Seavus AB   
Skeppsbron 5, 211 20 Malmoe, Sweden  
Tel: +46 40 300 940   
Fax.+ +46 40 300 941   
info@seavus.com, www.seavus.com

**SOLUTION DESCRIPTION FOR**

**Seminar Works Management System**

**IN RESPONSE TO**

**Request for proposal from**

**Presented to:**

**-----**

**VERSION: 1.0**

**Date:**

**© 1999-2015 Seavus Group**

Contents

[1. Executive summary 3](#_Toc412813803)

[2. Solution description 3](#_Toc412813804)

[2.1 Functional requirements 3](#_Toc412813805)

[2.1.1 Module for user registration 3](#_Toc412813806)

[2.1.2 Admin 4](#_Toc412813807)

[2.1.3 Module for user authentication 4](#_Toc412813808)

[2.1.4 Module for creating seminar work subjects 4](#_Toc412813809)

[2.1.5 Module for seminar work lifecycle 4](#_Toc412813810)

[2.1.6 Module for listing seminar work subjects and complete seminar works 4](#_Toc412813811)

[2.1.7 System Administration module 5](#_Toc412813812)

[2.2 Non-functional Requirements 5](#_Toc412813813)

[2.2.1 Security 5](#_Toc412813814)

[2.2.2 Accessibility 5](#_Toc412813815)

[2.2.3 Performance and response time 5](#_Toc412813816)

[2.3 Architectural and component-level design 5](#_Toc412813817)

[2.3.1 Architecture design 5](#_Toc412813818)

[2.3.2 Components 6](#_Toc412813819)

[2.4 Technologies 6](#_Toc412813820)

[2.5 Assumptions and Limitations 6](#_Toc412813821)

**Table of figures**

[Figure 1: Product lifecycle 3](#_Toc412038308)

[Figure 2: System Design 3](#_Toc412038309)

[Figure 3: Architecture overview 4](#_Toc412038310)

[Figure 4: System flow 4](#_Toc412038311)

# Executive summary



Figure 1: Product lifecycle

# Solution description

The solution description explains the functional and technical aspects of the implementation of the System for Seminar Works Management. AAA

The solution will consist of a web application, publically available for maximum accessibility and cooperation.

## Functional requirements

Functional requirement defines a function of a system and its components. A function is described as a set of inputs, the behaviour, and outputs.

### Module for user registration

The system will have three types of users: student, professor and admin. Administrator user will be automatically created.

This module will be publically available and accessible through a link from the main page.

This module will be implement as web page named Register.aspx. The web page will contain following fields (all fields will be mandatory):

1. Email that will be used as username. We must be sure that the user will enter valid email, for example: [user@server.com](mailto:user@server.com), not user—server….com.

2. Password. A textbox where user can type its password.

3. Dropdown list (combo box) with two options (Професор and Студент)

4. First name.

5. Last name.

6. Button that will save the user registration.

#### Student

Registers into the system with email, password, first name, last name and index number. After successful register the student can browse published seminar work subjects and apply for seminar subjects published by professors.

#### Professor

Registers into the system with email, password, first name and last name. After successful registration the professor can: create seminar work subjects, approve student’s applications hence creating seminar work drafts, lead the process of the seminar work lifecycle, and finally evaluate the seminar work, provide grade and publish.

### Admin

Admin user will have access to the System Administration Module only and will have a role of listing all users, changing/resetting user’s passwords and

### Module for user authentication

This module will provide user the functionality to login into the system by providing their username and password.

This module will be implement as web page named Login.aspx. The web page will contain following fields (all fields will be mandatory):

1. Email that will be used as username. We must be sure that the user will enter valid email, for example: [user@server.com](mailto:user@server.com), not user—server….com.

2. Password. A textbox where user can type its password.

### Module for creating seminar work subjects

In this module a professor will define a starting point for each seminar work – a research topic or thesis. The subject will contain: title, related course, content, apply conditions, date of closing, and expected output.

Once a subject is created, students can apply, and at a given time (usually the date of closing) the professor will choose one (or more) of the student’s applications, confirm and and promote the topic to a seminar work draft.

### Module for seminar work lifecycle

An approved topic assigned to one or more students is promoted as a seminar work draft. This draft is then accessible for the actors (assigned students and the professor) who collaborate by uploading documents and providing feedback in a form of comments and uploaded documents.

The professor finally concludes that the seminar work is complete, promotes it to finalization, provide evaluation and grade.

### Module for listing seminar work subjects and complete seminar works

This module consists of two sub-modules: module for listing seminar work subjects and module for listing complete seminar works.

This module will be the landing (home) page of the website.

#### Listing seminar work subjects

All seminar work subjects that are created by professors and are in the period of accepting applications. This module is accessible for view for the registered student and professor users, and only student users can apply.

#### Complete seminar works

Complete seminar works are listed and categorized per course. This module is publically available to all users including anonymous visitors of the site.

### System Administration module

System administration module will be accessible to the admin user only. This module is used to list all users, rest users’ passwords, delete users.

## Non-functional Requirements

Non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours.

### Security

### Accessibility

### Performance and response time

## Architectural and component-level design

Figure 2: System Design

### Architecture design

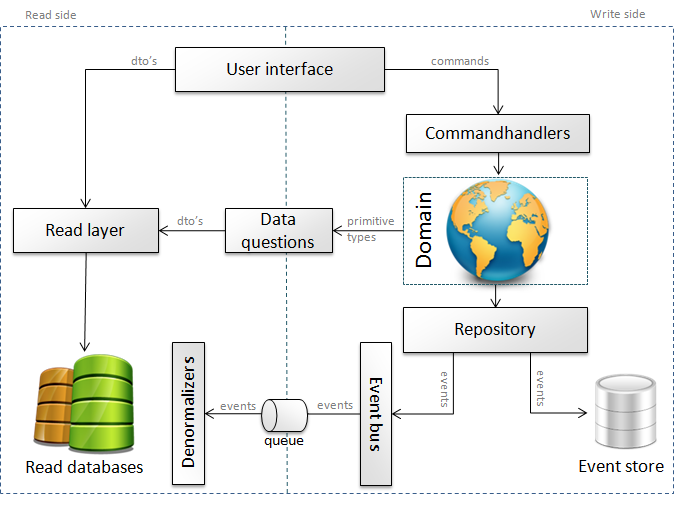


Figure 3: Architecture overview

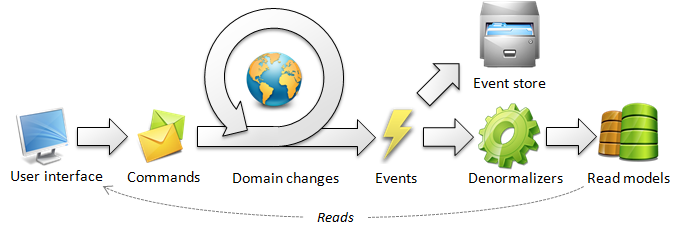


Figure 4: System flow

### Components

## Technologies

The project will be developed using following technologies:

* Microsoft Visual Studio 2013 Community Edition;
* Microsoft SQL Server 2012 Express Edition;
* Windows IIS web server or Visual Studio development web server;
* Microsoft .NET framework 4.5;
* Microsoft ASP.NET Web Forms;

Complete solution will be hosted on a server with sufficient hardware capabilities and pre-installed:

* Microsoft Windows OS 2008 or newer;
* Microsoft SQL Server 2012 of any edition
* Microsoft Windows IIS server version 7 or newer

## Assumptions and Limitations

In order for this project to be possible to deliver with the terms committed to in this document and appendixes (if any), the following assumptions and limitations have been made: