priorities, Risk Management* and *State Management* for each project of company.

Administrations and security functionalities (*Administration, Security, and Sharing Resource*) are based on RBAC paradigm. They should highlight the link between the application and the hierarchical user logged by describing access obtained by logging (level 0 to 3). Hierarchical structure of application is successful and serves functional requirements, ensuring quick access to desired information; keep it secure and the hierarchy that determines the restrictions and access.

To develop collaborative applications also required a few functional requirements:

- a company, which is a unique business partner (customer or collaborator); a business company can have collaborative status temporarily or permanently depending on the degree of trust and its interests;
- a project manager is responsible for a project, has a team with a good composition determined; a project is characterized by a suggestive name and is composed of several plans; a plan must have a suggestive name and the file that contains it must have the same name;
- an employee may have one of these statuses: the membership team, known as the designer or may be a project manager, called the charge. Each coordinator or employee have rights and restrictions established by the RBAC (Role Based Access Control);
- any project manager or responsible coordinator nor employee is registered with a username and password unique, recording is done only by an administrator who has full access, or the hierarchically superior;
- one team member has access to tasks list and the list of plans which belong to, the plans may download;

A project manager is full responsible for a project, has access to management tasks and coordinators, that he/she allows the addition or modification and task management plans and projects, accessing the administration page for it; add the files in a plan specifically designed for this is the right of company manager or the project manager through specific project. They restrict the access of members to upload, while

maintaining a small number of users with access to confidential data, (only the company manager and project manager) which allows a very clear management plans created by the collaborative software used for virtual companies.

EasyProject is developed as an own digital workplace that improves communication between your employees spread across the globe using ColMap model and Role Based Access Control (RBAC). Features and functionalities includes: **Task Tracking, Automated Emailing, Project Reports. Task Tracking** prevents a small problem from snowballing into a big one. EasyProject has a highly sophisticated, built-in task-tracking system that monitors tasks assigned to team members and alerts the team member if a task slips past its due date. **Automated Emailing** of EasyProject has made by a fully automated 'Selective emailing system' that performs peripheral tasks such as sending appropriate email to project members, whenever required. This ensures that every member is free to concentrate on the core project-management functions. **Project Reports** gives several comprehensive project reports for all your active projects, the members of each project along with the task-related privileges each member has. The list of project parameters that you can get reports for is exhaustive.

Generally, layer refers to a conceptual separation of the application, while tier defines physical boundaries. In the discussion of layered design, you often see both the terms *layer* and *tier*. The diagram shows the 3 layers of our application (figure 3).
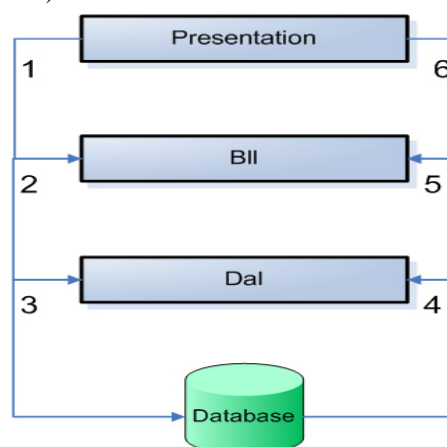

Fig. 3. Tier application approach

Layered development or multi-layer development refers to a development model where presentation, business logic and data are separated. This brings many advantages,

including cleaner code, increased maintainability and the ability to spread the application's load over multiple servers.

3.2 Authentication and Authorization Mechanism

EasyProject is an online research project management application that brings remote teams together to get the work done. Some of this application features are: users and roles management, creation of projects and task, generating reports, notifying users, assigning task to users etc. This application has 3 types of users: Administrator, Manager and User. In order to use application every user has to be authenticated. According to its role every user has specific attributes.

Administrator can create users, delete users, assign roles to users, edit users' rights and restrictions, view reports. *Manager* can create projects, view his own projects, edit details of every project, delete projects, create tasks according with project management plan, assign task to users, view reports. *User* can view assigned tasks, accept or reject tasks, resolve task and view reports.

Authentication is the process of determining the authenticity of a user based on the user's credentials. Whenever a user logs on to an application, the user is first authenticated and then authorized. The application's *web.config* file contains all of the configuration settings for an ASP.NET application. It is the job of the authentication provider to verify the credentials of the user and decide whether a particular request should be considered authenticated or not. An authentication provider is used to prove the identity of the users in a system. ASP.NET provides three ways to authenticate a user: Form, Windows and Passport authentication. In our application we have chose the form authentication [9], [10].

Forms Authentication mode is based on cookies where the user name and the password are stored either in a text file or the database. After a user is authenticated, the user's credentials are stored in a cookie for use in that session. When the user has not logged in and requests for a page that is insecure, he or she is redirected to the login page of the application. Forms authentication supports both session and persistent cookies. Authentication modes can be specified in the application's *web.config* file as shown below:

```
<configuration>
```

```
  <system.web>
    <authentication mode="Forms"/>
   <forms name="login"loginUrl="Def
ault.aspx" />
     <authorization>
        <deny users="?"/>
     </authorization>
   </system.web>
 </configuration>
```

The statement <deny users="?"> in the *web.config* file implies that all permissions are granted only to the authenticated users. The users who are not authenticated are not granted any permission. The symbol "?" indicates all Non Authenticated and Anonymous users.

The symbol "*" indicates all users inclusive of Authenticated and Anonymous users. Hence the statement <allow users = "*"> in the *web.config* file indicates that all permissions are granted to both the Anonymous and Authenticated users.

According to the type of the user, the menu displayed by the presentation layer is different. Each type of user can see after he is logged on only the pages assigned to his role. When the application is opened it can be seen only the *LogIn* page, as a personalized interface with shared resources. For example, administrator can see only the *User, UserAdmin* and *Reports pages*, the manager can see *Projects* and *Mreports* pages and the users can see only the *Tasks* page.

The following section lists the sequence of events that take place in the authentication and authorization process when a new request arrives. The IIS first checks the validity of the incoming request. If the authentication mode is anonymous (default) then the request is authenticated automatically. But if this authentication mode is overridden in the *web.config* file settings, the IIS performs the specified authentication check first before the request is passed on to ASP.NET.

Now ASP.NET checks whether Impersonation is enabled or not. If impersonation is enabled, ASP.NET executes with the identity of the entity on behalf of which it is performing executing the task. If impersonation is not enabled, the application runs with the identity of the IIS local machine's identity and the privileges of the ASP.NET user account. ASPNET or NETWORK SERVICE is the default ASP.NET unprivileged account on Windows XP and Windows Server 2003, respectively. Now, the identity that has already been authenticated and verified is used to request resources from the operating system. Then ASP.NET performs an authorization check on the requested resources and if the user is

authorized, it returns the request through IIS [10].

3.3 Database and Data Access Layer
Our database was created with SQL Server Express and placed in a special folder named App_Data. In an ASP.Net web site, a database placed in the App_Data folder is automatically added to the Server Explorer where we can add tables, stored procedures, data, functions, views. Alongside the tables generated with the ASP.Net configuration tool from Visual Studio 2005 when configuring application users and roles, we have created 3 more tables in order to store details about projects, tasks, project members.

The best choice in order to create a Data Access Layer is to use a DataSet. Technically part of the ADO.NET object library, the DataSet is one of the most fundamental and widely used data objects in the .NET framework. Used as an in-memory representation of database/XML data, it supports a plethora of handy methods, properties and contains various other object references for working with tables, rows, columns, relationships and more. Because the DataSet supports so many features and contains so many references to other objects in the ADO.NET library, it can take quite some time to figure out just what is available to you as a developer when working with this object.

The DataSet object is also directly bind able as a data source on the various GUI components that support data binding. It works very nicely with the ASP.NET DataGrid object since both objects were designed to contain data in a similar tabular format [10].

We have created our dataset in order to take full advantages of its features. The DataSet typically contains methods for accessing, modifying, inserting, deleting the underlying database data. Our database, for example, has Projects and tasks tables that record the existing projects and the tasks from the projects. In our DataSet we have a data table named Projects that contains specified methods like: *GetData()*, which will return information about all of the projects; *GetProjectsByManager(@manager_id)*, which will return the projects owned by a manager, *UpdateProject(@project_id)*, which will update the project, *InsertProject()* which will insert a new project, *DeleteProject(@project_id)* which will delete a project [11].

3.4 The Business Logic and Presentation Layer
Business Logic Layer (BLL) serves as an intermediary for data exchange between the presentation layer and the DAL and contains business rules implemented as classes. The classes are included in the App_Code directory.
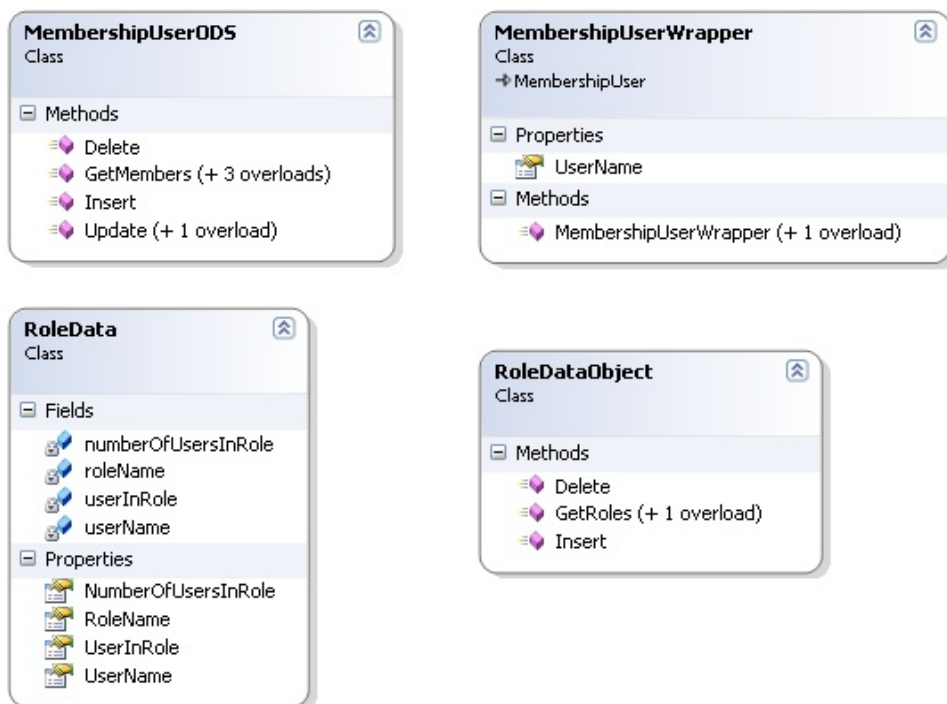

Fig. 4. Class Diagram

For instance the classes used by the administrator's user management module are (as

shown by the class viewer) depicted in figure 4.

The presentation layer is what the user actually sees. The html pages with some animation, the textboxes to take input and message boxes to alarm the user of the operation of his action. Presentation layer must be developed in such a way that it can be changed without doing any changes to the business or the database layer. This means that the presentation layer must be independent of the database and the business layers. Also the presentation layer should be made in such a way that changing the presentation to the user will take less time and effort.

In our ASP.NET application, the presentation layer or the user interface layer consists of .aspx pages. To ensure all the pages in the web site have a consistent look and feel, ASP.NET 2.0 introduces a new feature named master pages. In this approach, a common base master file that contains the common layout for all the pages is created and all the pages in the web site (called content pages) are inherited from the master page.

At runtime when the content page is requested through the browser, ASP.NET merges the output of the content page with the master page thereby providing a seamless way to reuse the master page layout and logic.

3.5 User Interface and GridView Control

With ASP.NET 2.0 *master pages*, you isolate the look and feel and standard behavior for all the pages in your application and move them to a master page. In the master page, you add placeholders (known as *ContentPlaceHolder* control) for the content pages (or child pages) to add their custom content. When users request the content pages, the output of the content pages are merged with the output of the master page, resulting in an output that combines the layout of the master page with the output of the content page [11].

The master page defines content areas using the *ContentPlaceHolder* control, and the content pages place their content in the areas identified by the *ContentPlaceHolder* control in the master page. Pages that use a master page to define the layout can place content only in the areas defined by the *ContentPlaceHolder*, thus enabling a consistent site design. Master pages are saved with the file extension .master. Apart from containing all the contents that are required for defining the standard look and feel of the application (css files, javascript files etc.), the

master pages also contain all the top-level HTML elements for a page, such as <html>, <head>, and <form> [12].

In order to display data to the user we used a GridView control, bound to a SqlDataSource control or a method that retrieves data from the DataSet.

Each member has own grids according with RBAC and schedules or phases assigned for retrieves data from the DataSet access, based on personalized rights.

The C# code for binding the GridView to the data is:

```
MembershipUser        manager        =
Membership.GetUser();   //gets   the
current user (manager)
string           managerID        =
manager.ProviderUserKey.ToString();
//gets the user's id
ProjectDataSetTableAdapters.Projects
TableAdapter      dt      =      new
ProjectsTableAdapter();
GridView1.DataSource            =
dt.GetProjectsByManager(managerID);
GridView1.DataBind();
```

4 Conclusions and Future Work

The web application presented in this paper is an ASP.NET/C# project management solution that implements the 3-tier architecture in order to separate data access, business rules and user interface. By using our application you can easily manage a general project, team members, assign tasks, generate reports, receive email notifications, and complete tasks (according to RBAC permissions). The user interface is intuitive, easy to access and easy use, offers a great user experience.

Benefits of EasyProject consist in: better team management, cost effectiveness and portability because this solution no special hardware or software required. As the entire EasyProject application is web-based, it allows you to manage and control your project teams from anywhere, anytime. It makes it easy for you to supervise collaborative and dynamic teams that could be spread all over the world.

As businesses become more global, project team members seem to be located in different areas of the world and business travel costs are soaring. Our application is designed to bring all project teams and project-related information together, in one central, online, organized collaborative workplace. The mission for a company is to provide businesses with a flexible, easy and cost-

effective collaborative platform for controlling project tasks and processes.

Since EasyProject is a totally web-based application, all you need to be able to use it is a sufficiently recent version of a web browser, and Internet access. The prototype does not require special hardware features or expensive software license, no need to install any drivers or extra hardware devices.
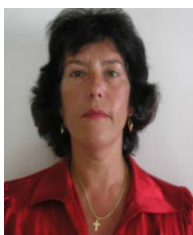
As a future development, the reporting module will be improved with new reports and charts and a document sharing section will be added. Also, our management solution will be adapted to mobile devices. In this way, the application will be accessible from any device connected to the internet that has a recent version of a web browser.

References
[1] D. Donnan, "CEO/Presidents' Forum, Action Plan for Trading Partner e-collaboration," *GMA CEO/Presidents Forum*, pg. 8, 2002.
[2] I. Marmaridis, J. A. Ginige and A. Ginige, "Web based architecture for Dynamic eCollaborative work," *International Conference on Software Engineering and Knowledge Engineering*, 2004.
[3] I. Marmaridis et al, "Architecture for Evolving and Maintainable," *Web Information Systems, IRMA04.544*, 2004.
[4] K. Schuster, *Cross-Industry Standard Key to eCollaboration Success*, News Release issued by Ticona, Philadelphia, 2002.
[5] V. Prasad and Y. Lee, "A scalable infrastructure for peer-to-peer networks using web service registries and intelligent peer locators," *Cluster Computing and the Grid, Proceedings. CCGrid 2003. 3rd IEEE/ACM International Symposium*, pp. 216-223, 2003.
[6] *PMI*, 2007, Available at: http://www.projectmanagement.tas.gov.au/guidelines/pm5_1.htm
[7] *PM WORLD TODAY – MARCH 2009*, Available at: http://www.pmforum.org/library/perspective/2009/PDFs/ mar/Editors-Perspective-on-the-March-2009-Edition.pdf
[8] *Project Management Software that boosts your project success*, Available at: http://www.projectmanager.com/Project-Manager-Brochure.pdf
[9] *Online Project Management Tools*, Available at: http://www.dezinerfolio. com/2008/10 /28/online-project-management-tools
[10] *ASP.NET Quick Start Tutorials*, Available at: http://quickstarts.asp.net/ quickstartv20/aspnet/
[11] *ASP.NET Data Access Tutorials*, Available at: http://www.asp.net /learn/data-access/
[12] http://www.web-based-software.com/

Lucia RUSU, PhD, Professor at Babes-Bolyai University of Cluj Napoca, Business Information Systems Department from 2001. She is graduate in Computer Science and Automatic Control, Technical University of Cluj Napoca, and Doctor in Economy, Business Information Systems domain from 1997. Her research interests: graphics and web programming, multimedia and distributed technologies, integrated systems EAI and ERP, workflow management systems, processes modeling and management, mobile applications development. She was member in several research grants in several domains: web applications management, quality management, collaborative systems, workflow management systems.
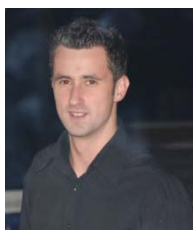
Vasile RUSU is graduate at Babes-Bolyai University of Cluj Napoca, Business Information Systems Department from 2009. His skils is focused on: Asp.Net technology, PHP, XML, Microsoft Office, Oracle database management systems. Research interests: workflow management systems, processes modeling and management, collaborative systems. He achieved ThInvest Junior award at Student Contest and Conference 2009. He is Customer's support engineer at SoftVision, Cluj-Napoca, Romania.