Iteration 2 Review

Contents

Implemented User Stories 2

Story ID: 4 - Table Exports 2

Story ID: 8 - Mobile Forms 2

Story ID: 7 – Public URLs 2

Story ID: 14 - Error Logging 2

Related Progress 3

Mobile Site 3

Hidden Options 3

CSS 3

Class Diagram 3

Database Schema 5

Entity Framework 5

SMO Database 6

Revised Release Plan 7

Difficulties and Learning Outcomes 8

Story ID: 4 - Table Exports 8

Story ID: 8 - Mobile Forms 8

Validation 8

Story ID: 7 – Public URLs 8

State diagram 9

Interaction diagram 10

# Revised User Stories

### Log In

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 1** | **Story Title: Log In** | **Priority: High** | **Estimate: 2 hours** |
| As a user I want to log in so that I can access my account | | | |

**Confirmation:**

1. **Success** – valid user logged in and referred to personal home page
   1. User Presented with a list of their associated sheets table entries.

Which can be selected to view the individual signatories.

1. **Failure** – display message
   1. Username and hash of password does not match that stored in the database.
   2. Display failed login message, allow retry up to 3 attempts.

### Generate Form

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 2** | **Story Title: Generate Form** | **Priority: High** | **Estimate: 5 hours** |
| As a User I want to generate custom registration forms so that I can capture data from clients | | | |

**Confirmation:**

1. **Success** – Valid table names provided, table created, stored and associated with user profile
   1. User returned to list of tables including newly created one.
2. **Failure** – Invalid Input
   1. User enters values disallowed by form validation
      1. Indicated by highlighting invalid field
   2. Failed table creation
      1. Notify user and allow retry

### Select Input type

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 3** | **Story Title: Select Input Type** | **Priority: High** | **Estimate: 5 hours** |
| As a User I want to select predetermined input types (radio buttons, checkboxes) for certain custom fields, and be able to specify custom answers. | | | |

**Confirmation:**

1. **Success** – Table created with defined input types and default question layout.
   1. User returned to list of tables including newly created one.
2. **Failure** – Invalid Input
   1. User enters values disallowed by form validation
      1. Indicated by highlighting invalid field
   2. Failed table creation
      1. Notify user and allow retry

### Table Exports

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 4** | **Story Title: Table Exports** | **Priority: High** | **Estimate: 5 Hours** |
| As a User I want to export Tables so that I can easily import them to other applications. | | | |

**Confirmation:**

1. **Success** – File downloaded to client machine.
   1. File downloaded to client machine at user defined location
2. **Failure** – Cannot export table.
   1. Table may be being updated by another device (AJAX call)
   2. Table not found

### Admin Access

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 5** | **Story Title: Admin access** | **Priority: High** | **Estimate:4 Hours** |
| As a sit administrator I want access users information to build profiles and send emails. | | | |

**Confirmation:**

1. **Success** – View list of usernames, full names, location and email addresses.
   1. View displayed to admin
2. **Failure** –
   1. Detail failures & messages

### 

### Account Creation

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 6** | **Story Title: Account Creation** | **Priority: High** | **Estimate: 5 Hours** |
| As a User I want to be able to create new account so that I can log in. | | | |

**Confirmation:**

1. **Success** – User account created
   1. user redirected personal homepage
2. **Failure** – Cannot create account, display message
   1. Account already exists
   2. Invalid inputs/password does not match

### 

### Public URLS

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 7** | **Story Title: Public URLs** | **Priority: Medium** | **Estimate:** |
| As a User I want to share public links to social networking sites so that clients can complete form remotely. | | | |

**Confirmation:**

1. **Success** – Public URL provided to user
   1. Option to share directly to social networking sites
2. **Failure** – Unable to provide public URL

### Mobile Forms

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 8** | **Story Title: Mobile forms** | **Priority: High** | **Estimate:** |
| As a User I want to populate tables from the mobile app so that I can present the form to customers in person | | | |

**Confirmation:**

1. **Success** – User verified and tables associated with user displayed
   1. User selects table to populate
   2. Form presented to screen
2. **Failure** – No Tables to display
   1. User provided with option of creating new table.

### 

### Statistics

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 9** | **Story Title: Statistics** | **Priority: Medium** | **Estimate: 2 Days** |
| As a User I want to analyse table data so that I can better target and understand my clients and determine target markets | | | |

**Confirmation:**

1. **Success** – Table analytics displayed
   1. User can filter results into categories
2. **Failure** – No Analytics available
   1. Table empty
   2. No data fields in table are capable of having analytics associated with them

### Payment

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 10** | **Story Title: Payment** | **Priority: Low** | **Estimate:**  **2 Weeks** |
| As a User I want Clients to pay subscription fees in-line with the registration form. | | | |

**Confirmation:**

1. **Success** – User logs in with 3rd party payment details (e.g. PayPal, PayClick, WePay, etc.)
   1. Details verified, payment received.
2. **Failure** – Details Not verified
   1. User notified of payment failure and returned to form

### OAuth

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 11** | **Story Title: OAuth** | **Priority: Medium** | **Estimate: 2 hours** |
| As a user I want to be able to log into the service using my Social Networking accounts. | | | |

**Confirmation:**

1. **Success** – valid user logged in using social network profile and referred to personal home page
   1. User Presented with a list of their associated sheets table entries.
2. **Failure** – display message
   1. Permissions not granted, Oauth failed.
   2. Display failed login message.

### Configure LAN Server

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 12** | **Story Title: Configure LAN Server** | **Priority: High** | **Estimate: 1 Hour** |
| As a User and tester I want to connect remotely to the test server to test the application | | | |

**Confirmation:**

1. **Success** – Home page loads and application functioning from a remote client.
2. **Failure** – Timeout or failure to load homepage and application.
   1. Host unreachable etc.

### Configure WAN Server

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 13** | **Story Title: Configure WAN Server** | **Priority: High** | **Estimate: 3 Hour** |
| As a User and I want to connect remotely to the production server to use the application from any remote network | | | |

**Confirmation:**

1. **Success** – Home page loads and application functioning from a remote network
2. **Failure** – Timeout or failure to load homepage and application.
   1. Host unreachable etc.

### Error Logging

|  |  |  |  |
| --- | --- | --- | --- |
| **ID: 14** | **Story Title: Error Logging** | **Priority: High** | **Estimate: 3 Hour** |
| As a Admin and I want to view error logs of the application to analyse errors and traffic. | | | |

**Confirmation:**

1. **Success** – All errors (except 404 errors) logged and accessible.

# Implemented User Stories

### Story ID: 4 - Table Exports

User can export tables in .CSV format to a location of their choosing. Values are separated by commas and return lines for each row.

A table must first be submitted to the SQL database before it can be exported to a .CSV file. The option to export the table is not visible until the table has been submitted. Tables with no data in them are exported as a row of column headers, we may limit exports to tables that have data entries in a future iteration.

### Story ID: 8 - Mobile Forms

A user view has been created for data input into each submitted form. This view is only currently formatted for desktop browsing. The option to enter this ‘user view’ is only visible after the table has been submitted as a form to the SQL database.

Each user view is dynamically generated based on the columns present in the specified SQL Form table. There are a limited number of default layouts dependent on the input field type (as selected by the user when setting up form) in question. For example radio buttons are presented in the user view when a Boolean value (bit value) is stored in the database.

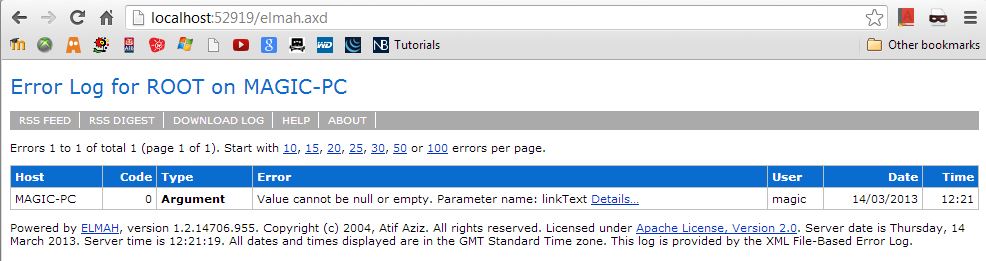
All user input fields are validated based on the expected type for the database, e.g. Numeric values only for Number fields.

### Story ID: 7 – Public URLs

The user can provide their clients with a public URL to access and fill in their forms. Clients do not need to be account holders to access this page. We see this being useful for businesses that would like to implement these forms across their social networking sites and also to advertise at events for users to access on their own devices. Currently the user must copy the URL from the provided textbox to paste where they like, however, we hope to integrate the site with Facebook/Twitter/Google in the future and provide direct sharing for the user.

### Story ID: 14 - Error Logging

We have implemented error logging using the ELMAH NuGet plugin for the application. We have this set to capture all errors and exceptions within the application excluding 404 errors. The error log is accessible through localhost/elmah.axd currently. This file will be restricted to admin access only when we implement the Admin Access user story.



# Related Progress

## Mobile Site

Mobile views of the site have been created for all views in the application. CSS is yet to be implemented for these views and only basic functionality exists here for the time being

## Hidden Options

Functionality for logged in users has been separated based on table states (submitted forms or uncommitted user tables.). Options to enter the user view for data entry and the export option are hidden from the user until the table is present (submitted) in the SQL database.

Once the table has been submitted as a form to the SQL database, the option to edit the table is no longer available; the table must be dropped and recreated for any changes to be made.

## CSS

Changes to the layout and CSS of the website from the default template have begun to take place. We see this as an ongoing process and one which will be concentrated on more so in the later iterations of the project.

# Class Diagram



# Database Schema

### Entity Framework



### SMO Database



# Macintosh HD:Users:shanemurphy:Desktop:Screen Shot 2013-03-14 at 12.41.11.pngRevised Release Plan

# Difficulties and Learning Outcomes

## Story ID: 4 - Table Exports

Some time spent seeking a code solution to separating .CSV file values into separate columns in MS Excel. Ultimately turned out to be a problem with settings in the Excel application itself.

## Story ID: 8 - Mobile Forms

Our main focus when dealing with mobile forms was dealing with user input into the system, and protecting our application from malicious or bad input. The Razor engine implemented in MVC4 automatically protects against cross site scripting (XSS) by rendering all input collected using the razor ‘@’ tag as plain text.

Cross site request forgery attacks (CSRF) can be protected against using the Html.AntiForgeryToke() method. This method issues a random token to the user to protect against this; however, it requires the use of cookies. We have not implemented this as yet, but may do so in a future iteration.

### Validation

We are using The Data Annotations Extensions package available from the NuGet installer to validate certain user inputs, e.g. the Email field required for registration. Other user input is validated using JQuery Validation. A number of different examples and methods are recommended online, leading to much time spent experimenting and trying out a number of different approaches.

## Story ID: 7 – Public URLs

In order to generate a custom URL such as localhost/magic/5, we needed to create a separate method to accept a username and retrieve the corresponding userId before passing to our original method. The problem with this is that as the user is not logged in, the SimpleMembership database has not been initialized at this point and we could not gain access to the database of users to find the UserId. We used the InitialiseSimpleMembershipAttribute(), which is a type of singleton design pattern to ensure only one instance of this connection is made.

We were unable to create a route such that the user could access the input for the tables through an URL similar to localhost/magic/5, the routing table seems to require some (any) values separated by a ‘/’ to replace as the specified default controller and action.

We settled to leave routing to localhost/public/form/username}/{tableId} for this iteration, allowing the user to access forms through localhost/public/form/shane/5 for example. We may return to this at a later iteration, but are content with results for now.

# State diagram



# Interaction diagram

