KV Web Engineering
SS 2010

durchgeführt am
Institut für Anwendungsorientierte Wissensverarbeitung
der Johannes Kepler Universität Linz

Leitung: a.Univ.-Prof. Dr. Birgit Pröll

Aufgabe 1

Gruppe F

21. 2010

Ziele:

Ein Web-Applikation, die für ungarische Studenten und Lehrer gemacht wurde, um sich über das MSDNAA Programm Informationen zu besorgen. Unsere Applikation haben wir derzeit für unsere Universität entwickelt. MSDNAA ist eine Abkürzung von Microsoft Developer Network Academic Alliance, was als ein Programm von Microsoft für Universitäten und Hochschulen gestartet wurde, in welchem die Lehrer und die Studenten und Studentinnen des Instituts fast jedes Programm von Microsoft kostenlos herunterladen können und danach können sie die Programme für Lehr-und Forschungszwecken kostenlos verwenden können. (In Rahmen der Lizenz kann man Beta-Versionen, neue Ausgaben und Development-Kit auch herunterladen.)

Der Zweck dieser Webseite ist es, weil es momentan keine solche Webseite gibt, darüber interaktive Informationen über die Ausgabe der neuen Programme zu erhalten. Die Lizenz von MSDNAA gehört zu einem (Bildungs-)Institut, deswegen können die Studenten, Studentinnen und Lehrer durch die MSDNAA Administratoren auf den Inhalt zugreifen.

Charakteristika von unserer Web-Anwendung:

Unsere aktuelle Web-Anwendung haben wir anhand der gelernten Aspekte versucht zu prüfen: Produkt, Nutzung, Entwicklung, Evolution. Im Produkt ist sehr wichtig der Inhalt, der jetzt im Hintergrund ist, weil die autorisierten Personen (wie Administratoren) den Contant aufladen werden, so ist die Anwendung in unserem Fall ein Rahmen und der Inhalt wird in der nächsten Zeit aufgeladen werden. Hier ist die wichtigste Eigenschaft die Navigierung, die Selbsterklärbarkeit, die Übersehbarkeit, die Nutzbarkeit und Benutzerfreundlichkeit,die unserem Menüsystem vollständig entsprechend ist. Die Erscheinung der Seite ist ähnlich der modernen Webseiten (zum Beispiel: Dropdown-Hauptmenu). Der Hintergrund, die Farben und die Texten sind aufeinander abgestimmt. Die Texte vom Menü sind kurz und sie zeigen uns das Wesen ("Look and Feel" Gefühl). Die Verfügbarkeit ist noch nicht die Frage, weil die Anwendung im Internet noch nicht benutzt wird, aber wurde es schon unter mehreren Web-Browser (siehe unten Test) geprüft. Leider ist die Anwendung bis jetzt nur auf Ungarisch verfügbar.
Entwicklungsbezogene Charakteristika: die Web-Anwendung benutzt viel Technologie (siehe Technologie) und wurde nach Repository Pattern (nach offiziellen
UML-Pattern) aufgebaut, die den EntwicklerInnen in der Zukunft hilft, weil die Source-Code der Anwendung einem Standard folgt. Die Applikation ist moduliert
und gut dokumentiert also übersichtlich. Weil die Anwendung noch neu ist, kann man noch nicht über Evolution sprechen, aber natürlich ist die Anwendung für Zukunft vorbereitet, sowohl aus dem Gesichtspunkt der Technologie (zum Beispiel: es kann sehr viele Daten in Datenbanken speichern und effizient behandeln) als auch aus dem Gesichtspunkte des Benutzer.

Categories of Web Applications:

Wir können unseren Web-Applikation in zwei Gruppen einordnen:

Interactive

Wegen des “Nachrichten” Menüpunktes (in Ungarisch “Hírek”), wo die Nachrichten über die neuen Programme und die wichtigen Informationen erscheinen.

Wegen des „Forum“ Menüpunktes (in Ungarisch “Fórum“), wo die Benutzer die vorrätigen Programme in neuen Versionen oder in anderen Sprachen bekommen können.

Collaborative

Im Forum können die Benutzer und die Administrators miteinander über die Entwicklung der Web-Anwendung oder über die falschen Download-Links reden. Das Forum ist eine Support-Schnittstelle.

Die Richtung der Entwicklung ist eine „Portal-Oriented“ Web-Applikation. Wir möchten ein Community Portal entwickeln und wir möchten dieses mit dem zentralen MSDNAA Download-Portal verbinden. Unsere Web-Anwendung kann man in die „Dynamic Web Application“, „Web-based Application“ und „Database-related Web Application“ Gruppen einordnen.

Technologie, Architektur, Web/DB-Anbindung:

Die Entwicklung-Tools, die wir verwendet haben, sind die Entwicklungsumgebung (Development Environment) „Microsoft Visual Studio 2008 (Professional Edition SP1)“, das Datenbanksystem „Microsoft SQL Server 2008 (Express Edition)“ und die Programmiersprache „C#“. Bei dieser Web-Anwendung haben wir drei Frameworks benutzt, die waren: „.NET Framework 3.5 SP1“, „ASP.NET Framework 2.0, 3.5“ und „ASP.NET MVC 1.0 Framework“.

Die verwendeten Technologien:

MVC(“Model-View-Controller”) Designpattern

Model View Controller (MVC, ‚Modell/Präsentation/Steuerung‘) ist ein Architekturmuster zur Strukturierung von Software-Entwicklung in die drei Einheiten Datenmodell (engl. model), Präsentation (engl. view) und Programmsteuerung (engl. controller). Ziel des Musters ist ein flexibler Pro-

grammentwurf, der eine spätere Änderung oder Erweiterung erleichtert und eine Wiederverwendbarkeit der einzelnen Komponenten ermöglicht.

Die drei Komponenten sind:

Modell:

Das Modell enthält die Daten und die Geschäftslogik. Es ist von Präsentation und Steuerung unabhängig. (Data Access Layer)

Präsentation:

Die Präsentationsschicht ist für die Darstellung der benötigten Daten aus dem Modell und steuert die Benutzerinteraktionen. Es ist die so genannte Benutzerschnittstelle. (Presentation Layer)

Steuerung:

Die Steuerung verwaltet eine oder mehrere Präsentationen und bearbeitet die Benutzeraktionen. Zu jeder Präsentation gibt es ein Modell. Die Steuerung manipuliert die Daten nicht, diese Aufgabe gehört zum Modell. Sie enthält weiterhin Mechanismen, um die Benutzerinteraktionen der Präsentation einzuschränken. (Eventhandler Layer)

LINQ to SQL:

Die „LINQ to SQL“ Technologie ist eine spezielle Implementierung von LINQ (Language Integrated Query). Die „LINQ to SQL“ transformiert die LINQ Ausdrücke in dynamische SQL Ausdrücke und führt die SQL Ausdrücke an der SQL Server aus.

LinqtoXML:

Die „LINQ to XML“ Technologie ist eine andere spezielle Implementierung von LINQ. Mit dieser Technologie kann man die Daten leicht erreichen, die in XML gespeichert werden.

Repository pattern

Das Repository Pattern ist ein Architekturmuster zur Behandlung der Daten. Die Repositories sind verantwortlich für die Erreichbarkeit der Daten, die Definierung der Daten, die Änderung der Daten und die Löschung der Daten.

BBCode

Das BBCode ist eine Markup-Sprache, mit dem kann man die Input-Texte in den Nachrichten und die Kommentare im Forum formatieren, weil die HTML und JavaScript Codes in diesem verboten sind.

6. FriendlyURL

Das FriendlyURL ist eine Technologie, mit dem können wir merkbare URLs machen. Es arbeitet mit Regular-Expressions und das Ergebnis von dieser Technologie ist eine freundliche URL für Browser und Benutzer.

Anforderungsanalyse:

Die Webseite soll einfach übersichtlich werden.

Die Schnittstelle soll einfach benutzbar werden.

Durch das Menü soll jede Seite erreichbar werden, wozu der Benutzer der Seite die Berechtigung hat und dieses Menü soll auch flexibel werden.

Dieses Programm soll enthalten eine Methode, womit man den Benutzer kontrollieren kann und eine Registrationsfläche.

Einige Teile des Programms sollen auch erreichbar werden, für solchen Besucher, wer noch nicht registriert ist.

Das Programm soll auch solche Fläche enthalten, wo die MSDNAA Administratoren auf die entstehenden Fragen antworten können.

Die Benutzer sollen differenziert werden.
Benutzer soll auch werden, wer mit verschiedenen Berichtigungen verwaltet wird, wie z.B. die Administratoren.

Erfahrungen mit verwendeter Technologie (Vorteile/Nachteile)

Die Grundtechnologie, in der die Anwendung entwickelt wurde, war die ASP . NET MVC (Model-View-Controller). Diese moderne und ziemlich neue Programmsprache ist nützlich, modern und damit kann man seine Programm modulieren. Erstmal war es nicht so einfach es zu benutzen, weil es manchmal kompliziert ist, braucht man mehr Zeit sich daran ein bisschen zu gewöhnen. Aber es ist später übersichtlicherer und r Aufwand lohnt sich. Neben dieser Programmsprache haben wir auch andere Technologien benutzt (siehe Technologie). Einige andere Nachteile und Vorteile:

Lineq to SQL: es ist nur DML (Data Manipulating Langauage) und man hat keine Möglichkeit neuen Daten zu definieren, nur die Daten zu behandeln aber es war in unseren Fall genug.

Lineq to XML: es wurde nur wegen des Menüsystems benutzt, dazu war es entsprechend. Wir haben die Grenze dieses Programms nicht erfahren.

BBCode: damit kann man die Eingabetexte formieren und in einer Textbox sind die HTML-Tags nicht erlaubt; sehr einfach zu benutzen und nützlich