
e-Valuate

Iteration Plan 2

Version 1.1.0

e-Valuate	Version: 1.1.0
Iteration Plan 2	Date: 06/May/20

Revision History

Date	Version	Description	Author
31/Oct/19	0.0.1	Created an initial draft of each section	Jordyn Dent
27/Dec/19	0.1.0	Reorganize Introduction Develop a detailed Plan/timeline Reorganize Resources Detail use cases Create empty task tables Detail the evaluation criteria	Kenny Houston
03/Jan/20	0.2.0	Add Use Case Diagram	Kenny Houston
05/Jan/20	0.2.1	Add Class Diagram	Kenny Houston
17/Jan/20	0.3.0	Modify Class Diagram to include encryptedPassword	Kenny Housto
20/Jan/20	0.4.0	Update the expected date for Use Case 12	Hau Ha
21/Jan/20	0.4.1	Update task list for UC02	Jordyn Dent
22/Jan/20	0.4.2	Modify end of iteration date	Kenny Houston
22/Jan/20	0.4.3	Update expected date for use case 15	Megan Phan
28/Jan/20	0.5.0	Update class diagram from client meeting	Kenny Houston
08/Feb/20	1.0.0	Fix grammar in Purpose section	Kenny Houston
06/May/20	1.1.0	Final touch ups and corrections	Kenny Houston

e-Valuate	Version: 1.1.0
Iteration Plan 2	Date: 06/May/20

Table of Contents

Introduction	5
Purpose	5
Scope	5
Definitions, Acronyms, and Abbreviations	5
References	5
Glossary	5
UC01 Create an Account	5
UC02 Log in to an Account	5
UC03 Create an Event	5
UC04 Create a Rubric	5
UC12 Access a List of Assigned Entries	5
UC13 Grade an Entry	5
UC15 Register for an event	5
Overview	5
Plan	5
28/Dec/19 - Design	5
06/Jan/20 - Development	5
17/Jan/20 - Testing	6
24/Jan/20 - Completion	6
Resources	6
Cost	6
Tools	6
Spring boot	6
Maven	6
MySQL	6
Hibernate	6
ReactJS	6
A Login API - TBD	6
Use Cases	6
UC01 Create an Account	6
UC02 Log in to an Account	6
UC03 Create an Event	7
UC12 Access a List of Assigned Entries	7
UC15 Register for an Event	8
Evaluation Criteria	8
Responsive	8
Ease of Use	8
Reliability	8

e-Valuate	Version: 1.1.0
Iteration Plan 2	Date: 06/May/20

Diagrams	9
Use Case Diagram	9
Class Diagram	10

e-Valuate	Version: 1.1.0
Iteration Plan 2	Date: 06/May/20

Iteration Plan 2

1. Introduction

1.1 Purpose

The purpose of Iteration 2 is to make e-Valuate more flexible and user/event-oriented. The goal of Iteration 1 was to give e-Valuate the basic functions necessary for it to run, but that means leaving out many of the features that tailor it to individual users. After Iteration 2, all data will be associated with specific users and/or specific events. Iteration 2 is where e-Valuate becomes fully functional, though obviously still limited in some ways.

1.2 Scope

This iteration encompasses five different use cases. Many of these use cases are deeply connected to each other and to previously implemented use cases. Linking web pages properly for a smooth and enjoyable user experience is crucial.

1.3 Definitions, Acronyms, and Abbreviations

All terms can be found in the glossary.

1.4 References

1.4.1 *Glossary*

1.4.2 *UC01 Create an Account*

1.4.3 *UC02 Log in to an Account*

1.4.4 *UC03 Create an Event*

1.4.5 *UC04 Create a Rubric*

1.4.6 *UC12 Access a List of Assigned Entries*

1.4.7 *UC13 Grade an Entry*

1.4.8 *UC15 Register for an event*

1.5 Overview

Iteration 2 builds off of Iteration 1 in that it focuses on more required features for e-Valuate. The purpose of Iteration 2 is to make e-Valuate more flexible and user/event-oriented. We are doing this by focusing on developing five use cases: UC01 Create an Account, UC02 Log in to an Account, UC03 Create an Event, UC12 Access a List of Assigned Entries, and UC15 Register for an Event. These five use cases come together to expand the functionality of the use cases developed in Iteration 1 while making e-Valuate resemble a standalone product.

The rest of this iteration plan is devoted to presenting the timeline for this iteration as well as resources and requirements.

2. Plan

2.1 28/Dec/19 - Design

The team will nail down the final requirements for this iteration. This includes modifying use cases and specifying tasks to be completed during development. Any diagrams or models will also be completed in this phase.

2.2 06/Jan/20 - Development

Individual work on assigned use cases begins here. The development related tasks created in the previous phase will be completed during this phase. This starts with developing the new back end models necessary for representing new information. Later, this includes developing the front end of each corresponding use case, as well as the interactions between the use cases.

Some testing will be done on an individual level. Also, when significant progress is made, the specific implementation will be approved by the client, so course corrections can be made as soon as possible.

e-Valuate	Version: 1.1.0
Iteration Plan 2	Date: 06/May/20

2.3 17/Jan/20 - Testing

Each use case must be tested vigorously by team members who did not develop them. This will ensure total functionality as well as help us avoid bugs and implementation errors. The final implementations will also need to be approved by the client.

2.4 24/Jan/20 - Completion

The iteration will be approved and complete by 31/Jan/20.

3. Resources

New resources for this iteration may include API's to assist with the login procedure.

3.1 Cost

We now have an underlying system which development will build on top of. The greatest costs during this iteration will likely come from the time it takes to weave the different use cases and pages together in a way which produces a well designed product. However, individual time costs will likely be less than Iteration 1 now that the application is being added to, rather than being built from scratch.

3.2 Tools

3.2.1 *Spring boot*

3.2.2 *Maven*

3.2.3 *MySQL*

3.2.4 *Hibernate*

3.2.5 *ReactJS*

3.2.6 *A Login API - TBD*

4. Use Cases

4.1 UC01 Create an Account

This Use Case will be taking advantage of our account API. The API will have to be integrated with our backend system to account for the links between accounts and other models. Accounts can play many roles in the form of Judges, Chairs, and possibly Contestants.

This use case has been assigned to Action.

Task	Assignee	Expected Start	Expected End	Actual Start	Actual End
Design Page	Action	05/Jan/20	10/Jan/20	05/Jan/20	10/Jan/20
Develop frontend	Action	13/Jan/20	18/Jan/20	13/Jan/20	22/Jan/20
Develop services and controllers	Action	22/Jan/20	28/Jan/20	23/Jan/20	10/Feb/20
Testing Period	All Members	28/Jan/20	01/Feb/20	10/Feb/20	11/Feb/20

4.2 UC02 Log in to an Account

When a user logs into their account, it is imperative that they only be able to access information associated

e-Valuate	Version: 1.1.0
Iteration Plan 2	Date: 06/May/20

with their account. This means, limiting their privileges regarding event details, limiting which entries can be graded, and restricting their access to score reports.

Some research is likely necessary to determine how to most safely implement these features.

This use case has been assigned to Jordyn.

Task	Assignee	Expected Start	Expected End	Actual Start	Actual End
Design page					
Develop frontend for the login page					
Research user privilege management					
Develop service and controller classes					

4.3 UC03 Create an Event

This use case will require some integration with UC04 Create a Rubric. From this iteration on, a rubric will be associated with one, and only one event. We may consider, down the line, the ability for chairs to copy rubrics from other events they manage. But, as of now, rubrics are always created from scratch, for a particular event.

This is a good opportunity to implement a hub for a given event from the chair's perspective. It could include links to the rubric creation page, and eventually include links to judge invitations and the like.

This use case has been assigned to Jackie.

Task	Assignee	Expected Start	Expected End	Actual Start	Actual End
Frontend coding	Jackie	21/Jan/20	26/Jan/20		
Backend coding	Jackie	21/Jan/20	26/Jan/20		

4.4 UC12 Access a List of Assigned Entries

This use case will require integration with UC13 Grade and Entry. It will be necessary to make sure that the wrong entries cannot be graded or that the right entries cannot be graded with the wrong rubrics. These details center around the event model and the security of the system.

This use case has been assigned to Hau.

Task	Assignee	Expected Start	Expected End	Actual Start	Actual End
Design/draw the page	Hau	05/Jan/20	10/Jan/20	05/Jan/20	10/Jan/20
Build frontend Access List of Assigned Entries page	Hau	12/Jan/20	19/Jan/20	12/Jan/20	19/Jan/20
Build ListofAssignedEntries Controller and	Hau	20/Jan/20	26/Jan/20	20/Jan/20	26/Jan/20

e-Valuate	Version: 1.1.0
Iteration Plan 2	Date: 06/May/20

Services classes					
Connect backend to frontend	Hau	20/Jan/20	26/Jan20	20/Jan/20	31/Jan/20
Testing Period	All Member	27/Jan/20	31/Jan/20	01/Feb/20	07/Feb/20

4.5 UC15 Register for an Event

This use case may require more discussion for the final implementation. How should an event be found by a contestant? Should they be able to search by name? Search by event ID or a similarly unique code? These decisions should be indicated in the Use Case documentation.

Once an event is chosen, registration is straight forward.

This use case has been assigned to Megan.

Task	Assignee	Expected Start	Expected End	Actual Start	Actual End
Design the web page	Megan	06/Jan/20	10/Jan/20	10/Jan/20	15/Jan/20
Implement the front end part	Megan	10/Jan/20	20/Jan/20	15/Jan/20	25/Jan/20
Build RegisterEvent controller and services	Megan	20/Jan/20	27/Jan/20	20/Jan/20	01/Feb/20
Connect front end with back end	Megan	20/Jan/20	27/Jan/20	01/Feb/20	04/Feb/20
Get the event's details from the database	Megan	20/Jan/20	27/Jan/20	04/Feb/20	05/Feb/20
Testing	Megan	27/Jan/20	30/Jan/20	05/Feb/20	07/Feb/20

5. Evaluation Criteria

5.1 Responsive

The system needs to react to the user's inputs quickly. If a user were to become frustrated with slow response times, they would be less likely to continue using the system in the future. A slow electronic system can quickly become more frustrating than doing everything by hand on physical paper.

5.2 Ease of Use

The users need to be able to navigate the system without the guidance of a third party. We will present the web app to the client, and provide no other guidance other than what the built-in system provides. If they can accomplish all the desired tasks without becoming frustrated, then the system will be deemed easy enough to use.

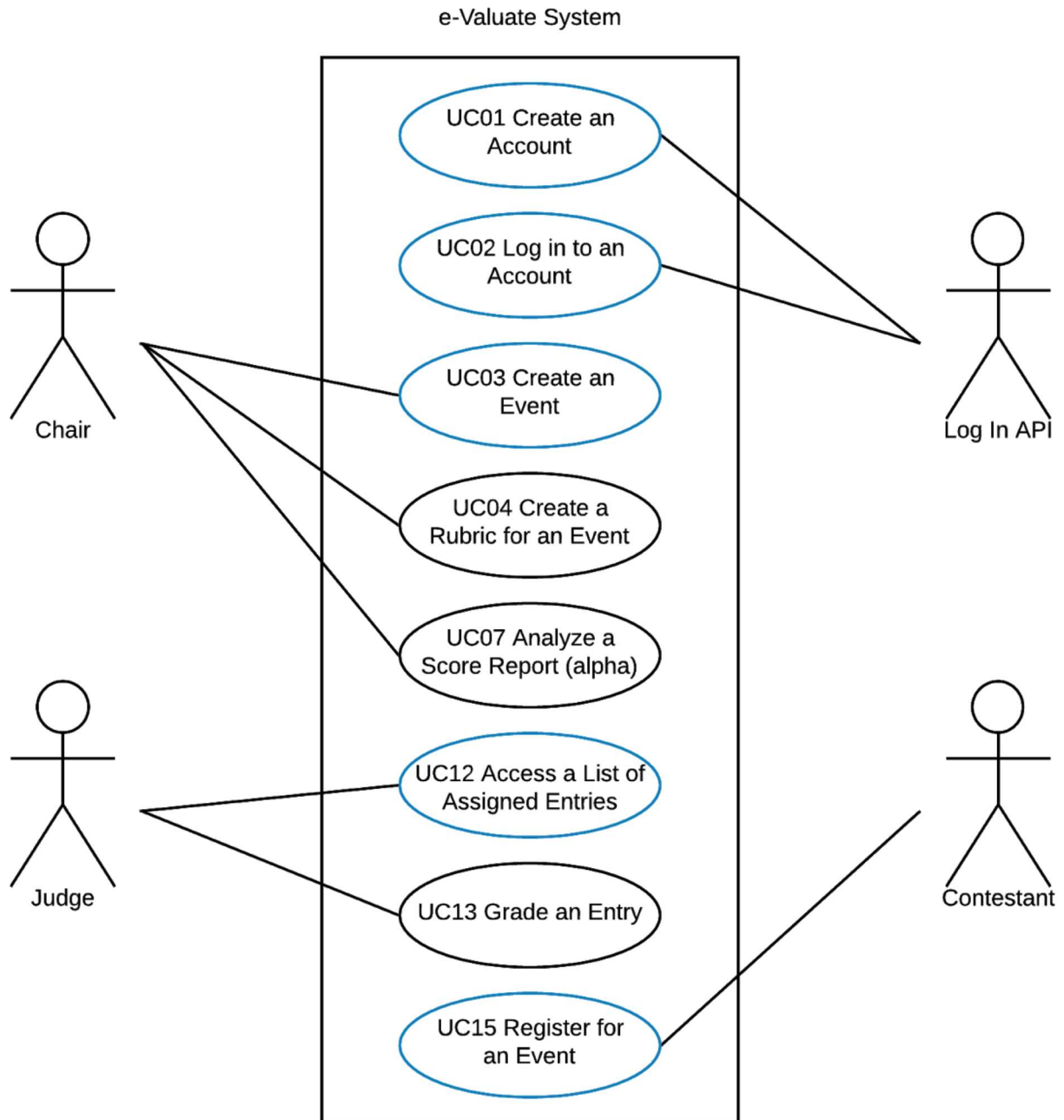
5.3 Reliability

The user must be able to have full confidence that the system will save all the required information without fail. Or at the very least, convey clearly to the user when information has failed to save properly.

e-Valuate	Version: 1.1.0
Iteration Plan 2	Date: 06/May/20

6. Diagrams

6.1 Use Case Diagram



Use cases in blue are new to this iteration.

e-Valuate	Version: 1.1.0
Iteration Plan 2	Date: 06/May/20

6.2 Class Diagram

