

PROJECT - REVIEWS WEB APPLICATION (PACKAGE)

OVERVIEW

The Reviews Web Application project is the capstone project for the ASP.NET Programming course. Students will begin working on the project early in the course developing the application in process that will approximate the student being hired to conceive and develop an application for an employer or client. Emphasis will be placed on both the development of the application and the process, including the skills required to communicate accurately and appropriately with an employer or client.

Given the extended timeframe of this project and the need to simulate a real world project, benchmarks are established to provide the student opportunities to communicate their status, obtain clarification and feedback from instructor, and to stay on track for the project.

OUTCOMES

- The student will write a proposal for an ASP.NET web application that will focus around the storage, management and presentation of reviews.
- The student will create a storyboard featuring the presentation views for the Reviews Application including both review and user management.
- The student will develop a model class for the Reviews Application that will generate a database using Entity Framework's Code First method.
- The student will develop MVC controllers and views for displaying and managing reviews.
- The student will develop MVC controllers and views for displaying and managing users.
- The student will create a custom theme for the Reviews Website based on the Bootstrap framework.
- The student will use model annotation for server-side validation.
- The student will use Ajax Helpers for client-side validation.
- The student will develop a UserProfile table and implement a registration and login process for security.
- The student will write a detailed peer review.
- The student will develop a presentation on their Reviews Application and present it to their peers.
- The student will reflect on the design process of the Reviews Application.

RESOURCES

- **Visual Studio Community 2015 or Visual Studio Community 2017**
- **SQL Server 2012 Management Suite or SQL Server 2016 Management Suite**
- **Microsoft Azure Account**
- **Microsoft Visio** or a similar diagramming software
- **Microsoft Word** or a similar word processor

NOTES

- This activity is designed to simulate the development environment when working for either an employer or client. The student will format and conduct all communications with regard to this project in a professional tone, whether communicating with the instructor or their peers.
- The term “item” is throughout this document to refer generically to the item/event that is the focus of the application. Students should choose appropriate names for the domain models.

PROJECT STRUCTURE AND BENCHMARKS

Conceptual Proposal

A brief narrative describing the purpose and function of the application the student will be developing. The basic functionality should be addressed with regard to viewers, users, and administrators.

Storyboard Proposal

A visual presentation of the website’s navigational layout and functionality. Panels should be detailed enough to convey the functionality of each page.

Domain Model

Using Entity Framework and Code First development, the student will code the domain model classes for the reviews in the application and generate a database from these classes. A class diagram of the domain model is required.

Database, Data, and Viewer Functionality

Using Entity Framework and Code First development, the student will generate a database from the domain model classes and enter the initial data. Viewer functionality will include searching and listing the review information by item and/or attribute.

Application Theme and UX

A complete theme using Bootstrap and customized to focus on the application will be added.

Admin Functionality

Admin functionality will include complete CRUD operations for the reviews.

Registration, Login and Security

The User model classes will be added and the database will be updated using Entity Framework Migration. Registration, login and security functionality will be implemented.

Reviewer Functionality

Reviewer functionality will include complete CRUD operations for reviews by an authenticated user.

Final Presentation

The student will prepare short (3-5 minute) class and video presentations demonstrating their web application.

Project Benchmark Point Structure

| Benchmark | Points | Due Date |
|------------------------------------------|------------|----------|
| Conceptual Proposal | 20 | |
| Storyboard Proposal | 20 | |
| Domain Model | 20 | |
| Application Theme and UX | 20 | |
| Database, Data, and Viewer Functionality | 35 | |
| Admin Functionality | 35 | |
| Registration, Login and Security | 35 | |
| Reviewer Functionality | 35 | |
| Total | 220 | |

BENCHMARK: CONCEPTUAL PROPOSAL

Objective

The objective of the Conceptual Proposal is to effectively communicate the focus and audience of the proposed web application. Attention should be paid to the specifics of the type of reviews that will be provided and the criteria for each review.

Specifications

1. Narrative describing the proposed application and audience
2. Narrative describing the structure of the reviews, i.e. database fields
3. Narrative describing the presentation of the reviews to the user including the sorting and filtering options.

Evaluation Criteria

| Criteria | Points |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| The narrative includes <ul style="list-style-type: none">• A general discussion of the purpose of the application.• A description of the audience.• A description of the reviews structure.• A description of how the reviews will be presented to the user including the sorting and filtering options. | 20 |

BENCHMARK: STORYBOARD PROPOSAL

Objective

The objective of the Storyboard Proposal is to visually present the web application's functionality including both the navigational layout and a rough view of the pages including forms and tables.

Specifications

1. Panels are included for all pages
2. Panels are connected indicating links and the application's flow
3. Posts are included as multiple panels (Posts may appear as simple links if the student is unsure of a pages posting functionality.)
4. A rough layout of forms and tables is indicated on panels

Evaluation Criteria

| Criteria | Points |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| <ul style="list-style-type: none">• All pages are included in the storyboard.• The flow from page to page is clearly indicated.• If know, page Posts are indicated as multiple panels.• Panels include a rough layout of forms and tables. | 20 |

BENCHMARK: DOMAIN MODEL

Objective

The objective of the Model benchmark is to develop the reviews domain model classes.

Specifications

1. All domain models classes and contexts are coded
2. A class diagram is provided and includes; table names, fields, primary keys and foreign keys

Evaluation Criteria

| Criteria | Points |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| <ul style="list-style-type: none">• Review model classes and contexts are coded• The class diagram includes; table names, fields, primary keys and foreign keys | 20 |

BENCHMARK: DATABASE, DATA, AND VIEWER FUNCTIONALITY

Objective

The objective of the Database, Data, and Viewer Functionality benchmark is to use Entity Framework's Code First method to generate a SQL database, enter the initial data, and develop the controllers and views for the application necessary to filter, sort and display the reviews to the viewer.

Specifications

1. A SQL database has been generated by Entity Framework
2. Initial data has been entered into the database
3. A landing page for the reviews website
4. A basic navigation system to facilitate the viewer functionality
5. A controller/view to display all reviews
6. A controller/view to filter reviews based on at least one review attribute
7. A controller/view to display a review detail

Evaluation Criteria

| Criteria | Points |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| The application includes <ul style="list-style-type: none">• A complete SQL database has been created by Entity Framework• A website landing page for the reviews website• The initial data is entered into the database• A basic navigation system to facilitate the viewer functionality• A controller/view to display all reviews• A controller/view to filter reviews based on at least one review attribute• A controller/view to display a review detail | 20 |
| Submission Requirements <ul style="list-style-type: none">• GitHub Repository (URL)• Functioning Azure Web Application (URL)• Cloud-base Persistence (Connection Information) | 15 |

BENCHMARK: APPLICATION THEME AND UX

Objective

The objective of the Application Theme and UX benchmark is to develop a complete theme using Bootstrap and customize it to meet the needs of the specific reviews application. The theme and UX should focus strongly on both the user experience and creativity.

Specifications

1. A customized bootstrap.css file with accompanying minified css and JavaScript files
2. A theme that is consistent throughout the application and with the focus of the website
3. An improved and coherent UX
4. A reflection on the specifics of the theme and UX

Evaluation Criteria

| Criteria | Points |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| <ul style="list-style-type: none">• A customized bootstrap.css file with accompanying minified css and JavaScript files are included and applied.• A theme that is consistent throughout the application and with the focus of the website is developed.• An improved and coherent UX is included.• A reflection on the specifics of the theme and UX and a discussion about how the user experience is affected. | 20 |
| Submission Requirements <ul style="list-style-type: none">• GitHub Repository (URL)• Functioning Azure Web Application (URL)• Cloud-base Persistence (Connection Information) | 15 |

BENCHMARK: ADMIN FUNCTIONALITY

Objective

The objective of the Admin Functionality will provide the administrator complete CRUD operations for the reviews.

Specifications

1. A landing page for the administrator
2. A basic navigation system to facilitate the Admin functionality
3. A basic navigation system to facilitate the Admin functionality
4. All CRUD operations for parent class (Breweries)
5. All CRUD operations for child class (Reviews)

Evaluation Criteria

| Criteria | Points |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| The application includes <ul style="list-style-type: none">• A landing page for the administrator• A basic navigation system to facilitate the Admin functionality• All CRUD operations for parent class (Breweries)• All CRUD operations for child class (Reviews) | 20 |
| Submission Requirements <ul style="list-style-type: none">• GitHub Repository (URL)• Functioning Azure Web Application (URL)• Cloud-base Persistence (Connection Information) | 15 |

BENCHMARK: REGISTRATION, LOGIN AND SECURITY

Objective

The objective of the Registration, Login and Security benchmark is to develop the user model classes and to update the database using Entity Framework Migration. Registration, login and security functionality will be implemented.

Specifications

1. The **ApplicationUser** model is extended with custom properties
2. The **AspNetRoles** table includes an Admin role and a minimum of one additional role
3. The administrative landing page contains a User Management link
4. The User Management page contains CRUD operations for users, including role management
5. A registration and login page
6. Role bases security implemented to authorize only the Admin role to access the administrative landing page.

Evaluation Criteria

| Criteria | Points |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| The application includes <ul style="list-style-type: none">• The ApplicationUser model is extended with custom properties• The AspNetRoles table includes an Admin role and a minimum of one additional role• The administrative landing page contains a User Management link• The User Management page contains CRUD operations for users, including role management• A registration and login page• Role base security implemented to authorize only the Admin role to access the administrative landing page. | 20 |
| Submission Requirements <ul style="list-style-type: none">• GitHub Repository (URL)• Functioning Azure Web Application (URL)• Cloud-base Persistence (Connection Information) | 15 |

BENCHMARK: REVIEWER FUNCTIONALITY

Objective

The objective of the Reviewer Functionality benchmark is to application pages that will allow registered Users to perform standard CRUD operations on their reviews.

Specifications

1. A landing page for the Users
2. A basic navigation system to facilitate the User functionality
3. User access to all CRUD operations of the secondary entity (reviews)
4. Role based security that provides only Users access to this functionality

Evaluation Criteria

| Criteria | Points |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| The application includes <ul style="list-style-type: none">• A landing page for the Users• A basic navigation system to facilitate the User functionality• User access to all CRUD operations of the secondary entity (reviews)• Role based security that provides only Users access to this functionality | 20 |
| Submission Requirements <ul style="list-style-type: none">• GitHub Repository (URL)• Functioning Azure Web Application (URL)• Cloud-base Persistence (Connection Information) | 15 |

BENCHMARK: PEER REVIEW

Objective

The objective of the Peer Review benchmark is to provide the student with the opportunity practice critiquing skills for web application. The student will take a critical look a peer website and provide the peer with an object and detailed review.

Specifications

1. Complete the peer review for a website
2. Provide the peer review to the website's developer
3. Respond to the peer review

Evaluation Criteria

| Criteria | Points |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| <ul style="list-style-type: none">• Completed peer review for a website is included• Peer review provided to the website's developer• Respond to the peer review | 20 |

CAPSTONE PROJECT: DEPLOYED APPLICATION, PRESENTATIONS, AND REFLECTION

Objective

The objective of the capstone project is for the student to complete a fully functional web application including a persistence element and to deploy the application and persistence into the cloud. The benchmark will also focus on providing the student an opportunity to develop a formal presentation and to create a short, three to five minute, video presentation demonstrating their finished web application. The video presentation will be developed in a manner consistent with providing an artifact for their portfolio.

Marking Guides Overview

| Requirement | Points |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| Application Submission Requirements <ul style="list-style-type: none"> • GitHub Repository (URL) • Functioning Azure Web Application (URL) • Cloud-base Persistence (Connection Information) | 40 |
| Level I Features and Requirements <ul style="list-style-type: none"> • Item and Item Review Entities • Open Application • Item Admin: List, Search, Detail, Filter, CRUD • Reviews Admin: List, Search, Detail, Filter, CRUD • Security Issues Addressed • Customized, Consistent, and Appropriate Theme • Consistent and Functional Navigation and Flow for Users • Static Images and Aside • Pages: About and Contact | 80 |
| Level II Features and Requirements <ul style="list-style-type: none"> • User Entity • Admin and Reviewer Authorization • Users Admin: List, Search, Detail, Filter, CRUD • Roles: CRUD • User Role Management • Carousel and Dynamic Aside | 20 |
| Level III Features and Requirements <ul style="list-style-type: none"> • Event Entity • Events Admin: List, Search, Detail, Filter, CRUD • Reviews Admin [Reviewers]: List, Search, Detail, Filter, CRUD • Item, Reviews, or Events w/ Images • Third-party Controls | 20 |
| Web Application Checklist | 10 |
| Video Presentation | 10 |
| Class Presentation | 10 |
| Reflection Document | 10 |
| Total Possible Points | 200 |

APPLICATION SUBMISSION REQUIREMENTS

| Requirement | Points |
|-------------------------------------------------|--------|
| GitHub Repository (URL) | 10 |
| Functioning Azure Web Application (URL) | 15 |
| Cloud-base Persistence (Connection Information) | 15 |
| | 40 |

WEB APPLICATION CHECKLIST REQUIREMENTS

| Requirement | Points |
|-------------------------------------------------|--------|
| Checklist Submitted | 2 |
| Functioning Azure Web Application (URL) | 4 |
| Cloud-base Persistence (Connection Information) | 4 |
| | 10 |

VIDEO PRESENTATION REQUIREMENTS

| Requirement | Points |
|---------------------------------------------------------------------------|--------|
| All features are demonstrated. | 2 |
| The presentation's flow is organized and easy to follow | 2 |
| The presenter's voice is clear. | 2 |
| The presenter speaks in complete sentences. | 2 |
| The view is left with a coherent understanding of the entire application. | 2 |
| | 10 |

CLASS PRESENTATION REQUIREMENTS

| Requirement | Points |
|---------------------------------------------------------------------------|--------|
| All features are demonstrated. | 2 |
| The presentation's flow is organized and easy to follow | 2 |
| The presenter's voice is clear. | 2 |
| The presenter speaks in complete sentences. | 2 |
| The view is left with a coherent understanding of the entire application. | 2 |
| | 10 |

REFLECTION DOCUMENT REQUIREMENTS

Overview

The focus of the course is the development and deployment of a working web application that includes database connectivity and administrative functionality. The application choice is student driven and each student is encouraged to explore and extend their application beyond the basic requirements. A key course outcome is the final application, video presentation, and a reflection document representing a portfolio artifact.

Requirements

The reflection document should follow good writing conventions and rules.

The student should address the following items within the reflection document.

- Describe the purpose and functionality of the web application.
- State and discuss elements of strength within the web application.
- State and discuss elements of weakness within the web application.
- Discuss specific additions/modification to be included in the next version of the web application.
- Reflect on the development process.
 - Discuss concepts that were unclear at the beginning and became clearer as the semester progress.
 - Discuss concepts that still remain unclear.
 - Describe at least one concept or feature that could/should be included in the project.
- Discuss the value of the project as a learning experience.

FEATURES AND REQUIREMENTS SPECIFICATION SHEET FOR WEB APPLICATION

| | Level I | Level II (include all Level I requirements) | Level III (include all Level II requirements) |
|--------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| Models, Database, and Relationships | <input type="checkbox"/> One-Many: Breweries to ItemReviews <input type="checkbox"/> Item: 6 properties <input type="checkbox"/> ItemReview: 3 properties | <input type="checkbox"/> Many-Many: AspNetUsers to AspNetRoles <input type="checkbox"/> Extended AspNetUsers: 3 properties | <input type="checkbox"/> One-Many: Breweries to Events <input type="checkbox"/> One-Many: Reviewers to Reviews |
| Data | <input type="checkbox"/> Item: 6 properties, 20 entries <input type="checkbox"/> ItemReview: 20 entries across all breweries | <input type="checkbox"/> AspNetRoles: Admin, Reviewer <input type="checkbox"/> Extended AspNetUsers: 5 entries | <input type="checkbox"/> Events: 6 entries across all breweries |
| Navigation | <input type="checkbox"/> All Users: Home, Breweries, About, Contact, Admin [no authorization] | <input type="checkbox"/> Admin: Breweries, Reviews, Users <input type="checkbox"/> Reviewer: Register, Login, Profile | <input type="checkbox"/> Admin: Events <input type="checkbox"/> Reviewer: Reviews |
| Viewer Functionality | <input type="checkbox"/> Breweries: List, Search, Filter, Detail <input type="checkbox"/> Reviews: List, Search, Filter, [Detail] | | <input type="checkbox"/> Events: List, Detail |
| Admin Functionality | <input type="checkbox"/> Breweries: List, Search, Detail, Filter, CRUD <input type="checkbox"/> Reviews: List, Search, Detail, Filter, CRUD | <input type="checkbox"/> Users: List, Search, Detail, Filter, CRUD <input type="checkbox"/> Roles: CRUD <input type="checkbox"/> Manage User Roles | <input type="checkbox"/> Events: List, Search, Detail, Filter, CRUD |
| Reviewer Functionality | | <input type="checkbox"/> Register and Login <input type="checkbox"/> Profile: Edit | <input type="checkbox"/> Reviews: List, Search, Detail, Filter, CRUD |
| Security | <input type="checkbox"/> Client Side Validation <input type="checkbox"/> Server Side Validation <input type="checkbox"/> Cross Site Scripting <input type="checkbox"/> HTML Sterilization <input type="checkbox"/> Over Posting | <input type="checkbox"/> Reviewer Authorization <input type="checkbox"/> Admin Authorization | |
| Theme and UX | <input type="checkbox"/> Customized, Consistent, and Appropriate Theme <input type="checkbox"/> Consistent and Functional Navigation and Flow for Viewers | <input type="checkbox"/> Consistent and Functional Navigation and Flow for Reviewers <input type="checkbox"/> Consistent and Functional Navigation and Flow for Admins | |
| Features | <input type="checkbox"/> Static Images <input type="checkbox"/> Static Aside <input type="checkbox"/> Pages: About and Contact | <input type="checkbox"/> Dynamic Aside (Database Controlled) | <input type="checkbox"/> Item or Event Images |
| Third-party Controls | | <input type="checkbox"/> Carousel | <input type="checkbox"/> Input or Visualization: i.e. Datepicker, Rating, etc. (Syncfusion) |
| | 80 Points | 20 Points | 20 Points |

FEATURES AND REQUIREMENTS CHECKLIST: LEVEL I

Note: The Student Comment section **MUST** be completed for each element that was not fully completed. Discuss the specific issues encountered that prevented the completion of this element. Also, describe any elements that may deviate from the specific requirements.

| | Level I | Student Comments | Points | Self-Score |
|--------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|-----------|------------|
| Models, Database, and Relationships | <input type="checkbox"/> One-Many: Item to ItemReview <input type="checkbox"/> Item: 6 properties <input type="checkbox"/> ItemReview: 3 properties <input type="checkbox"/> Extended AspNetUsers: 3 properties, 5 entries | | 20 | |
| Data | <input type="checkbox"/> Item: 6 properties, 20 entries <input type="checkbox"/> ItemReview: 20 entries across all breweries | | 10 | |
| Navigation | <input type="checkbox"/> All Users: Home, Breweries, About, Contact, Admin [no authorization] | | 5 | |
| Viewer Functionality | <input type="checkbox"/> Breweries: List, Search, Filter, Detail <input type="checkbox"/> Reviews: List, Search, Filter, [Detail] | | 10 | |
| Admin Functionality | <input type="checkbox"/> Breweries: List, Search, Detail, Filter, CRUD <input type="checkbox"/> Reviews: List, Search, Detail, Filter, CRUD | | 20 | |
| Reviewer Functionality | | | | |
| Security | <input type="checkbox"/> Client Side Validation <input type="checkbox"/> Server Side Validation <input type="checkbox"/> Cross Site Scripting <input type="checkbox"/> HTML Sterilization <input type="checkbox"/> Over Posting | | 5 | |
| Theme and UX | <input type="checkbox"/> Customized, Consistent, and Appropriate Theme <input type="checkbox"/> Consistent and Functional Navigation and Flow for Viewers | | 5 | |
| Features | <input type="checkbox"/> Static Images <input type="checkbox"/> Static Aside <input type="checkbox"/> Pages: About and Contact | | 5 | |
| | | | 80 | |

FEATURES AND REQUIREMENTS CHECKLIST: LEVEL II

Note: The Student Comment section MUST be completed for each element that was not fully completed. Discuss the specific issues encountered that prevented the completion of this element. Also, describe any elements that may deviate from the specific requirements.

| | Level II (include all Level I requirements) | Student Comments | Points | Self-Score |
|--------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------|---------------|-------------------|
| Models, Database, and Relationships | <input type="checkbox"/> Many-Many: AspNetUsers to AspNetRoles <input type="checkbox"/> Extended AspNetUsers: 3 properties | | 2 | |
| Data | <input type="checkbox"/> AspNetRoles: Admin, Reviewer <input type="checkbox"/> Extended AspNetUsers: 5 entries | | 1 | |
| Navigation | <input type="checkbox"/> Admin: Breweries, Reviews, Users <input type="checkbox"/> Reviewer: Register, Login, Profile | | 2 | |
| Viewer Functionality | | | | |
| Admin Functionality | <input type="checkbox"/> Users: List, Search, Detail, Filter, CRUD <input type="checkbox"/> Roles: CRUD <input type="checkbox"/> User Role Management | | 5 | |
| Reviewer Functionality | <input type="checkbox"/> Register and Login <input type="checkbox"/> Profile: Edit | | 2 | |
| Security | <input type="checkbox"/> Reviewer Authorization <input type="checkbox"/> Admin Authorization | | 3 | |
| Theme and UX | <input type="checkbox"/> Consistent and Functional Navigation and Flow for Reviewers <input type="checkbox"/> Consistent and Functional Navigation and Flow for Admins | | 2 | |
| Features | <input type="checkbox"/> Dynamic Aside (Database Controlled) | | 2 | |
| Third-party Controls | <input type="checkbox"/> Carousel | | 1 | |
| | | | 20 | |

FEATURES AND REQUIREMENTS CHECKLIST: LEVEL III

Note: The Student Comment section **MUST** be completed for each element that was not fully completed. Discuss the specific issues encountered that prevented the completion of this element. Also, describe any elements that may deviate from the specific requirements.

| | Level III (include all Level II requirements) | Student Comments | Points | Self-Score |
|--------------------------------------------|-------------------------------------------------------------------------------------------------------------------|-------------------------|---------------|-------------------|
| Models, Database, and Relationships | <input type="checkbox"/> One-Many: Breweries to Events <input type="checkbox"/> One-Many: Reviewers to Reviews | | 5 | |
| Data | <input type="checkbox"/> Events: 6 entries across all breweries | | 1 | |
| Navigation | <input type="checkbox"/> Admin: Events <input type="checkbox"/> Reviewer: Reviews | | 2 | |
| Viewer Functionality | <input type="checkbox"/> Events: List, Detail | | | |
| Admin Functionality | <input type="checkbox"/> Events: List, Search, Detail, Filter, CRUD | | 2 | |
| Reviewer Functionality | <input type="checkbox"/> Reviews: List, Search, Detail, Filter, CRUD | | 5 | |
| Security | | | | |
| Theme and UX | | | | |
| Features | <input type="checkbox"/> Item or Event Images | | 3 | |
| Third-party Controls | <input type="checkbox"/> Input or Visualization: i.e. Datepicker, Rating, etc. (Syncfusion) | | 2 | |
| | | | 20 | |