|  |  |  |  |
| --- | --- | --- | --- |
|  | 12/30/2009 |  | |
| |  |  | | --- | --- | |  |  | |  |  | |  |  | |  |  | | |  | |
| GiveAQuiz.com  *Proposal* | | | |
|  |  | | |  |  |  | | --- | --- | --- | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |  |  |  | |
|  |  | | Shawn Wildermuth |

Table of Contents

[Overview 3](#_Toc249905939)

[Deliverables 3](#_Toc249905940)

[Price, Timing and Payment 4](#_Toc249905941)

[Project Specification 5](#_Toc249905942)

[Goals 5](#_Toc249905943)

[Non Goals 5](#_Toc249905944)

[User Stories 6](#_Toc249905945)

[Story 1: Glenda. 6](#_Toc249905946)

[Story 2: Chloe. 6](#_Toc249905947)

[Story 3: Jake. 6](#_Toc249905948)

[Story Flowchart 7](#_Toc249905949)

[Technologies Used 8](#_Toc249905950)

[Screen by Screen Specification 9](#_Toc249905951)

[Home Page 10](#_Toc249905952)

[Register User 11](#_Toc249905953)

[Login 12](#_Toc249905954)

[About 13](#_Toc249905955)

[Quiz Creator – Quiz Info 14](#_Toc249905956)

[Quiz Creator – Quiz Editor 15](#_Toc249905957)

[Quiz Creator – Question Editor 16](#_Toc249905958)

[Quiz Management Page 17](#_Toc249905959)

[Quiz Results Page 18](#_Toc249905960)

[Quiz Results Report 19](#_Toc249905961)

[Quiz Detail Report 20](#_Toc249905962)

[Pick a Quiz Page 21](#_Toc249905963)

[Take a Quiz Page 22](#_Toc249905964)

GiveAQuiz.com

Proposal

# Overview

**GiveAQuiz.com is a new web site that allows users to create quizzes and have other users take the quizzes.** This is differentiated from polling web sites as this project will allow the creator of a quiz to view the each individual user’s results instead of just summary information (which is especially important for grading contextual results like essay questions). This web site is aimed squarely on users that need tocreatequizzes and give grades on the results.

## Deliverables

This project shall consist of the following deliverables:

* Database, code and other assets to build and host the website.
  + This code shall be housed and available via a MS-PL licensed CodePlex project.
  + This code shall be exemplar quality.
* Whitepaper on building GiveAQuiz.com.
  + Shall not be less than 3,500 words.
  + Videos on building GiveAQuiz.com.
* No fewer than six videos whose length shall be between fifteen and thirty minutes.
  + Videos shall be full-screen desktop (e.g. Camtasia) with quality voice overs.
  + The videos shall be delivered weekly.

We propose that the project be built transparently, therefore once the project commences the code will be available on CodePlex as the project matures. This allows Microsoft to review the code as the project is being built to lower the chance that any surprises about what is being built will happen.

Once the code has been completed, it will be hosted on Wildermuth Consulting Services, LLC’s servers at its own cost (which we may defray with reasonable advertising if necessary).

## Price, Timing and Payment

This project shall take approximately five weeks to build the web site and another two weeks to write the whitepaper. Therefore we suggest the following schedule milestones:

* **Week 1**: Kick-off project
* **Week 3**: Review of progress with expectation that most of the database and ASP.NET work will be complete.
* **Week 5**: Handoff of final project to include Silverlight 4, reports and final design.
* **Week 7**: Delivery of the whitepaper
* **Week 8**: Start of delivery of videos.

The cost of the project shall be $24,000 and be paid in installments at the following payment milestones:

* 1/3 on Kick-off
* 1/3 on Code Hand-off
* 1/3 on Delivery of Whitepaper

# Project Specification

This spec is not, by any stretch of the imagination, complete.All of the wording will need to be revised several times before it is finalized. The graphics and layout of the screens is shown here merely to illustrate the underlying functionality. The actual look and feel will be developed over time with the input of graphics designers and iterative user feedback.

## Goals

The goals of this project include:

* Present to developers a real-world example of how to use Microsoft data technologies to build web sites.
* Provide a community solution for online quizzes.
* Represent best practices for architecting solutions using data technologies.
* Produce code that is of high quality so that it can be published as an open source project (under MS-PL).

## Non Goals

This version will *not* support the following features:

* No localization is going to be supported.
* OpenID will not be implemented.

## User Stories

In designing products, it helps to imagine a few real life stories of how actual (stereotypical) people would use them. We'll look at three scenarios.

### Story 1: Glenda.

Glenda teaches Algebra in a public high school. She wants to give quizzes to her students but the current way of printing out quizzes and spending all night grading them is a pain. On top of that, using paper tests does not seem very environmentally friendly to Glenda. She decides to turn to the Internet for a solution. After googling for quizzing engines with Bing, she decides to try out “GiveAQuiz.com”. She registers with the new site ready to create her first quiz. She walks through the quiz designer and creates a ten question quiz to test out on her students. She is asked to supply a pass-phrase she can hand out to her students so they can take the quiz without having to register with the site. She sets the time limit of fifteen minutes for the test. A couple days pass and she returns to the “GiveAQuiz.com” site to see the results. She sees that most of her students have taken the test and realizes that two of the questions were missed by the majority of the students. She uses this information to update her lesson plan to provide more information on those parts of the curriculum.

### Story 2: Chloe.

Chloe is a high school student. While in Glenda’s class, she was instructed to visit the “GiveAQuiz.com” to take a quick quiz. When she returns home, she visits the site and heads to the “Take a Quiz” section. The site requests she enter her name and pass-phrase supplied by her teacher. She proceeds to answer the questions, one at a time, and submits the test (after being given a chance to review her answers).

### Story 3: Jake.

Jake is a software consultant. He has been spending time at a client helping mentor their team on different aspects of development. He is pulled aside by one of the managers who confided in him that they wanted to add a couple of members to the team. In order to figure out which members of the team to add, they wanted him to come up with a simple programming test for the potential team members. To simplify giving the test, Jake turned to “GiveAQuiz.com”. He registered in order to create a quiz and created a quick twenty questions to test the candidates. He returned after the tests were given to see the results and looked at the quiz summary report to see the top three of the candidates to tell the manage which ones he’d move to the new team.

## Story Flowchart

This is a simplified flow of how different users will use the site:



## Technologies Used

Since the purpose of this project is to highlight a real-world usage of Microsoft technologies to solve business problems, this project shall be an example of how to build an application. The project shall use a number of Microsoft data technologies including:

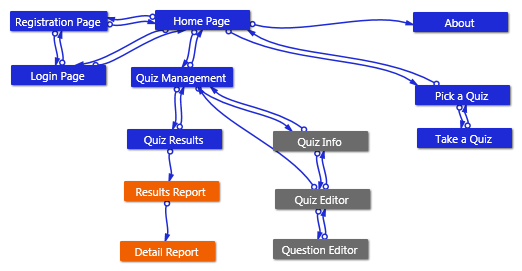
* Code name "M" modeling language: For designing/maintaining database design and building the entity data model (EDM).
* Quadrant: To allow editing of system metadata.
* Entity Framework 4: Using EDMX from SQL Server Modeling Services for data access.
* WCF Data Services 2: For communication to AJAX and Silverlight components.
* SQL Server Reporting Services: For generating Quiz Result reports, drill-down reporting and scheduling them for e-mail delivery.

## Screen by Screen Specification

**GiveAQuiz.com**consists of quite a few different screens. Most screens will follow a standard format, with a look and feel to be designed in the future by a graphic designer. This document is more concerned with the functionality and the interaction design, not the exact look and layout.

All screens are created in HTML with the exception of the QuizCreator which will be a Silverlight 4 application.

The navigation between the screens is shown below. The screens in Blue are HTML pages, Orange ones are SSRS Reports and the ones in Gray are Silverlight pages in the Quiz Creator Silverlight Application:



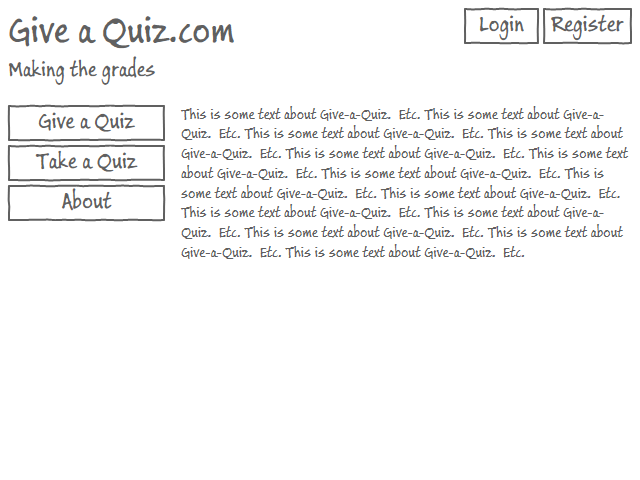
Each screen in **GiveAQuiz.com** is known by a canonical name which will always appear, in this document, with an underline, so you know we're referring to a screen by name, for example, Home Page.

### Home Page

The Home Page serves three purposes:

1. Allow people to learn about the service and consider whether they want to sign up.
2. Allow members who have already signed up to log on.
3. Allow people who want to sign up to create an account.

The Home Page looks like this:



On this, and on *all screens*, clicking on the **GiveAQuiz.com** logo in the top left corner goes back to Home Page.

Technical Note  
The top pane of the page should be embedded into a MasterPage view so that global changes are isolated to one location in the site.

### Register User

The Register User Page allows users to register with the site.

The Register User Page looks like this:

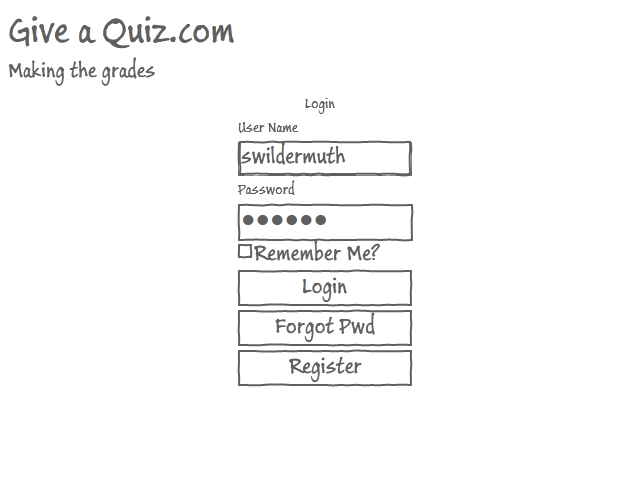


Technical Note  
Only users who need to create and manage quizzes will be required to register with the site. Quiz takers are not required to register. The registration will be handled using ASP.NET Forms Authentication using the built-in providers. We will implement a e-mail confirmation for new registered users to confirm that registrations are valid.

### Login

The Login Page is used to allow existing users to log into the system.

The Login Page looks like this:

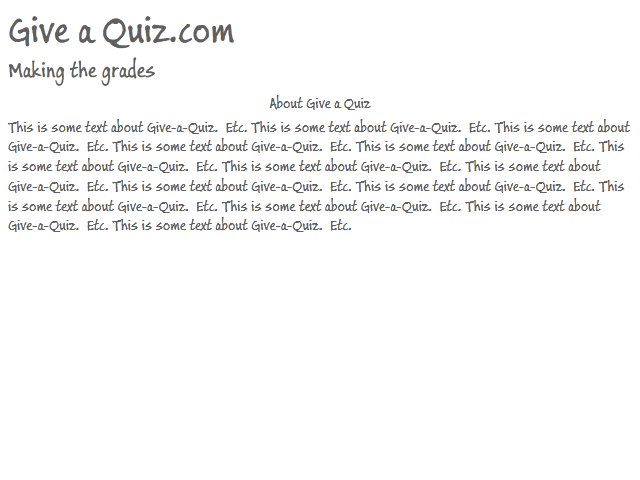


Technical Note  
The authentication/authorization will use standard ASP.NET Forms Authentication. The Forgot Password facility will be enabled.

### About

The About Page will include information on how the site was built with links to the different technologies used and the whitepaper/videos that are produced as part off this project

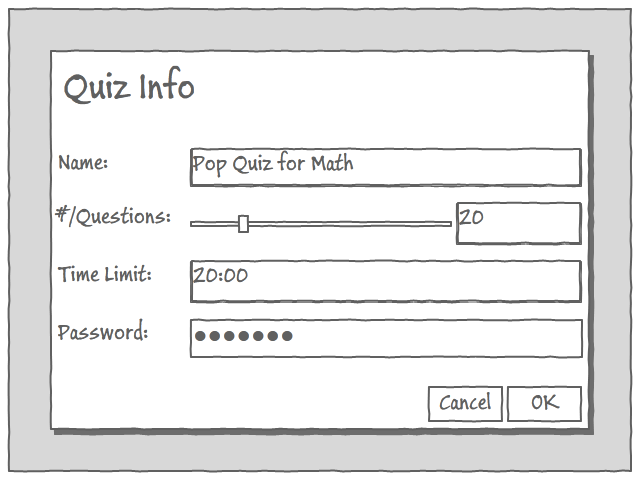
The About Page looks like this:



### Quiz Creator – Quiz Info

The Quiz Creator – Quiz Info is the first page of the Quiz Creator that is presented to the user when editing or creating a quiz. This is where the basic information about the quiz is presented to the user.

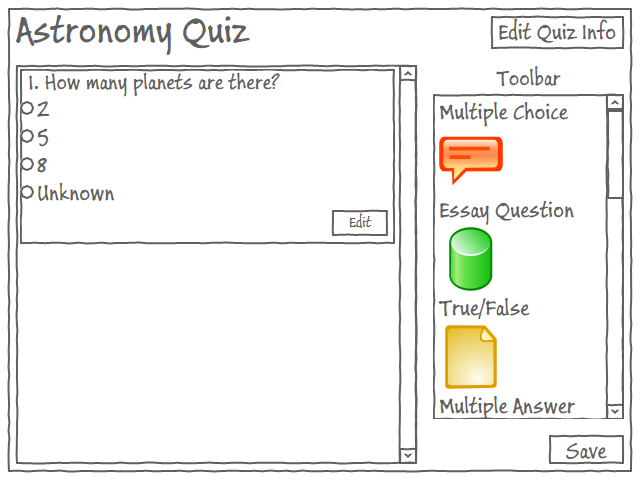
The Quiz Creator – Quiz Info looks like this:



### Quiz Creator – Quiz Editor

The Quiz Creator – Quiz Editor allows users to pull question types from the ‘toolbar’ to create their test. The questions support drag-n-drop to resize as well. Each of the questions supports HTML generation and an editor to allow the quiz author to specify the question, answers and right answer (if any).

The Quiz Creator – Quiz Editor looks like this:

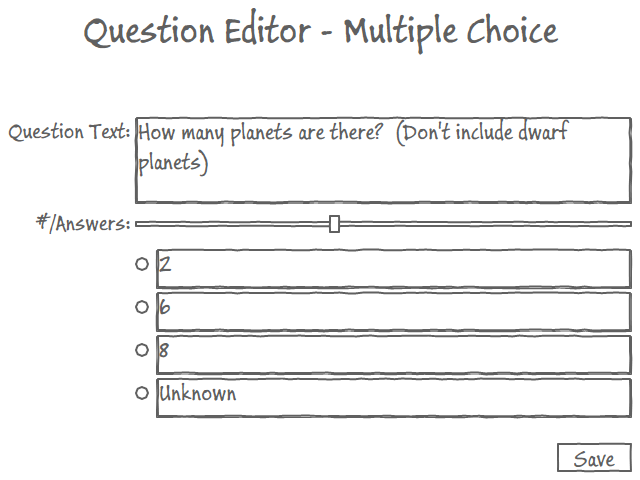


Technical Note  
Each question type will be packaged as an add-in to support adding new question types simply. The question type will be used on the server to generate the question HTML as well as in the Silverlight application to support preview and editing of the individual question.

### Quiz Creator – Question Editor

The Quiz Creator – Question Editor allows users to edit individual questions based on the question type.

The Quiz Creator – Question Editor looks like this:



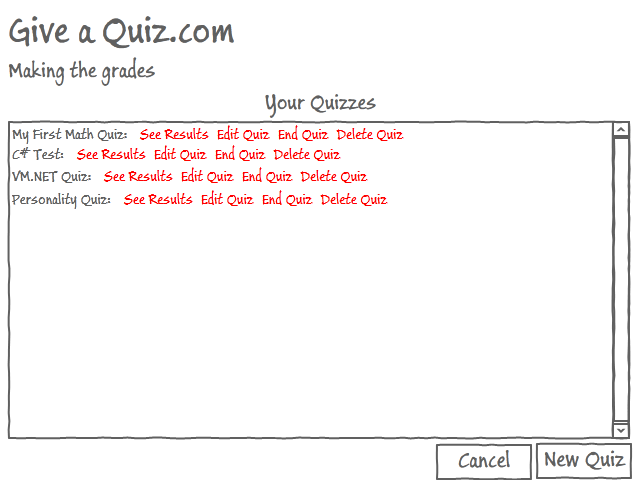
Technical Note  
The Quiz Creator – Question Editor will be implemented as a pop-up using Silverlight’s ChildWindow class.

### Quiz Management Page

The Quiz Management Page allows registered users to do the following tasks:

* Create new Quizzes
* Edit existing Quizzes
* End Running Quizzes
* Delete existing Quizzes

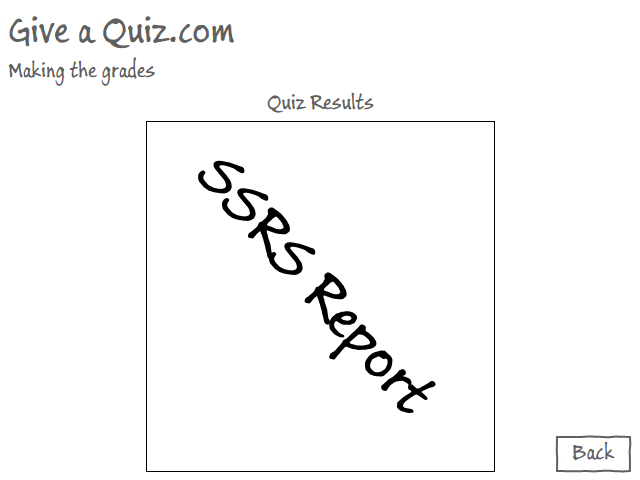
The Quiz Management Page looks like this:



## 

### Quiz Results Page

The Quiz Results Page allows quiz authors to view the results of a test. This will display a report as generated by SQL Server Reporting Services.

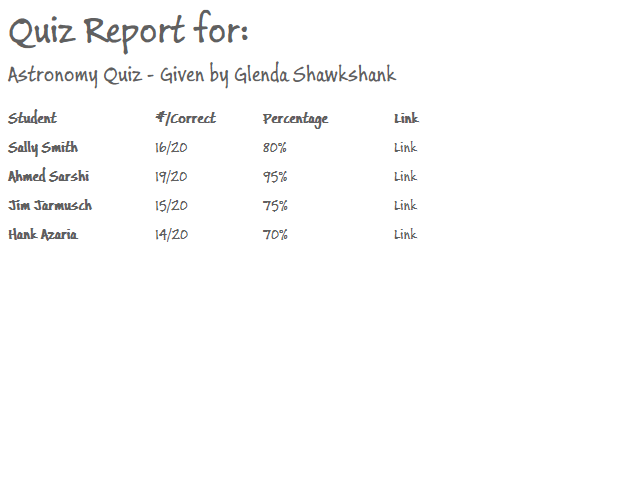


Technical Note  
The report will be embedded on the page and any navigation to dril-down reports will be in-place.

### Quiz Results Report

The Quiz Results Report shows the full results (based on upon each participant in the test). Details of each participant’s results can be retrieved by following a drill-down into the details.

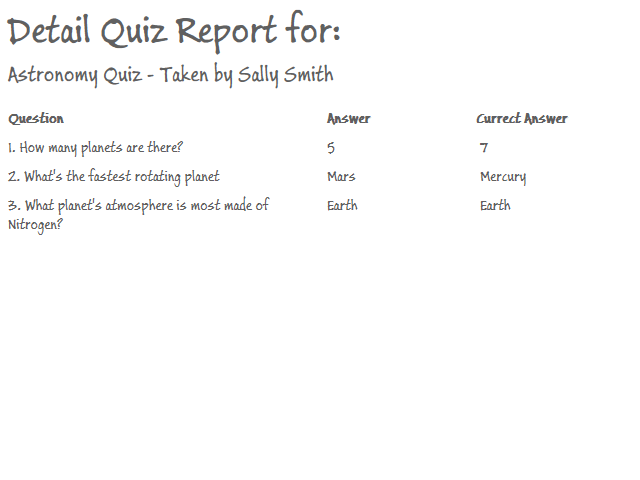
The Home Page looks like this:



### Quiz Detail Report

The Quiz Detail Report shows the detail answers from a specific quiz taker.

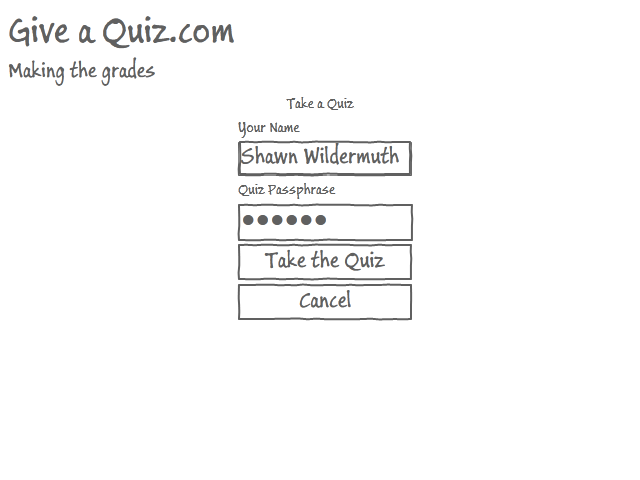
The Quiz Detail Report looks like this:



### Pick a Quiz Page

The Pick a Quiz Page allows anyone to take a quiz via a pass-phrase.

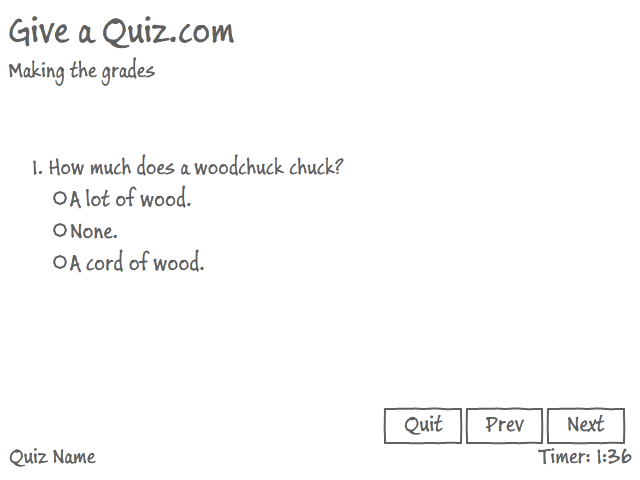
The Pick a Quiz Page looks like this:



### Take a Quiz Page

The Take a Quiz Page gives a quiz by presenting the questions one at a time and allowing the user to answer the questions. The quiz taker can go forward and back to review questions once they’ve been answered.

The Take a Quiz Page looks like this:



Technical Note  
Currently this is planned to be generated in HTML, but if the timing of tests proves impossible via JScript and HTML, we may implement this using Silverlight.