Project Part II

Note: Some diagrams may be hard to read, so full-sized diagrams are also included in a folder.

Team: Tiffany Christensen, Chia-Lo Hsu, Joseph Marylander

Title: Closet Bot

Project Summary:
Closet Bot is an application that allows users to add and tag different articles of clothing into their "closet". Based on what is in a given user's closet and specified user conditions (for instance, "winter"), the app will generate a collection of outfits for you that you can then save if desired to an outfit closet.

Project Requirements:

No Business Requirements

User Requirements				
ID	**Requirement**	**Topic Area**	**User**	**Priority**
UR_01	User can sign up	Login	All	Critical
UR_02	User can log in	Login	All	Critical
UR_03	User can log out	Login	All	Critical
UR_04	User can add clothing to their "closet"	Closet Interactions	All	High
UR_05	User can edit existing clothing in their closet	Closet Interactions	All	Medium
UR_06	User can remove clothing from their "closet"	Closet Interactions	All	Medium
UR_07	User can view their own closet	Closet Interactions	All	High
UR_08	User can generate outfit based on season	Closet Interactions	All	High

UR_09	User can save generated outfit t outfit closet	Closet Interactions	All	Medium
UR_10	User can view saved outfit closet	Closet Interactions	All	Medium

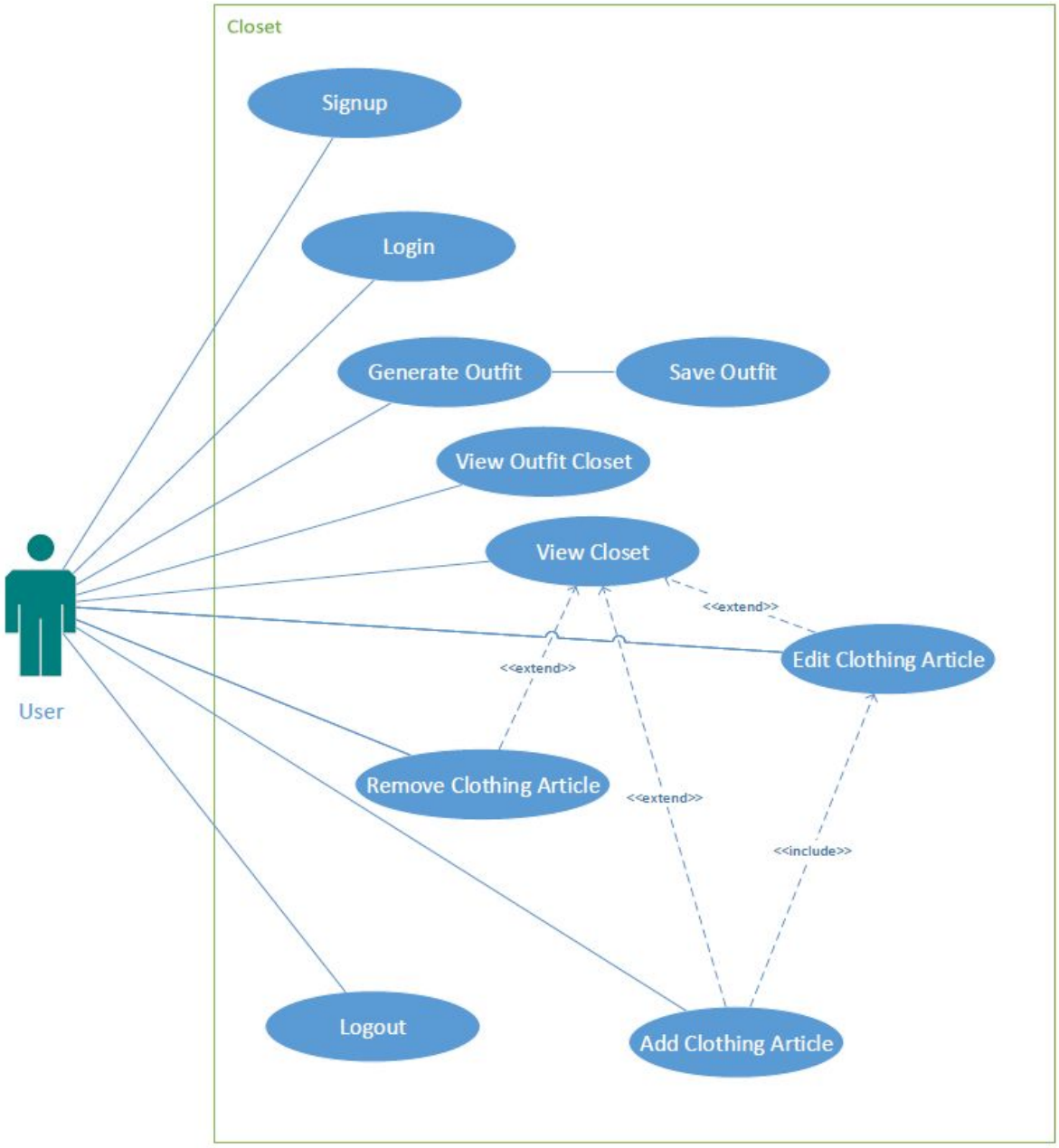
Functional Requirement			
ID	**Requirement**	**Topic Area**	**Priority**
FR_01	User closet information is stored in database	Database	Critical
FR_02	User login password hashes are stored in database	Database	Critical
FR_03	User general info is stored in database	Database	Medium
FR_04	Clothing tags are stored in database	Database	High
FR_05	When creating a new clothing article, user is presented with dropdown of clothing type based on select (ex. Top → tanktop)	Usability	High
FR_06	When creating a new clothing article, user is presented with dropdown of tags in various categories for further categorization	Usability	High
FR_07	When user add clothing articles or outfits to a given closet, database is updated with changes	Database	High

Non-Functional Requirements			
ID	**Requirement**	**Topic Area**	**Priority**
NF_01	When a user changes their closet, update time should be quicker than < 1s	Performance	High
NF_02	A user should be able to access their closet remotely	Usability	Medium

Use Cases

Actor: User

Use Case Overview



Sub-Diagrams

None needed.

Use Case Documents

Use Case ID	**UC_01**		
Use Case Name	Add Clothing Article		
Description	User can add an article of clothing to their closet		
Actors	User		
Preconditions	User is logged in		
Post-Conditions	The closet has one more article of clothing along with its corresponding tags		
Frequency of Use	Frequently throughout the day		
Flow of Events		**Actor Action**	**System Response**
	1	Click button to create a clothing article	Display clothing dialog which prompts for clothing tags
	2	Select clothing type	Display clothing subtype dropdown
	3	Select subtype	
	4	Select color	
	5	Select season	
	6	Select pattern	
	7	Click save	Closes clothing dialog. Saves clothing information to database.
Variations	N/A		
Exceptions	N/A		
Developer Notes	Type & subtype example -- if user clicks on clothing type “shirt”, a dialog of subtypes (for instance, “tank, longsleeve, shortsleeve) appears based on previously selected type		

Use Case ID	**UC_02**		
Use Case Name	Remove Clothing Article		
Description	User can remove an article of clothing from their closet		
Actors	User		
Preconditions	User is logged in and there is at least one article in the closet		
Post-Conditions	The closet has one less article of clothing and all corresponding tags have been removed		
Frequency of Use	Frequently throughout the day		
Flow of Events		**Actor Action**	**System Response**
	1	Find clothing article to be removed	
	2	Click remove button next to the clothing article	Removes clothing article from database
Variations	N/A		
Exceptions	N/A		
Developer Notes	N/A		

Use Case ID	**UC_03**
Use Case Name	Edit Clothing Article
Description	User can edit attribute tags on existing clothing article
Actors	User
Preconditions	User is logged in and there exists at least one article of clothing in the closet
Post-Conditions	The clothing article in the closet is updated with the new tags
Frequency of Use	Frequently throughout the day

Flow of Events		**Actor Action**	**System Response**
	1	Find clothing article to be edited	
	2	Click edit button next to clothing article	Display clothing dialog
	3	Select clothing tag to be changed	Display selected tags in clothing dialog
	4	Click save	Update closet database with new tags
Variations	N/A		
Exceptions	N/A		
Developer Notes	The clothing dialog is the same one that appears when adding a new clothing article, except the fields are prepopulated with the existing attributes.		

Use Case ID	**UC_04**		
Use Case Name	Generate Outfits		
Description	User can generate a collection of outfits to wear		
Actors	User		
Preconditions	User is logged in		
Post-Conditions	Generated outfits are displayed to the user		
Frequency of Use	Frequently throughout the day		
Flow of Events		**Actor Action**	**System Response**
	1	Click Generate Outfits	Prompt user for the season
	2	Select a Season	Display generated outfits dialog
Variations	User selects next outfit (if they do not like it) or save the outfit to their outfit closet (if they want to keep it)		
Exceptions	If the closet is empty, the outfit generator will generate a default		

	“birthday suit” for the user.
Developer Notes	Outfit iterator will iterate through the collection of outfits generated for the user

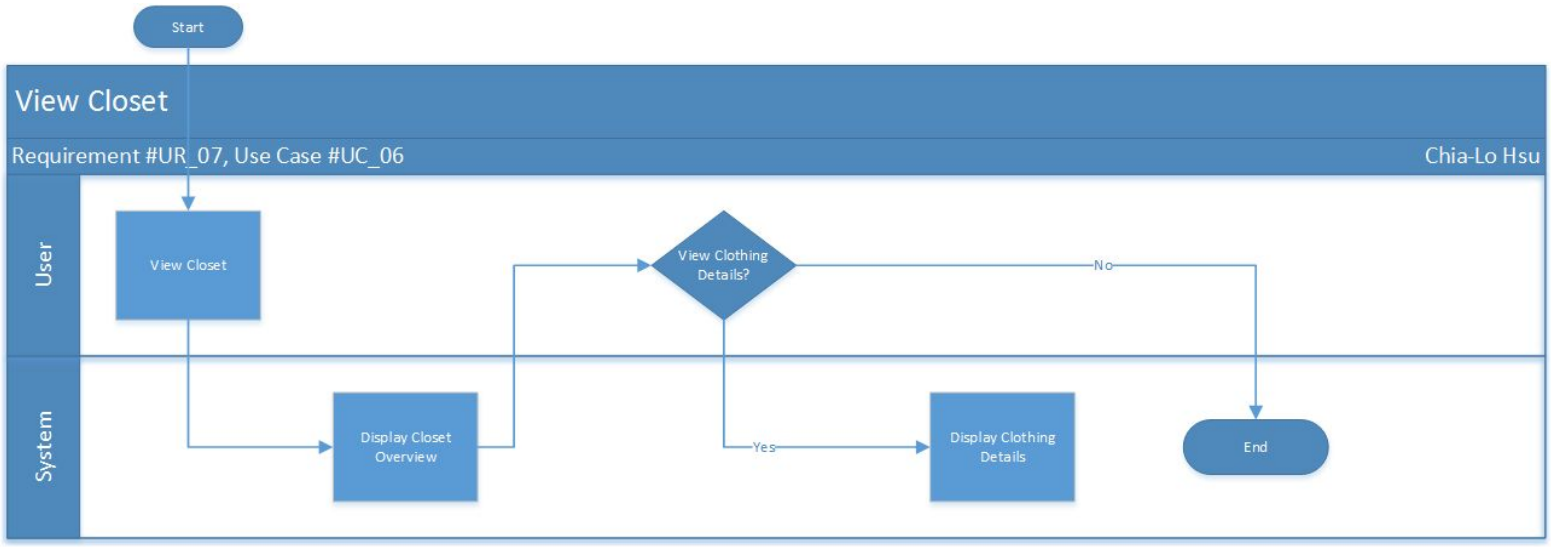
Use Case ID	**UC_05**		
Use Case Name	Save Outfit		
Description	User can save a generated outfit to outfit closet		
Actors	User		
Preconditions	User is logged in		
Post-Conditions	Generated outfit is saved to outfit closet		
Frequency of Use	Frequently throughout the day		
Flow of Events		**Actor Action**	**System Response**
	1	Click Generate Outfits	Display generated outfits dialog
	2	Select outfit from iterator of randomly generated outfits	
	3	Click Save outfit	Outfit is added to outfit closet
Variations	N/A		
Exceptions	N/A		
Developer Notes	N/A		

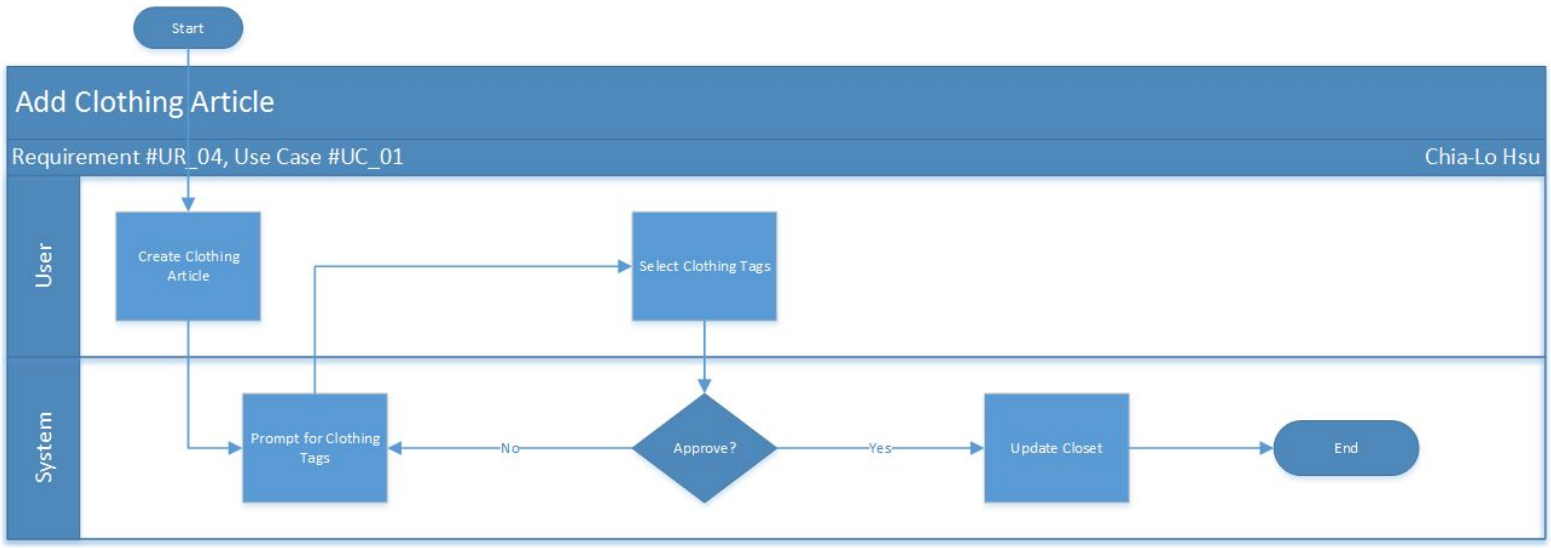
Use Case ID	**UC_06**
Use Case Name	View Closet
Description	User can view all contents of his/her closet
Actors	User

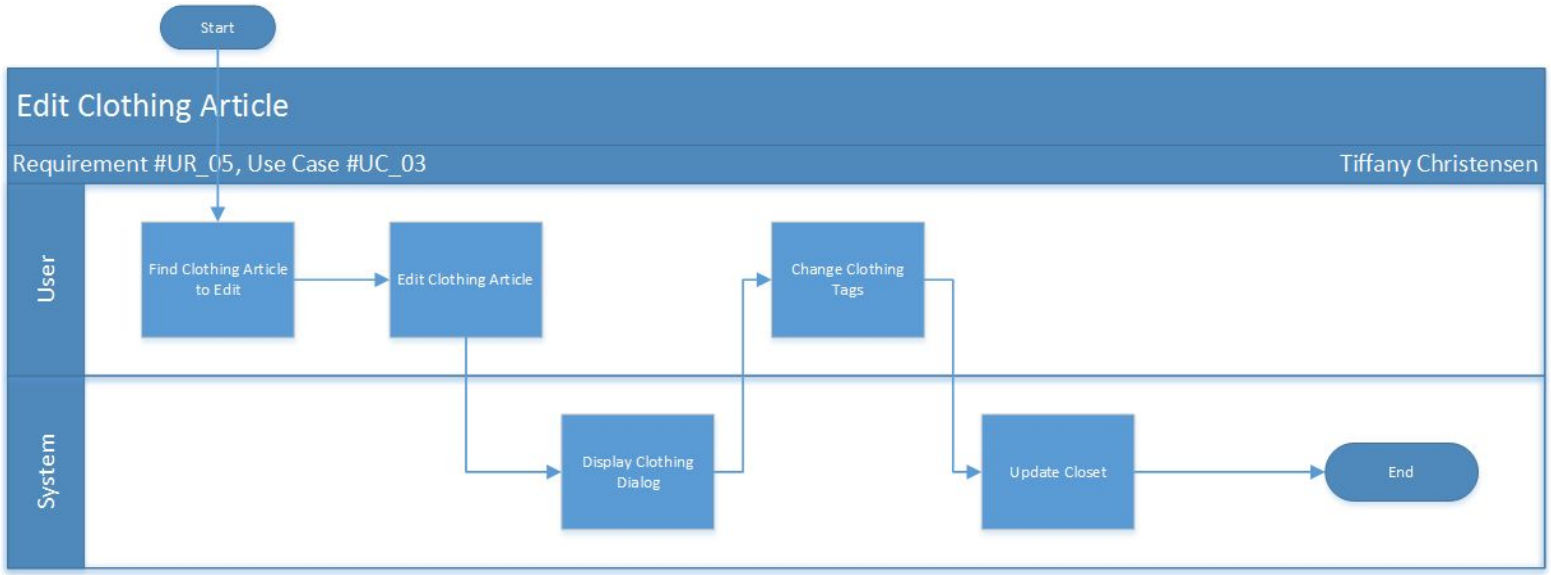
Preconditions	User is logged in		
Post-Conditions	Closet contents are displayed		
Frequency of Use	Frequently throughout the day		
Flow of Events		**Actor Action**	**System Response**
	1	User logs in	Redirects user to their closet Displays articles of clothing in table
Variations	N/A		
Exceptions	N/A		
Developer Notes	N/A		

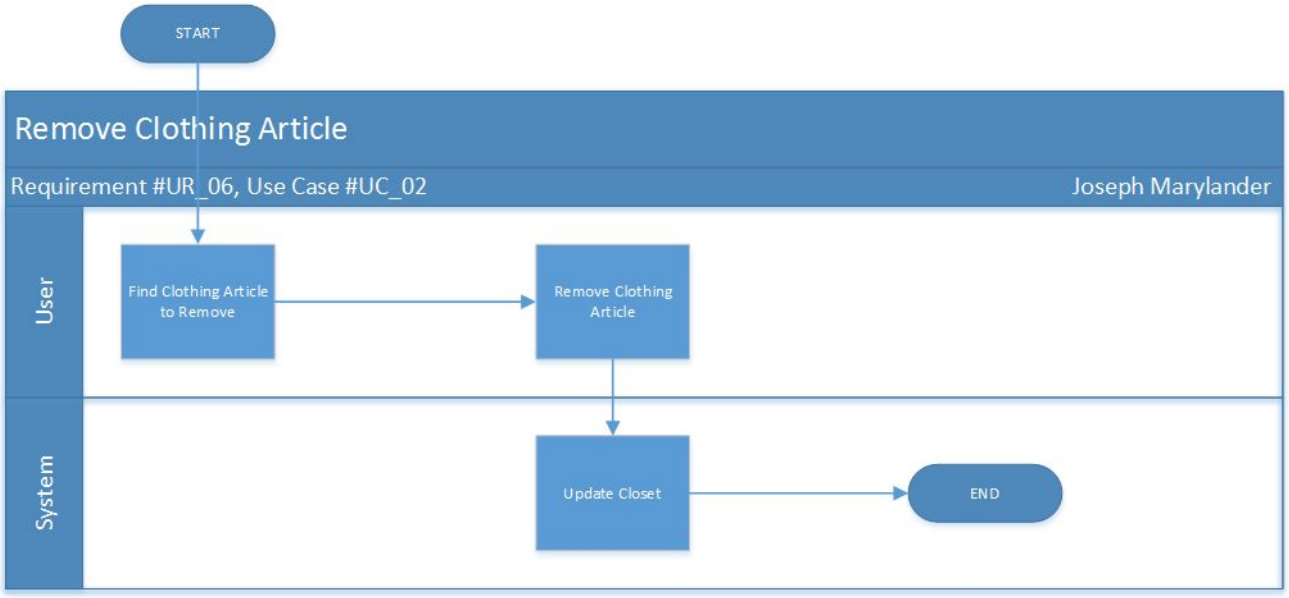
Use Case ID	**UC_07**		
Use Case Name	View Outfit Closet		
Description	User can view outfit closet of their saved outfits		
Actors	User		
Preconditions	User is logged in		
Post-Conditions	Saved outfits are displayed		
Frequency of Use	Frequently throughout the day		
Flow of Events		**Actor Action**	**System Response**
	1	Click on Outfit Closet	Redirects to user's outfit closet and displays outfits in table
Variations	N/A		
Exceptions	N/A		
Developer Notes	N/A		

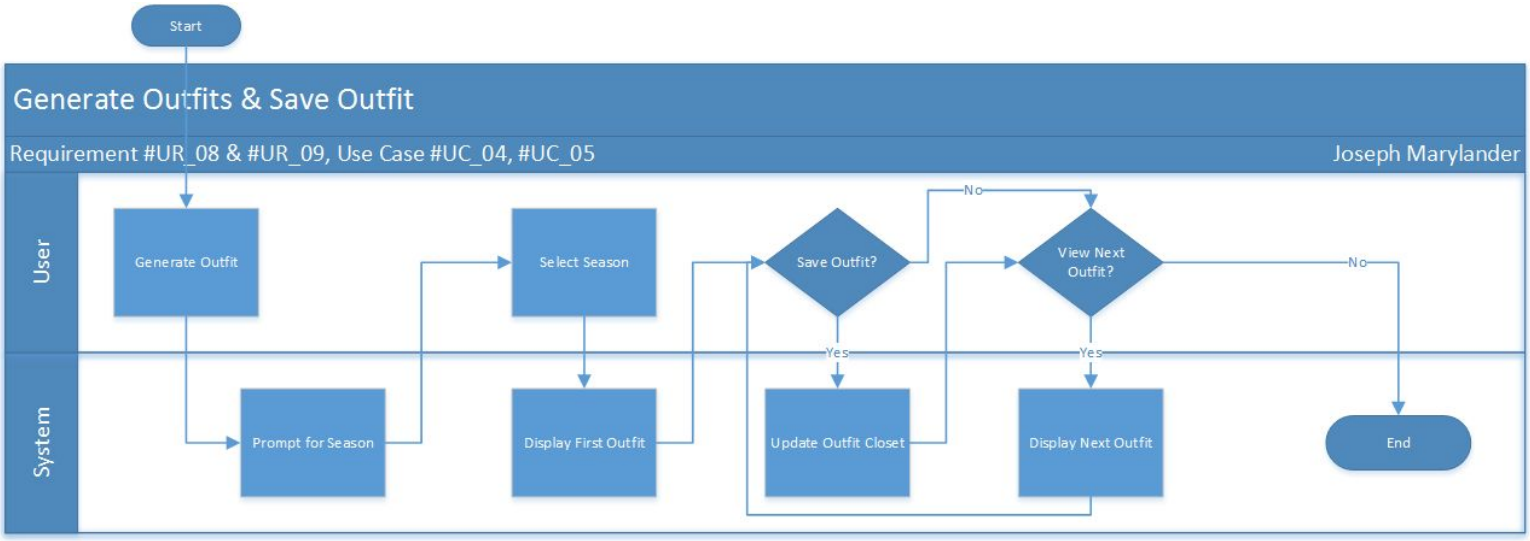
Activity Diagrams

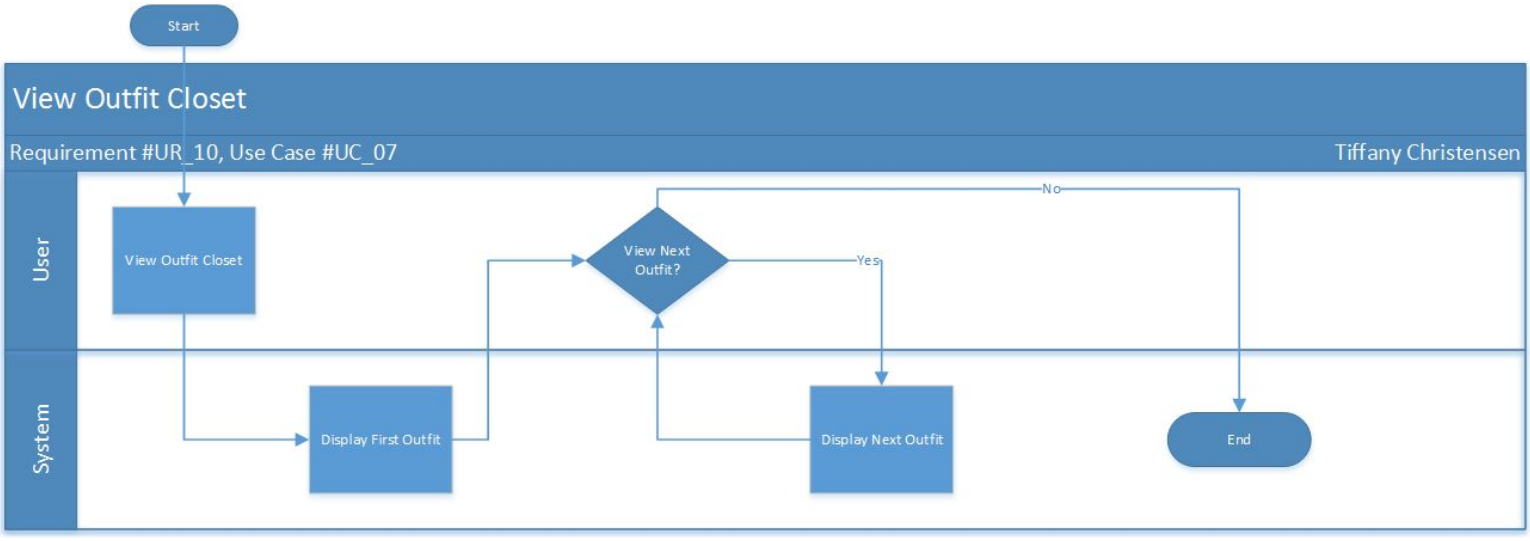







START
Remove Clothing Article
Requirement #UR_06, Use Case #UC_02
Joseph Marylander
User
Find Clothing Article to Remove
Remove Clothing Article
System
Update Closet
END

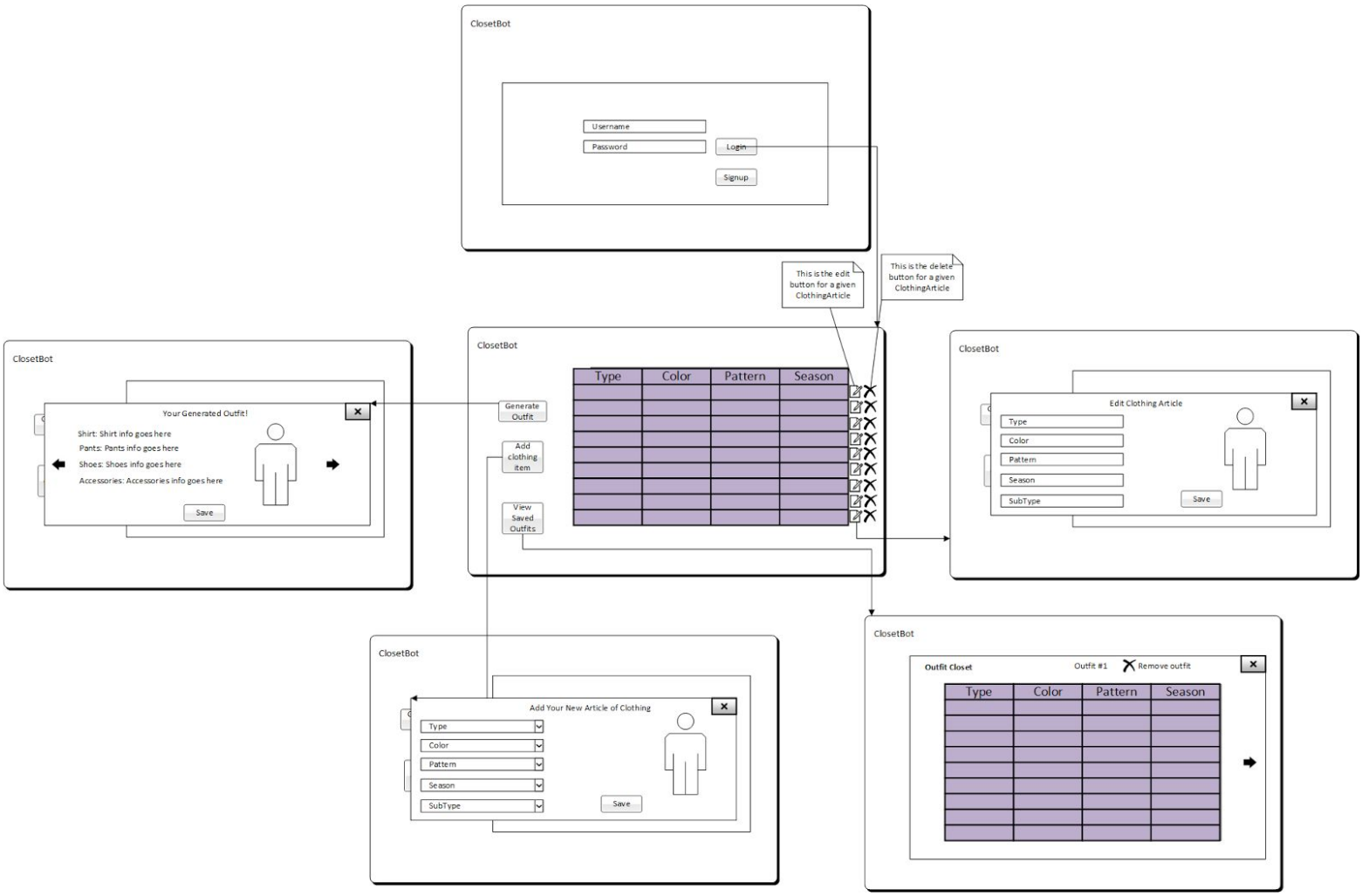

Start
Generate Outfits & Save Outfit
Requirement #UR_08 & #UR_09, Use Case #UC_04, #UC_05
Joseph Marylander
User
Generate Outfit
Select Season
Save Outfit?
No
View Next Outfit?
No
Yes
Yes
System
Prompt for Season
Display First Outfit
Update Outfit Closet
Display Next Outfit
End


Start
View Outfit Closet
Requirement #UR_10, Use Case #UC_07
Tiffany Christensen
User
View Outfit Closet
View Next Outfit?
No
Yes
System
Display First Outfit
Display Next Outfit
End

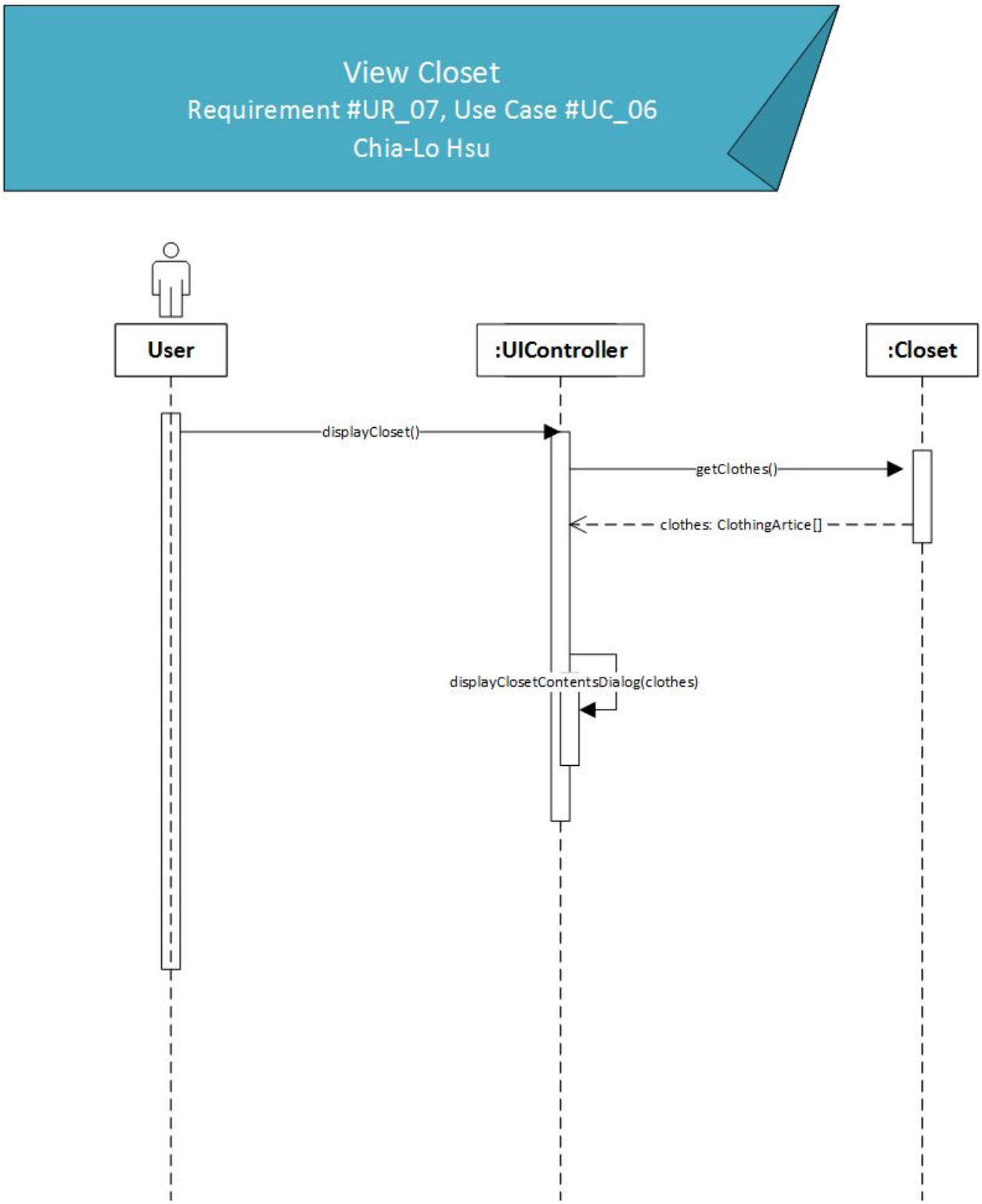
Data Storage

We plan on using a MySql database to store individual user's data. Each tuple in the database will have some basic info about the user as well as their username and password hash. Additionally, we will use blob storage to store a user's closet and outfit closet.

UI Mockups



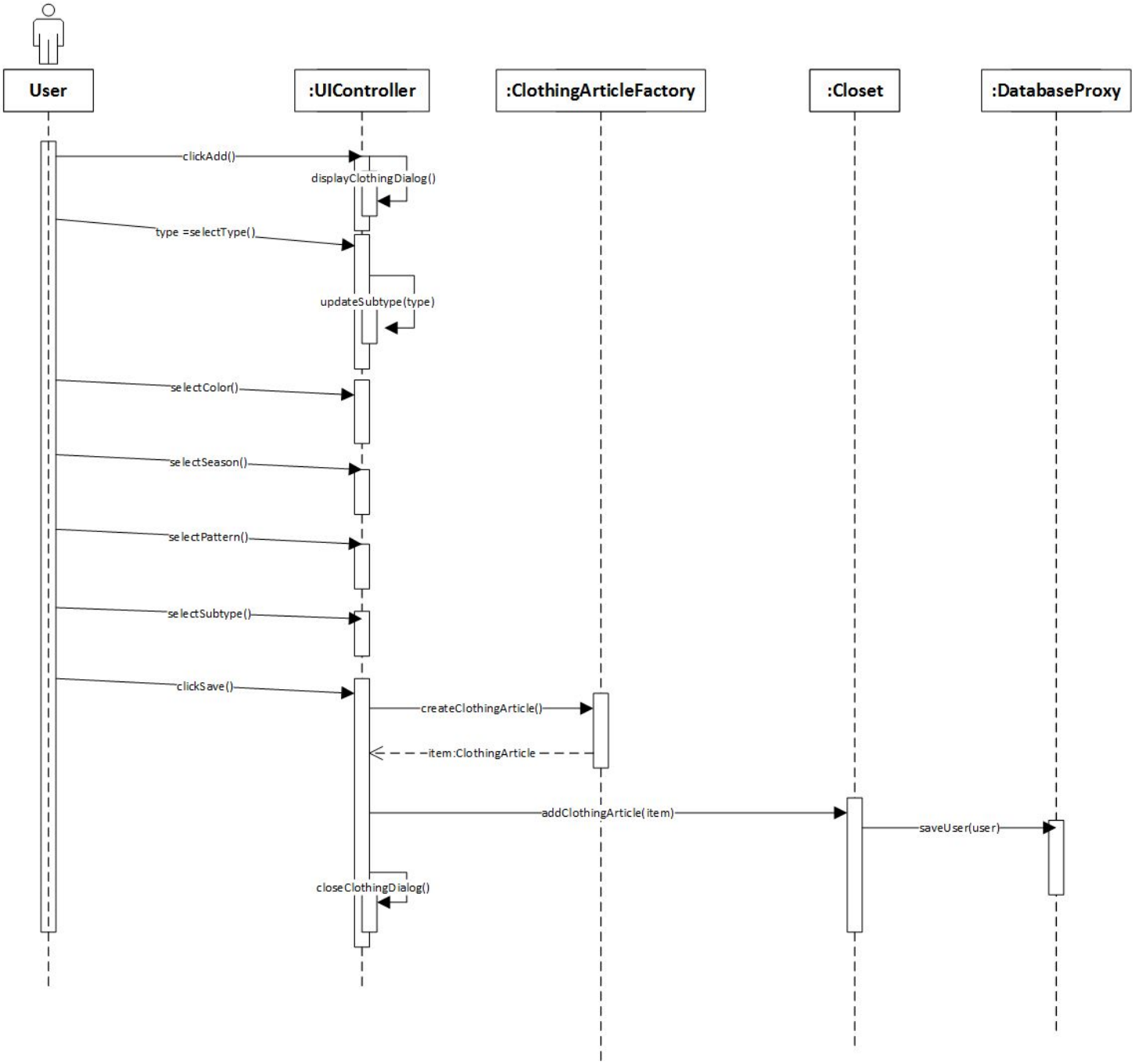
User Interactions (Sequence Diagram)



Add Clothing Article

Requirement #UR_04, Use Case #UC_01

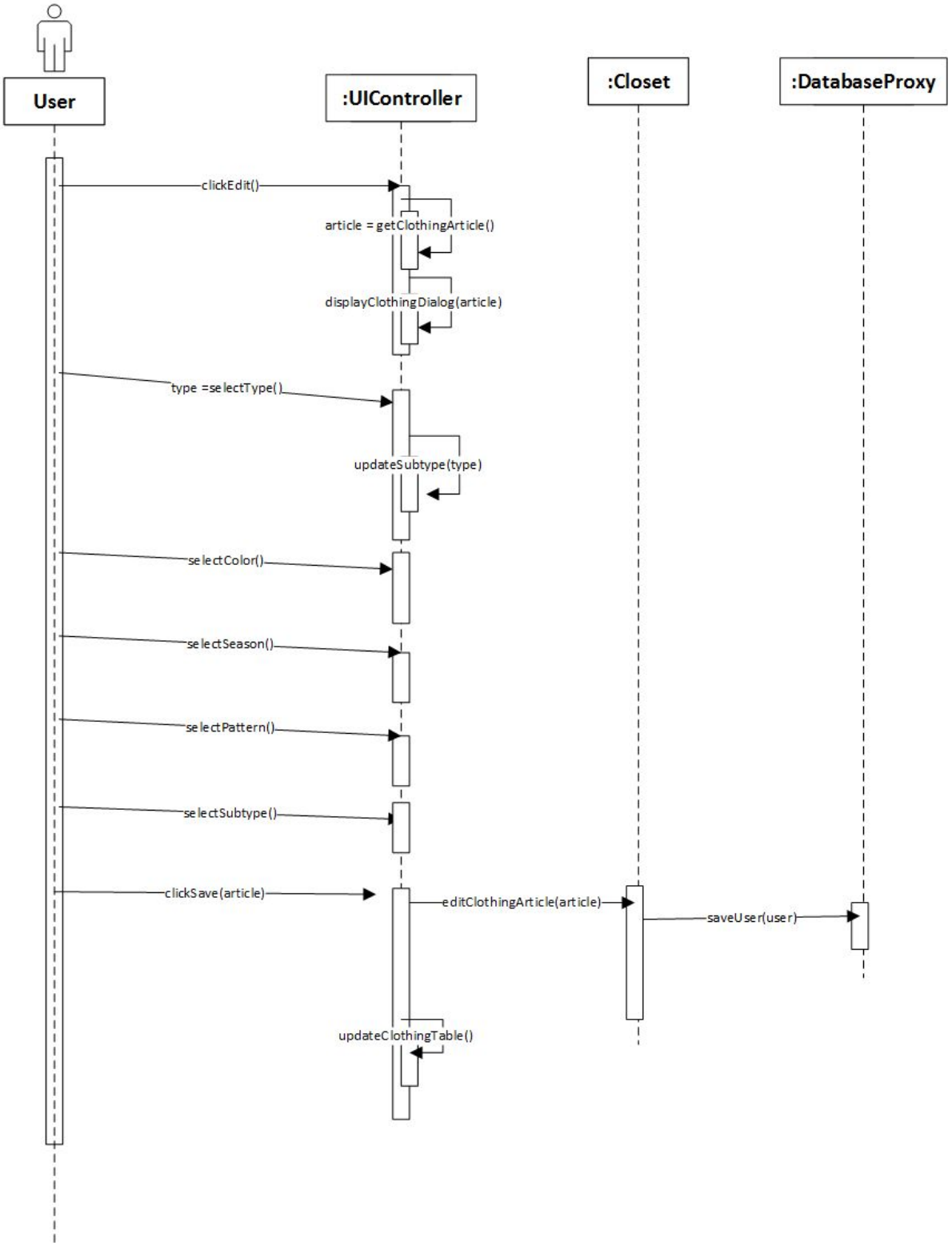
Chia-Lo Hsu



Edit Clothing Article

Requirement #UR_05, Use Case #UC_03

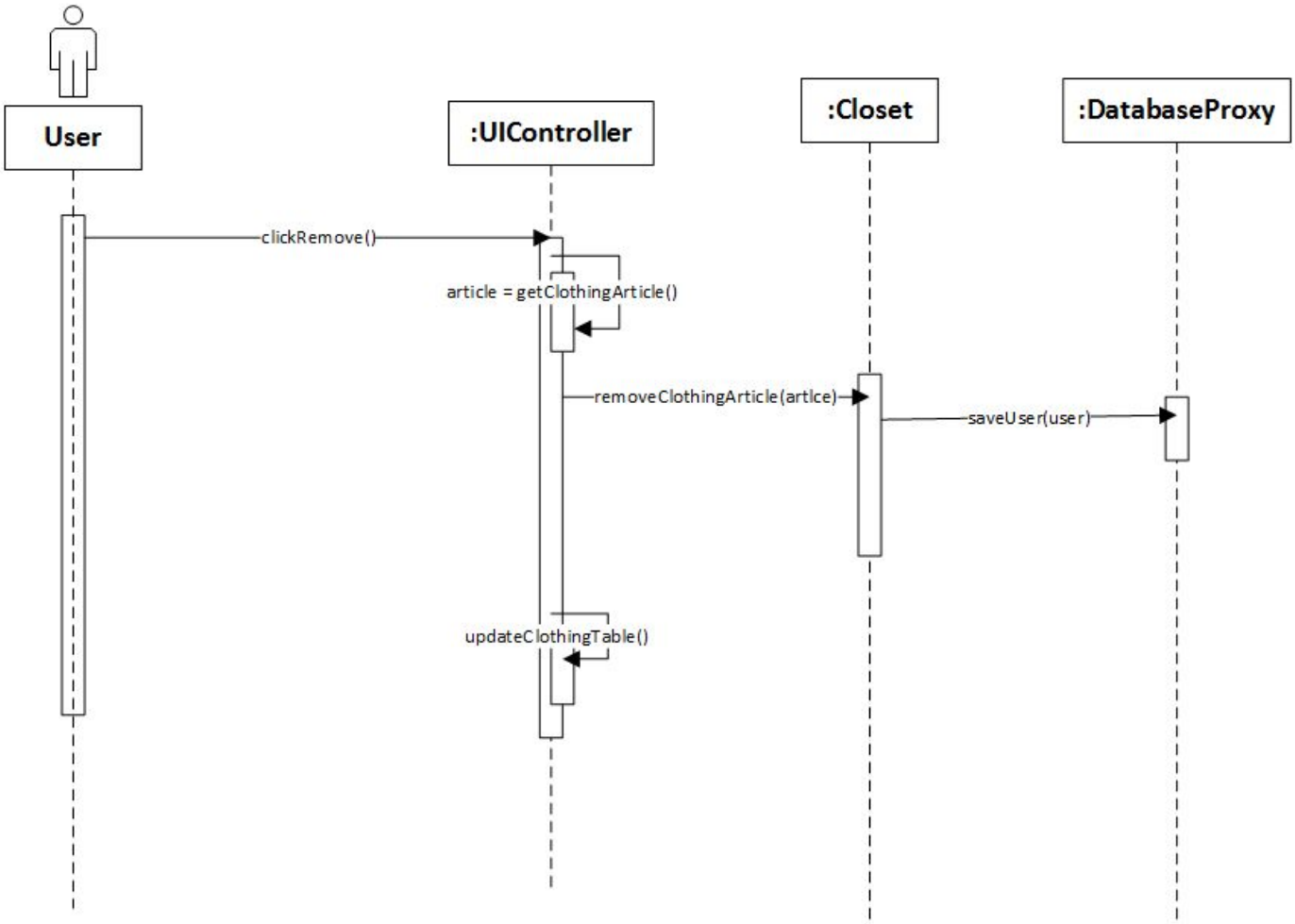
Tiffany Christensen



Remove Clothing Article

Requirement #UR_06, Use Case #UC_02

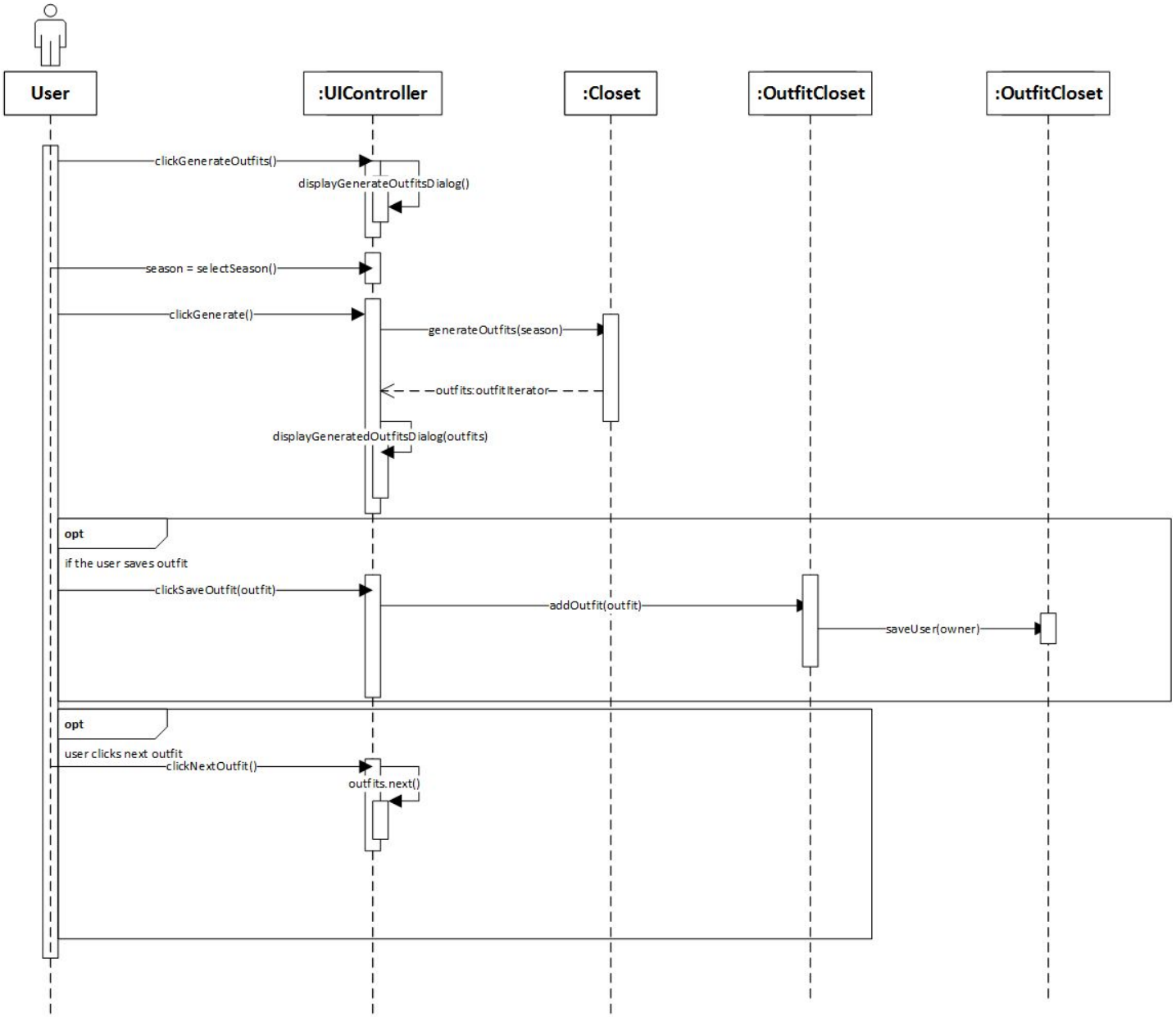
Joseph Marylander



Generate Outfits & Save Outfit

Requirement #UR_08 & #UR_09, Use Case #UC_04 & #UC_05

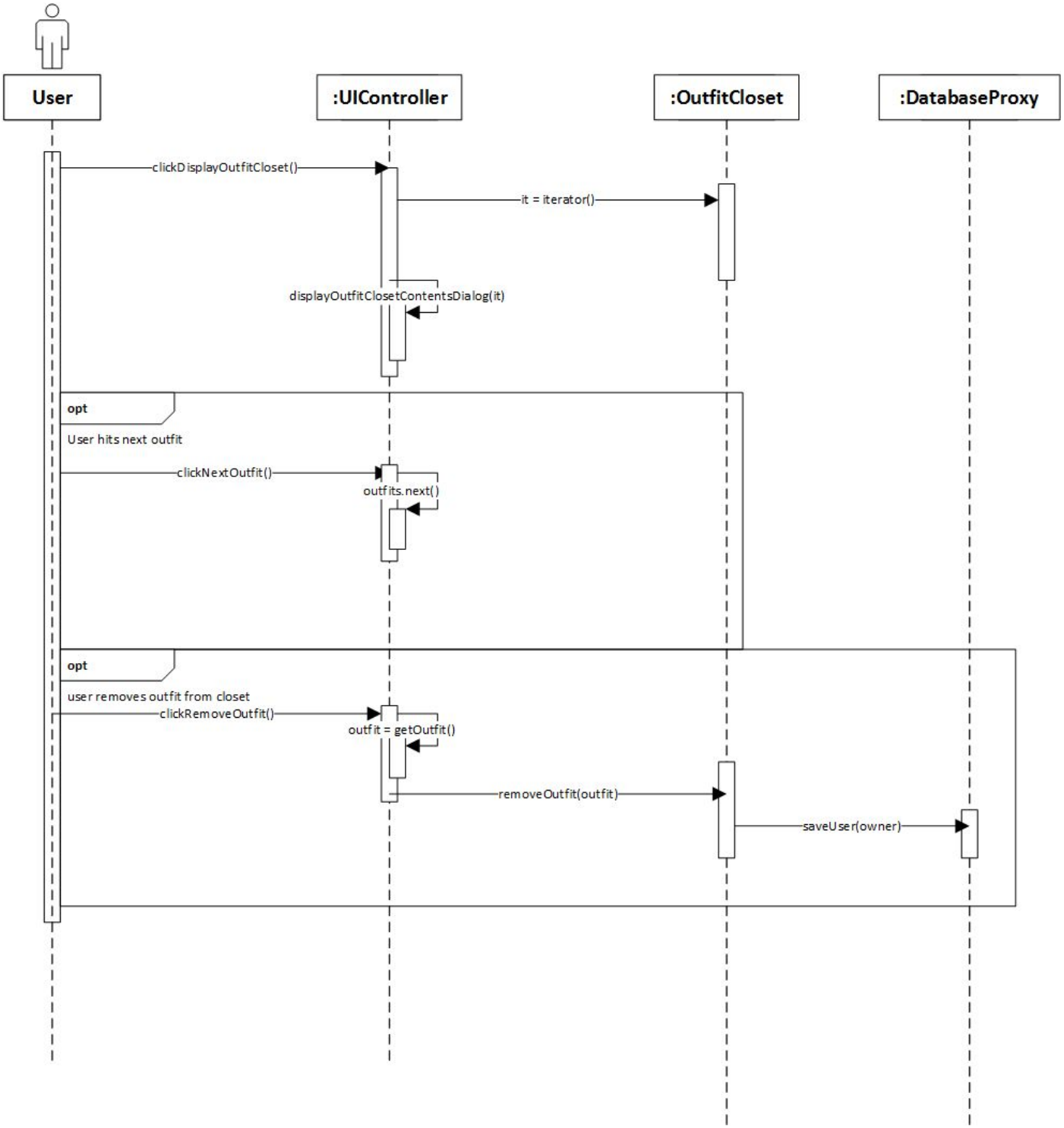
Joseph Marylander



View Outfit Closet

Requirement #UR_10, Use Case #UC_07

Tiffany Christensen



Class Diagrams