

# *ITERATION 2* *ARTIFACTS*

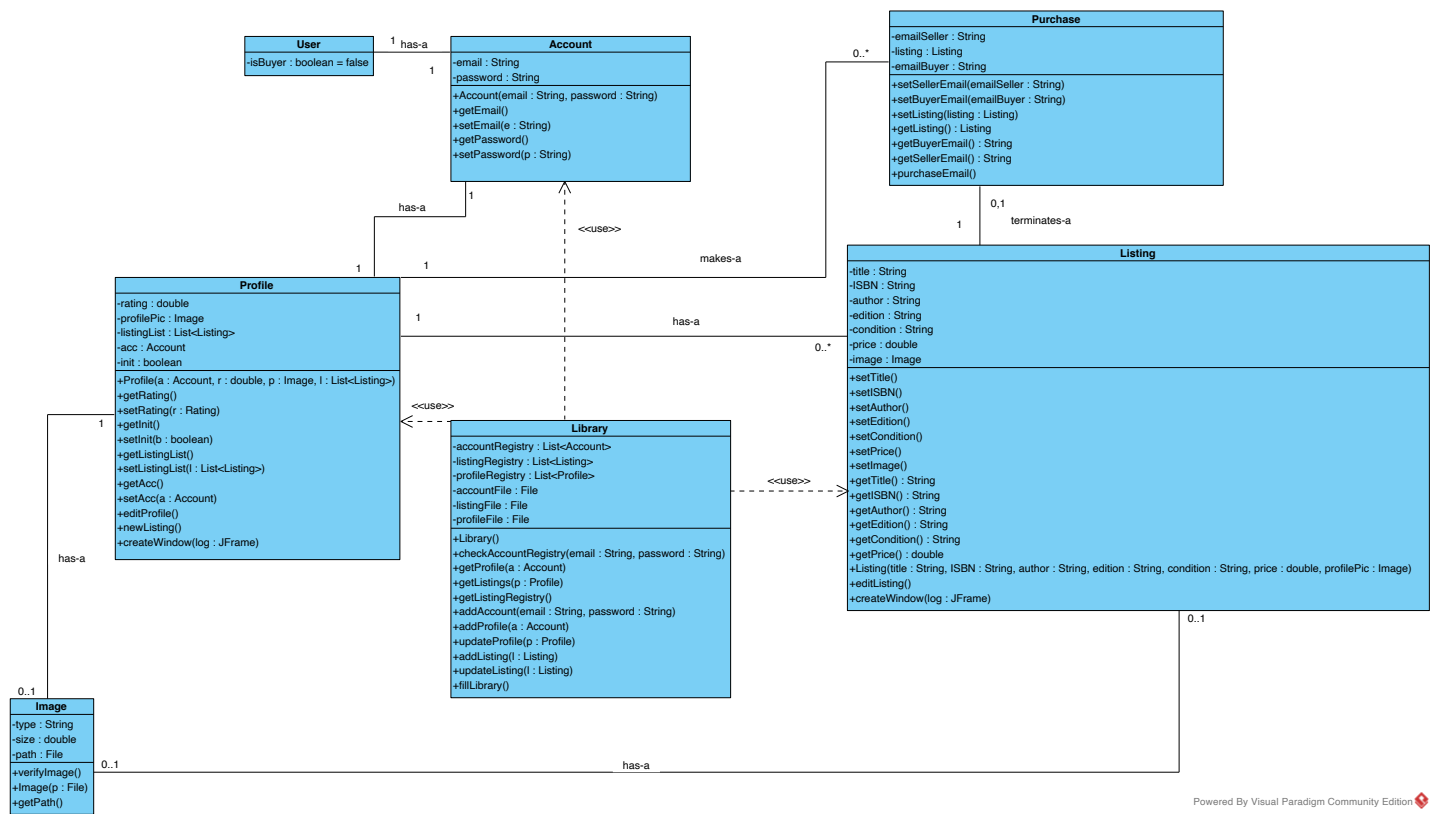
Tom Pechulis  
Chris Torres  
Tyler Crumrine



# Table of Contents

Class Diagram	3
GRASP	5
Sequence Diagrams	8
Test Coverage Plan	18
User Interface	20
Timecards Report	24
Gantt Diagram	26
Links	33

# Class Diagram



GRASP

# GRASP

## Classes:

- Account
  - the Account class is only responsible for its own data members, so it only requires getters and setters
- Profile
  - newListing: each user has to be able to publish a listing to the system, and it needs to be linked to their profile
  - editProfile: user has ability to edit profile picture and password when logged in. This is an attribute of the Profile, not the Account because it is the information displayed on the home page to the user.
- Image
  - verifyImage: any file a user passes get initialized as an image since it only requires a file path to be passed; calling this method will allow the system to make sure a file is the correct type and size before assigning it to the Profile/Listing
- Purchase
  - purchaseEmail: When a purchase is requested, the seller is notified via email with the buyer's contact information and information on which listing is being purchased
- Listing
  - editListing: This allows the user to change the information of a currently active listing, including updating title, author, and even image.
- Library
  - checkAccountRegistry: used for sign in, allows system to see if user account credentials are valid
  - addAccount: when a request for a new account to be created is made, the system adds it to the relevant file
  - addProfile: adds a profile for a newly created account to the relevant file
  - updateProfile: if the user edits their profile information, this is used afterwards to update the relevant file information to match
  - addListing: creates a new instance of Listing and adds the information to the relevant file
  - updateListing: functions the same as updateProfile, but for the Listing file
  - fillLibrary(): fills lists with information from the files once the application is run

## Patterns:

- Creator
  - Both the Profile and Listing classes act as a Creator for the Image class. Images only exist in the instance of a user updating their profile picture, or adding an image to a new or existing book listing.
- High Cohesion

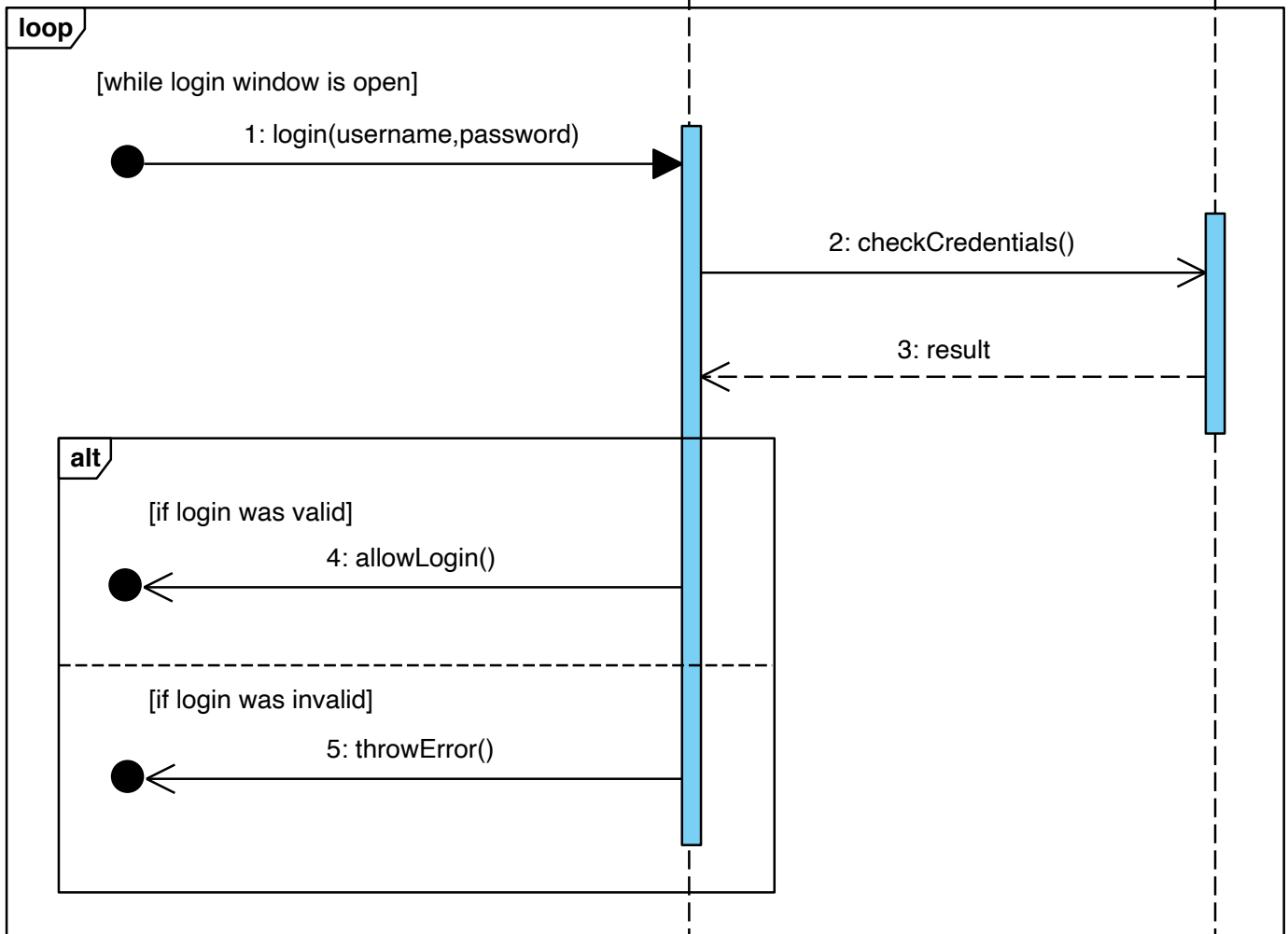
- The classes are divided in such a way that each one is only responsible for a specific task. The least cohesive class is probably Profile since it has to be responsible for hosting the current account, active listings, and give the user the ability to edit both of these classes.
- Information Expert
  - The Library class acts as a place to store all information for the system. It has the responsibility of tracking created accounts, their associated profiles, and their associated listings, as well as updating the files storing the information.

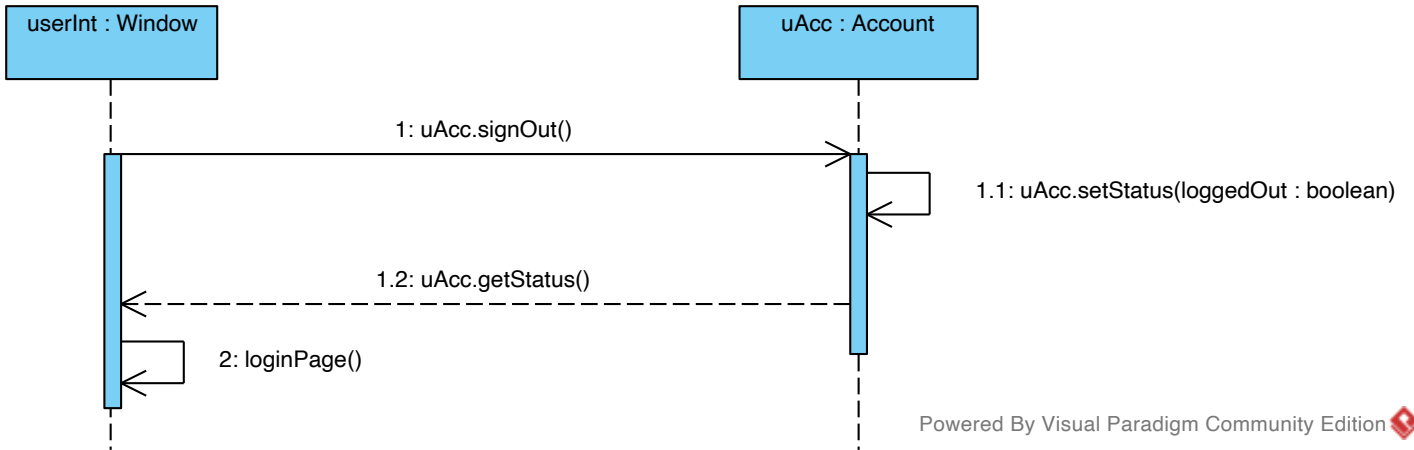
# Sequence Diagrams

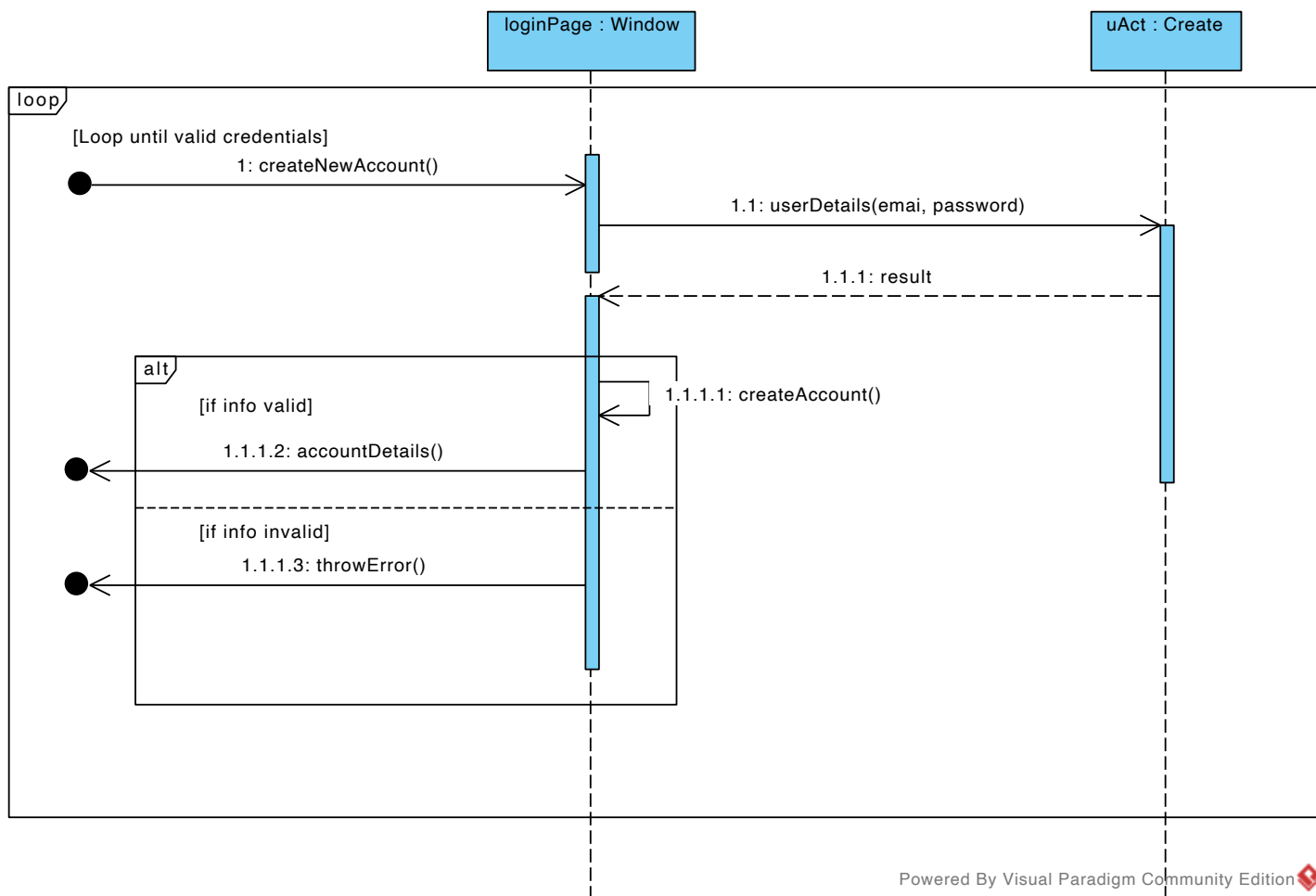


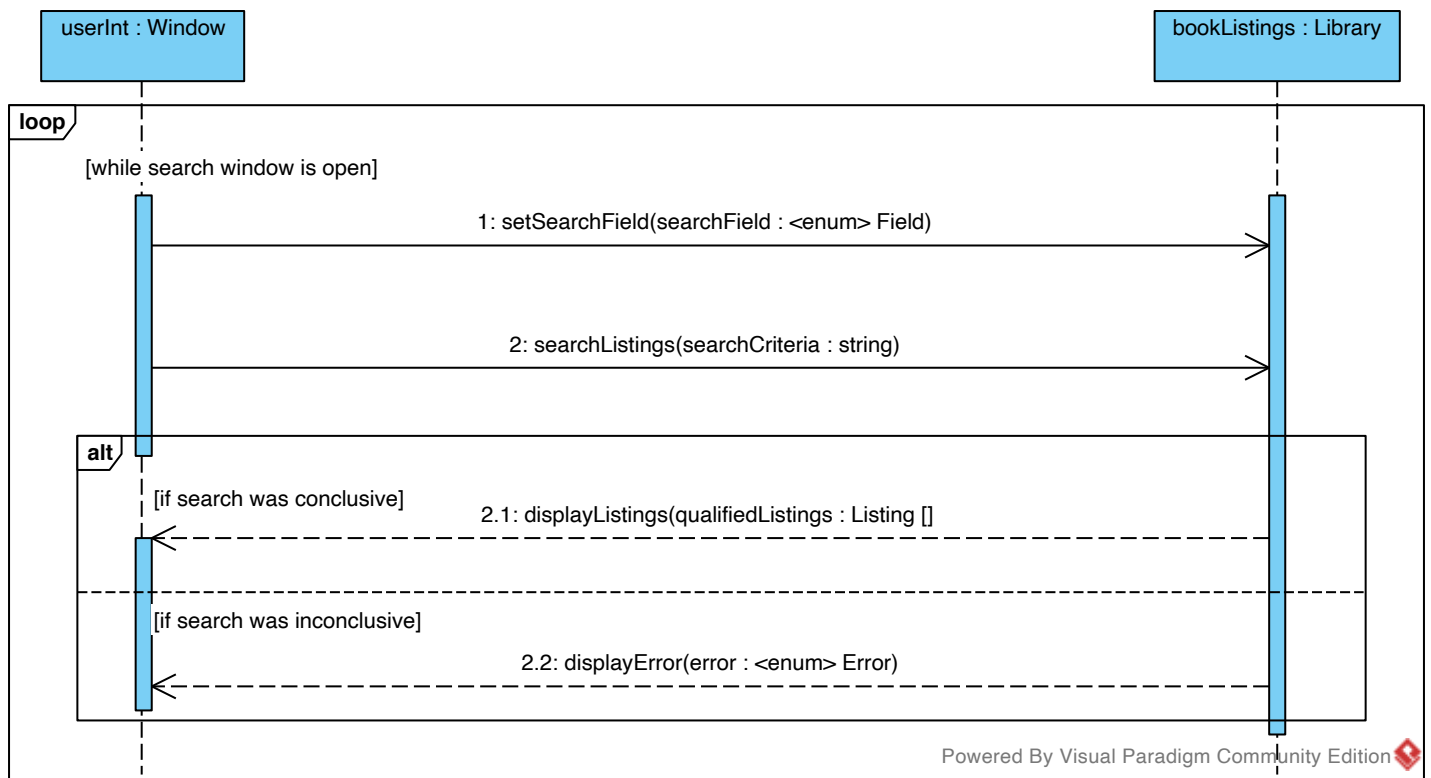
loginPage : Login

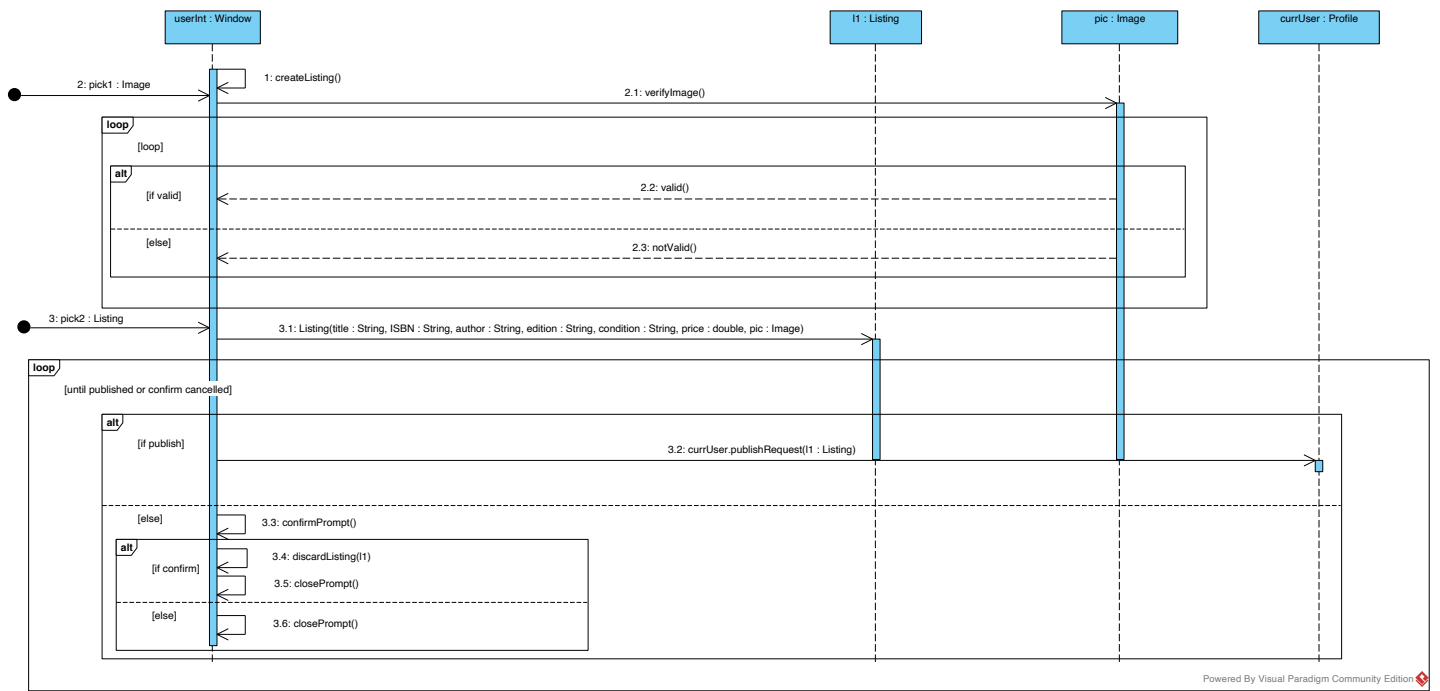
uAct : Account

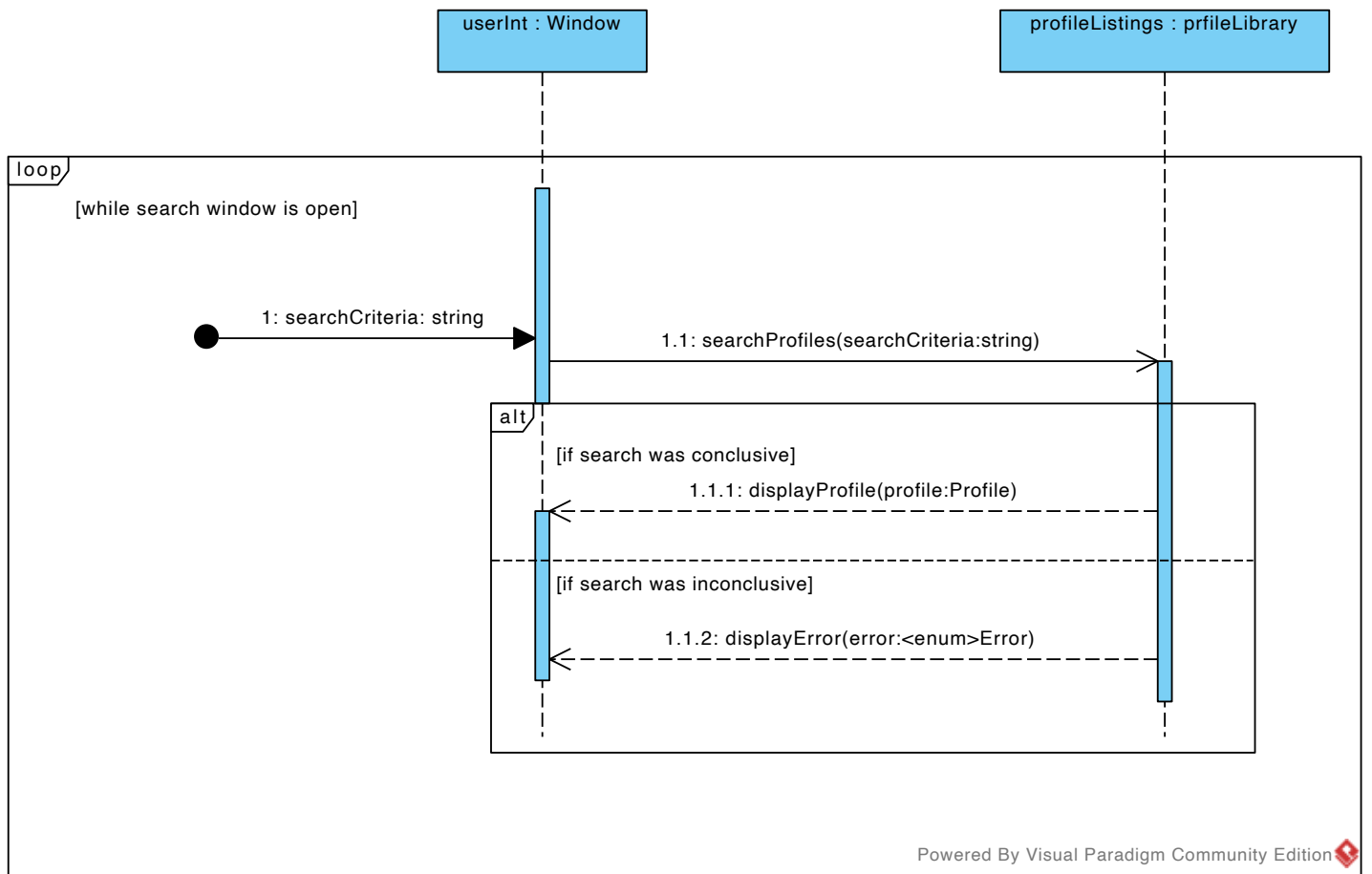


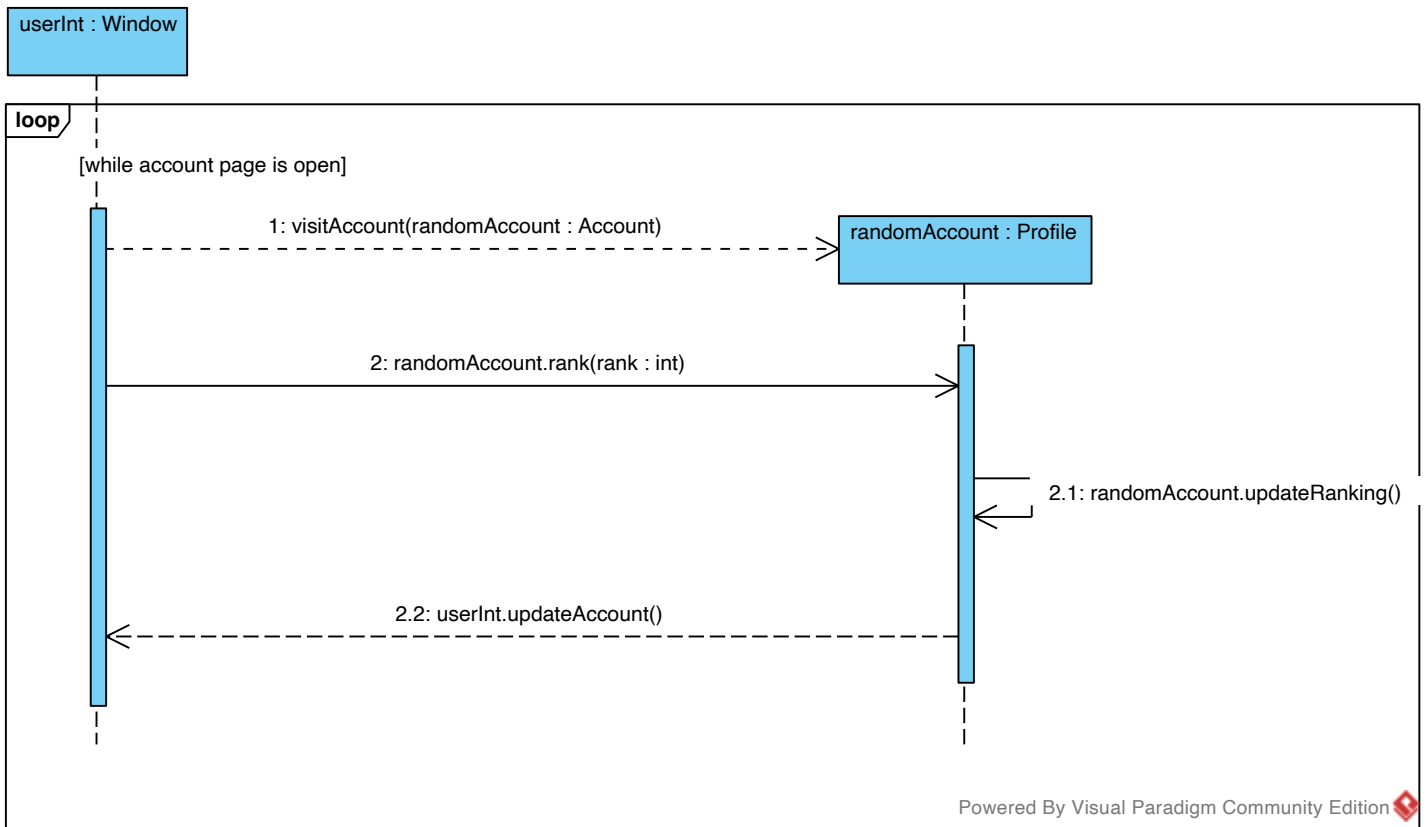


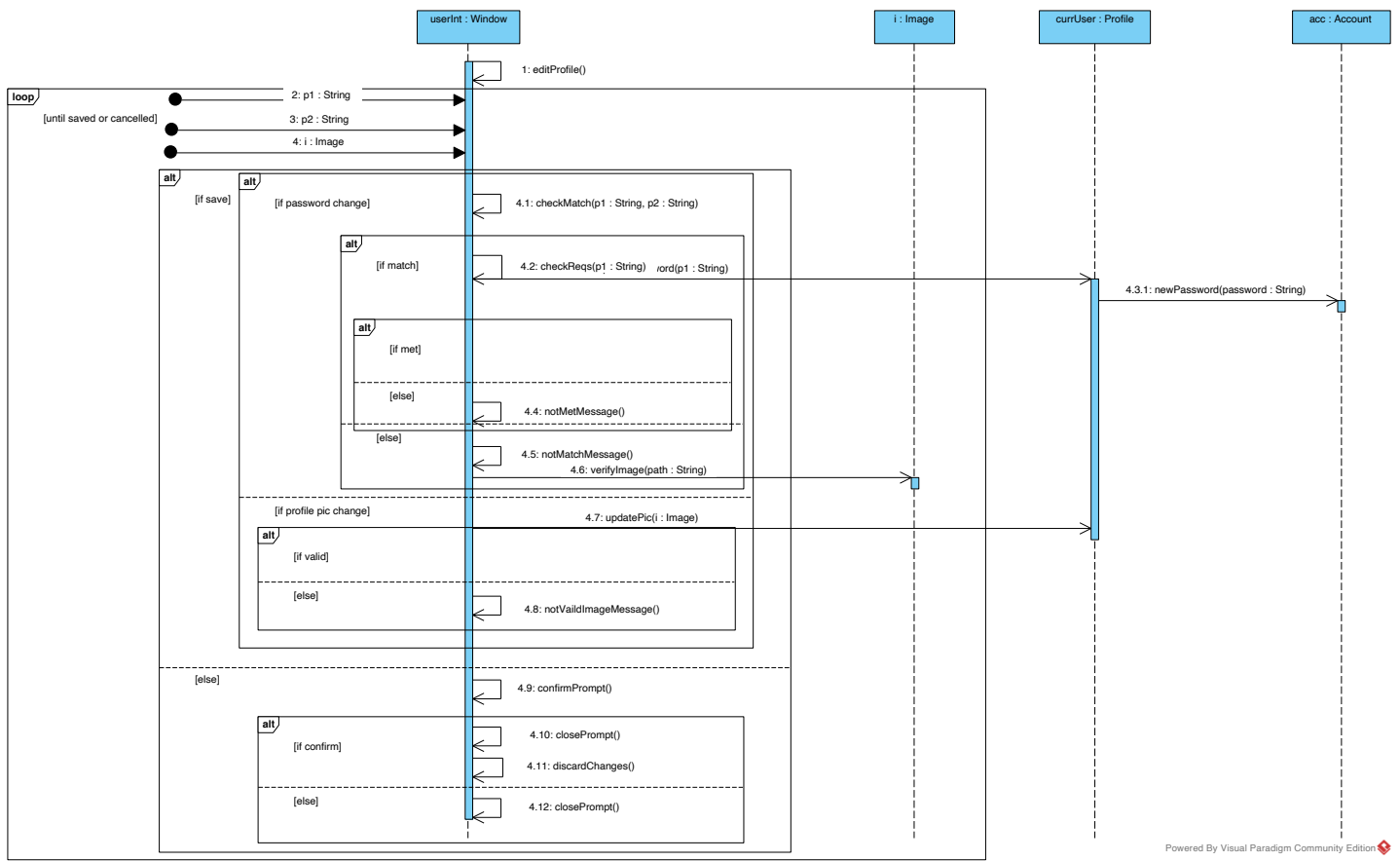




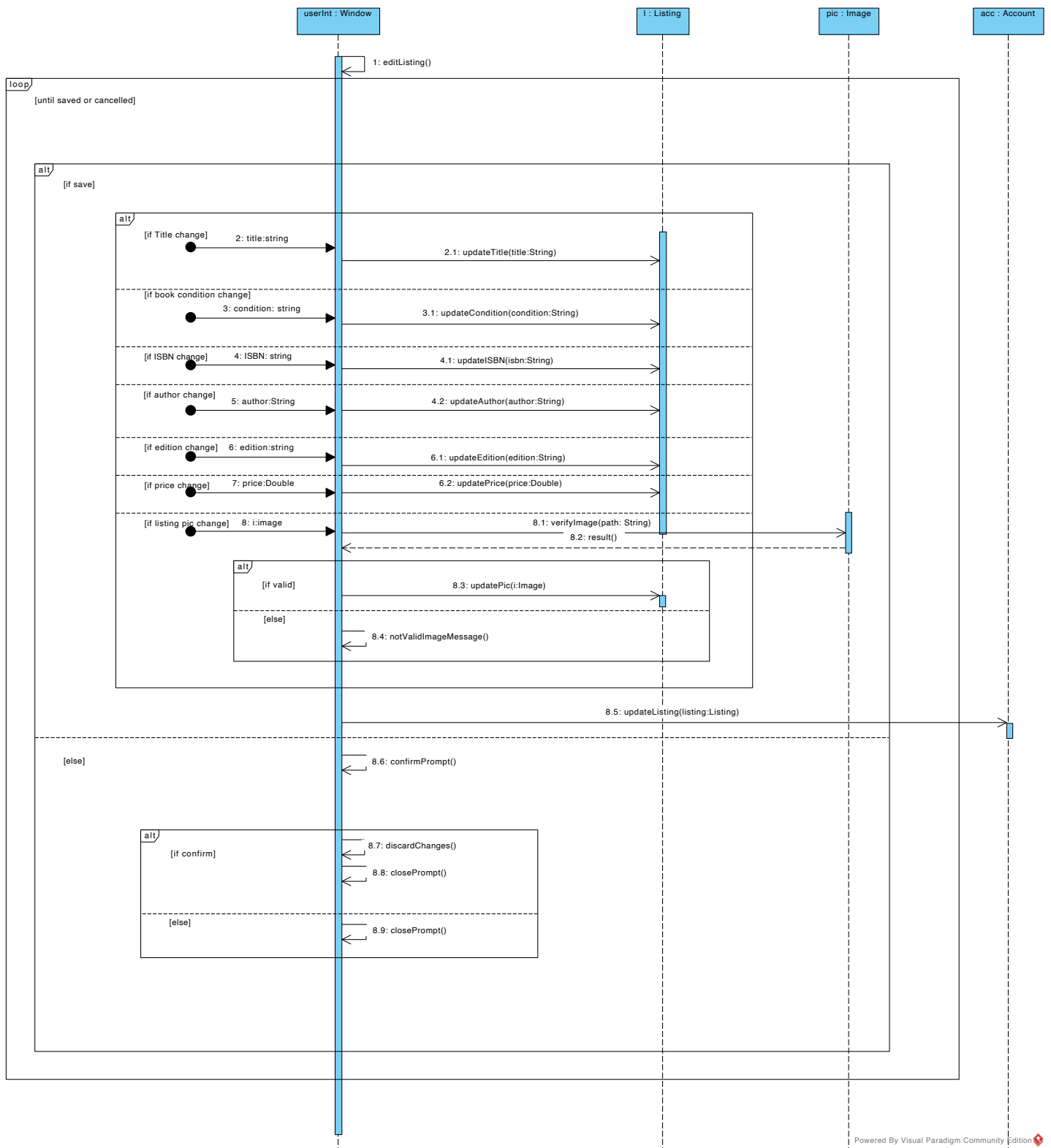












# Test Coverage Plan

Actor	Module	Case Type	Test Case
User	Login	Default	valid email, valid password
		Negative	invalid email, valid password
		Negative	valid email, invalid password
		Negative	empty fields
User	Account Creation	Default	email not taken, passwords match, password requirements met
		Negative	email taken, passwords match, password requirements met
		Negative	email not taken, passwords match, password requirements not met
		Negative	email not taken, passwords do not match, password requirements met
		Negative	empty fields
User	Create new Listing	Default	image file verified, all fields filled
		Negative	image file invalid, all fields filled
		Negative	image file valid, all fields not filled
User	Sign-out	Default	user is signed in
User	Edit Profile	Default	updated picture, picture valid, updated password, passwords match, password requirements met
		Alternate	updated password, passwords match, password requirements met
		Alternate	updated picture, picture valid
		Negative	updated picture, picture type invalid
		Negative	updated picture, picture type valid, picture size invalid
		Negative	updated password, passwords match password requirements not met
		Negative	updated password, passwords do not match, password requirements not met
		Negative	updated picture, picture type invalid, picture size invalid
		Negative	updated password, passwords do not match, password requirements not met
User	Edit listing	Default	image file verified, changed fields updated
		Alternate	no information changed, old settings kept
		Alternate	no image change, changed fields are updated
		Negative	image file invalid
		Negative	image file invalid, changed fields not updated until image is valid
User	Rate Seller	Default	rating selected 1-5
User	Purchase	Default	seller is emailed with contact information of buyer, and information of listing

# User Interface

B.B.B - Create Account

Email:

Password:

B.B.B - Edit Profile

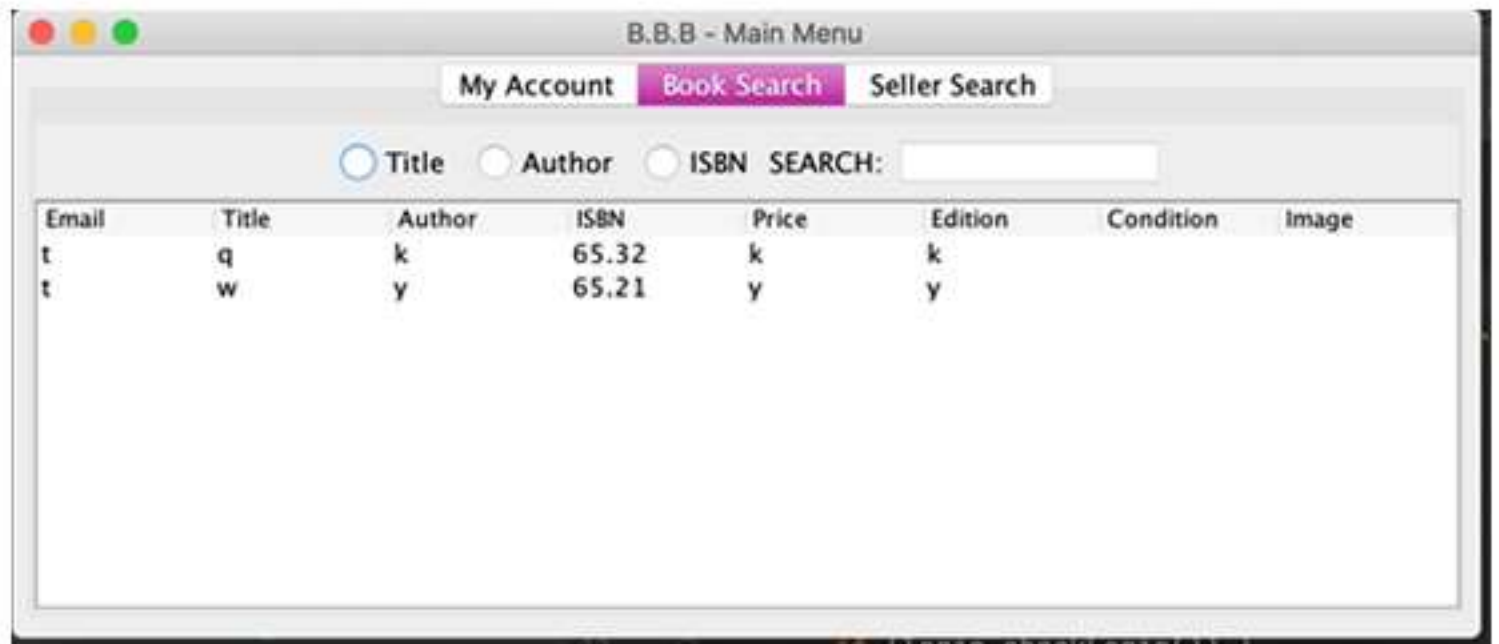
New Password:

Upload Image:

B.B.B - Login

Email:

Password:



B.B.B - New Listing

Title:

Author:

ISBN:

Price:

Edition:

Condition:

Upload Image:

Choose File (PNG or JPG only, < 50 mb)

Save Changes

Cancel

# Timecards Report



Time Sheet																
Member	Hours	9/11/20	9/13/20	9/15/20	9/16/20	9/17/20	9/18/20	9/21/20	10/6/20	10/13/20	10/16/20	10/20/20	10/22/20	10/23/20	10/26/20	10/27/20
Tyler Crumrine	25	1	2	2	1	1		3	1	2	1	2	4	1	2	2
Chris Torres	27	1	2	2	2	1		3	1	2	4	2	3		2	2
Tom Pechulis	25	1	1	2	2	1	1	3	1	2	1	2	1	2	3	2

# Gantt Diagram

# Baylor Blackmarket Bookstore

---

Oct 20, 2020

<http://>

Project manager

Tom Pechulis

Project dates

Sep 11, 2020 - Nov 25, 2020

Completion

65%

Tasks

31

Resources

3

## Tasks

2

Name	Begin date	End date
Iteration 1	9/11/20	9/21/20
Fully Dressed Use Cases	9/11/20	9/13/20
System Sequence Diagrams	9/11/20	9/13/20
Requirements	9/15/20	9/15/20
Gantt Diagram	9/15/20	9/18/20
Wire Frame Diagrams	9/15/20	9/18/20
Traceability Matrix	9/15/20	9/18/20
Domain Model	9/15/20	9/18/20
Operation Contracts	9/15/20	9/21/20
Presentation	9/18/20	9/21/20
Website Set-Up	9/18/20	9/21/20
Iteration 2	9/24/20	10/29/20
Sequence Diagrams	9/26/20	9/30/20
Class Diagram	9/30/20	10/3/20
GRASP Patterns	10/7/20	10/11/20
Prototype Implementation	9/24/20	10/15/20
Testing	10/8/20	10/19/20
Updated Analysis	10/21/20	10/23/20
Updated Website	10/23/20	10/27/20
Presentation	10/27/20	10/29/20
Iteration 3	10/31/20	11/24/20
Persistence + Business Layer	10/31/20	11/6/20
OCL	11/6/20	11/11/20
User Guide	11/11/20	11/15/20
Revised Analysis	11/18/20	11/22/20
Use Case Video	11/15/20	11/22/20
UI	10/31/20	11/12/20

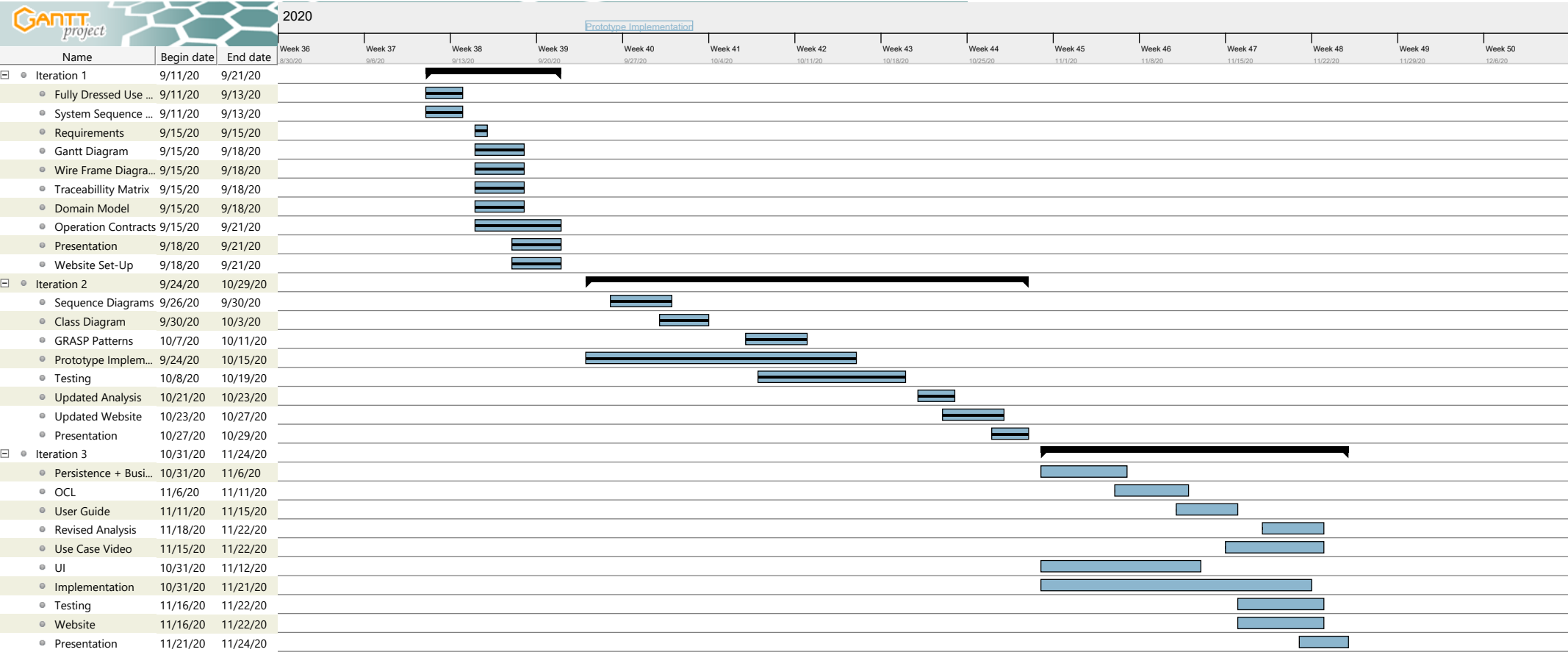
Tasks

Name	Begin date	End date
Implementation	10/31/20	11/21/20
Testing	11/16/20	11/22/20
Website	11/16/20	11/22/20
Presentation	11/21/20	11/24/20

Resources

Name	Default role
Tyler Crumrine	developer
Chris Torres	developer
Tom Pechulis	project manager

Gantt Chart







# Issue Tracking/Git/Point Redistribution

Github Link: <https://github.com/TomPechulis/BBB.github.io>

Issue Tacking: <https://trello.com/b/wGqLWLLN/csi-3471-bbb>

No Suggested Point Redistribution needed