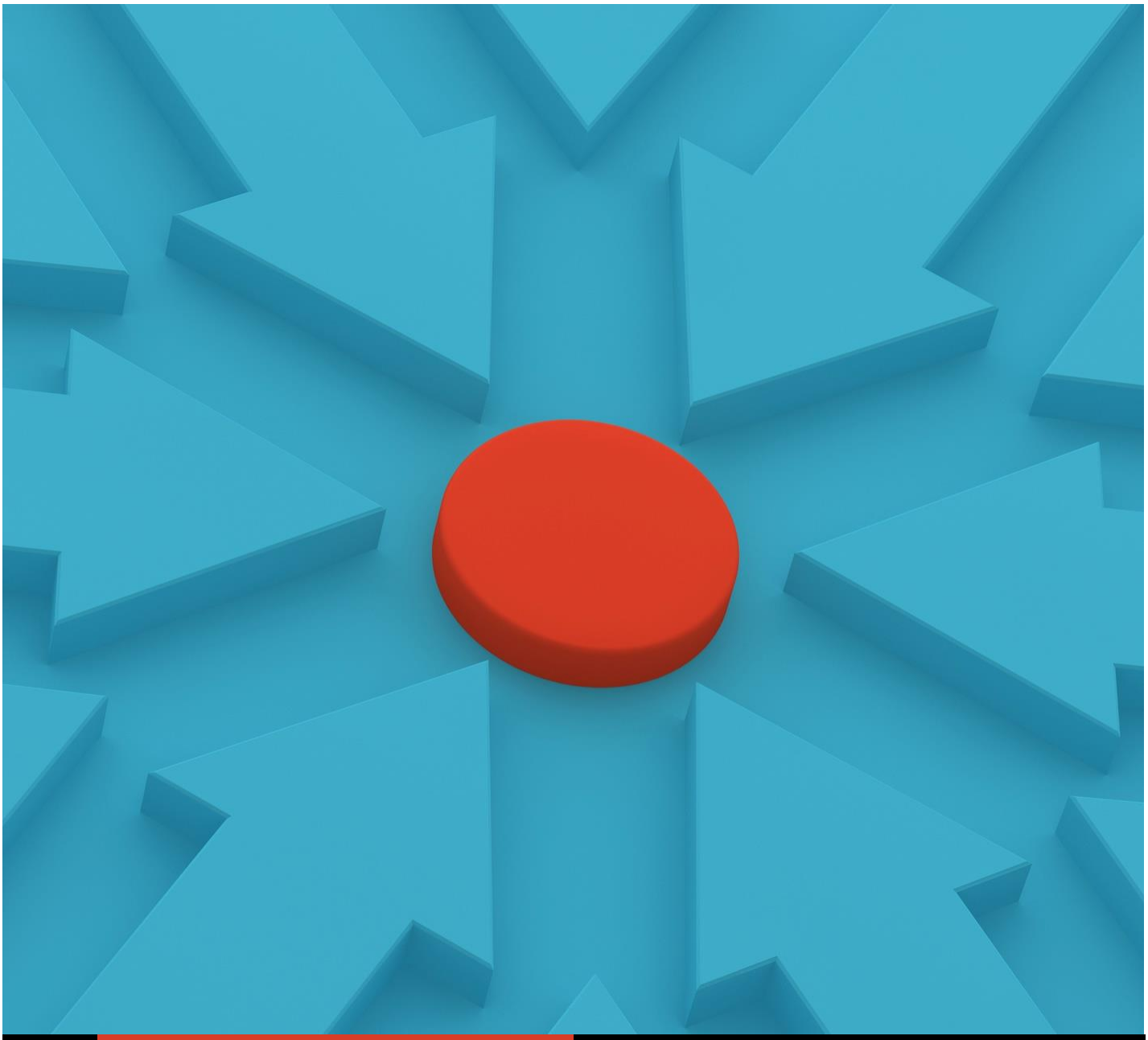


Updated Class Model Diagram, Activity Diagram, Focused Sequence Diagram, and Pseudocode for the Best Purchase Application.

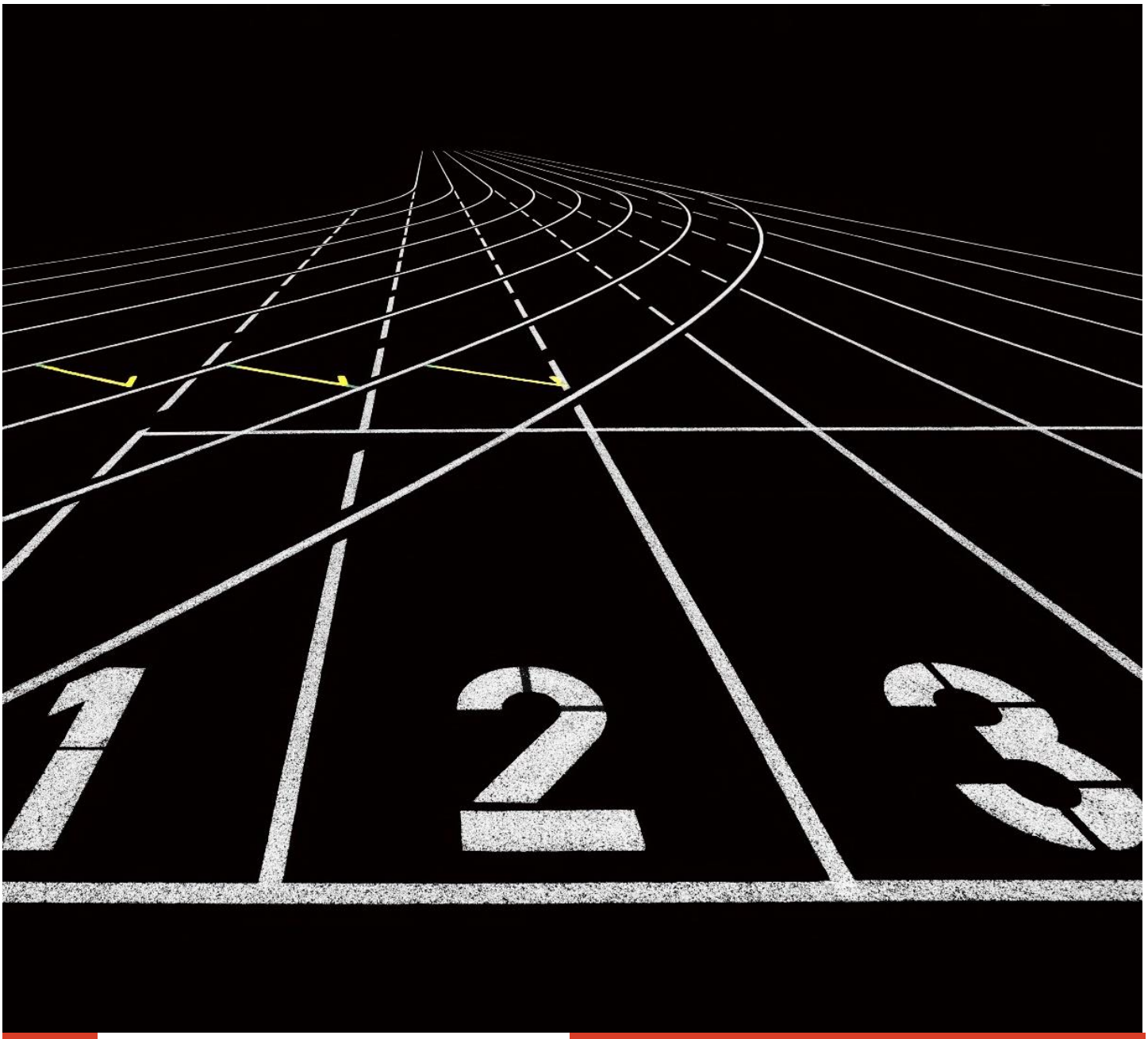
Abstract: What does the updated Class Model diagram look like? What does the Activity diagram look like for a selected method from the Class Model diagram? What does the Focused Sequence diagram look like for an activity within the Activity diagram? What is the pseudocode for the selected method?

Harrison Huston

Part 4, February 28, 2023

TABLE OF CONTENTS

FIGURES

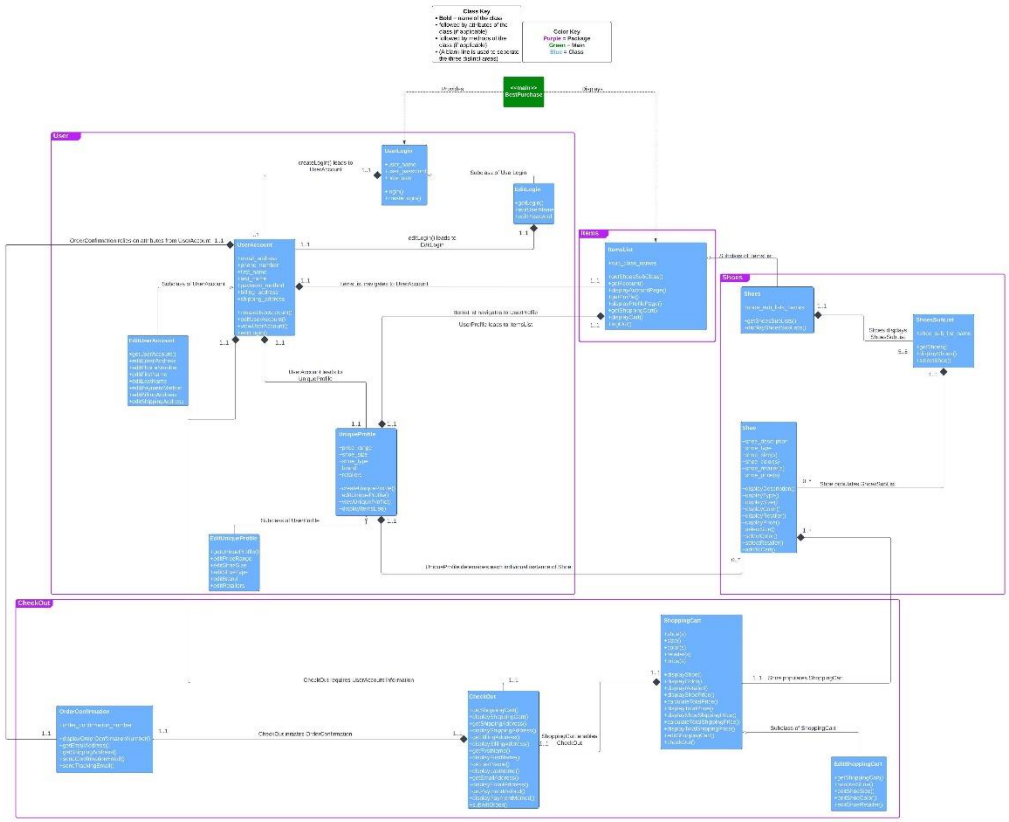


INTRODUCTION

Part 4 focuses on the BestPurchase application and specifying design details. An updated Class Model diagram is presented. A method from the Class Model diagram is selected and an Activity diagram pertaining to this method is displayed. A Focused Sequence diagram is created based on an activity from the Activity diagram. Lastly, pseudocode is created for the selected method.

4.1 UPDATED CLASS MODEL

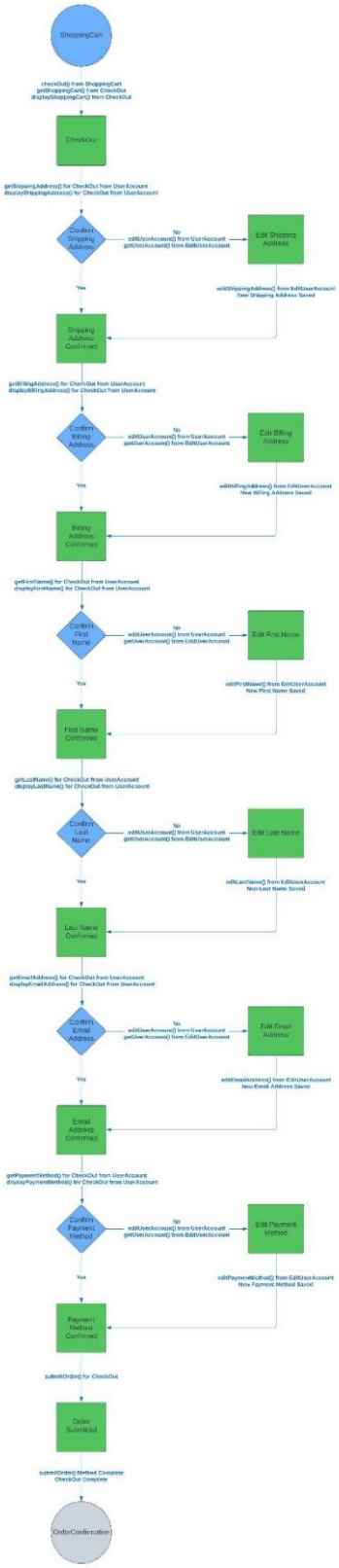
The below Class model has been revised in the following manner. The method getAccount() within the EditUserAccount class has been changed to getUserAccount() to better conform with the rest of the diagram. The method getCart() within the EditShoppingCart class has been changed to getShoppingCart() as well. The method editShoppingCart within the ShoppingCart class, was also changed to editShoppingCart() to conform with the correct method format. Outside of these minor edits, the Class Model remains the same as the previous part's version, which can be viewed in *Appendix 2*.



("Module 4," 2023; Williams, 2023)

4.2 ACTIVITY DIAGRAM

The selected method within the above Class Model diagram is the submitOrder() method, which resides within the CheckOut class, within the CheckOut package. The submitOrder() method allows for the submission of an order through the BestPurchase application and leads to the start of the creation of an order confirmation. The method is intricate in how it depends on a variety of classes and subclasses, as well as certain methods within those classes. The below Activity Diagram illustrates these interactions and activity's flow. It is important to notice in the visualization of the diagram below, that the submitOrder() method relies on the confirmation of the shipping address, billing address, first name, last name, email address, and payment method of the customer to successfully run the method.



(“Module 1,” 2023)

4.2.1 Focused Sequence Diagram

The below sequence diagram illustrates the activity of an Account Holder completing the checkout process. This would precede the population of the ShoppingCart class and the running of the checkOut() method.

(“Module 4,” 2023; Williams, 2023)

4.3 PSEUDOCODE

The below pseudocode syntactically, is most similar to Java. It does not include code that will compile, but does provide a high level pseudocode implementation of the method and methods associated with the submitOrder() method.

```
/* If checkOut() method is true, i.e. user enters CheckOut, run getShoppingCart() and
displayShoppingCart() for CheckOut class */
IF (checkOut() == True) {
        getShoppingCart();
        displayShoppingCart();
}
/* Else editShoppingCart() within the ShoppingCart class and getShoppingCart() within
the EditShoppingCart class */
ELSE {
        editShoppingCart();
        getShoppingCart();
}

// While CheckOut() is True, i.e. user entered CheckOut
WHILE (CheckOut() == True) {

        /* Run methods to get and display shipping address, billing address, first name,
                last name, email address, and payment method */
        getShippingAddress();
        displayShippingAddress();
        getBillingAddress();
        displayBillingAddress();
        getFirstName();
```

```
displayFirstName();
getLastName();
displayLastName();
getEmailAddress();
displayEmailAddress();
getPaymentMethod();
displayPaymentMethod();

/* While shippingAddress, billingAddress, firstName, lastName, emailAddress, or
        paymentMethod are False */
WHILE (shippingAddress == False || billingAddress ==False || firstName ==
        False || lastName == False || emailAddress == False || paymentMethod ==
        False) {

        // If shippingAddress is confirmed, set shippingAddress to True, else False
        IF (shippingAddress == confirmed){
                shippingAddress == True;
        }
        ELSE {
                shippingAddress == False;
        }

        // If billingAddress is confirmed, set shippingAddress to True, else False
        IF (billingAddress == confirmed){
                billingAddress == True;
        }
        ELSE {
```

```
        billingAddress == False;
}

// If firstName is confirmed, set shippingAddress to True, else False
IF (firstName == confirmed){
        firstName == True;
}
ELSE {
        firstName == False;
}

// If lastName is confirmed, set shippingAddress to True, else False
IF (lastName == confirmed){
        lastName == True;
}
ELSE {
        lastName == False;
}

// If emailAddress is confirmed, set shippingAddress to True, else False
IF (emailAddress == confirmed){
        emailAddress == True;
}
ELSE {
        emailAddress == False;
}
```

```
            // If paymentMethod is confirmed, set shippingAddress to True, else False
            IF (paymentMethod == confirmed){
                paymentMethod == True;
            }
            ELSE {
                paymentMethod == False;
            }
        }

        /* Once nested while loop breaks, i.e. all areas have been confirmed, run
        submitOrder() method, set CheckOut() method to false, breaking parent while
        loop and ending CheckOut process */
        submitOrder();
        CheckOut() == False;
}
```

CONCLUSION

The updated Class Model diagram displays a revised version of the Class Model diagram with the finalized packages, classes, attributes, and methods. The selected method, submitOrder() method, from the Class Model diagram is illustrated through the Activity diagram. The activity of an Account Holder completing the checkout process is shown via the Focused Sequence diagram. Lastly, the pseudocode portrays an implementation of the submitOrder() method.

SUMMARY

In summary, the updated Class Model diagram is provided and a method from the diagram is selected. This method is visualized via the Activity diagram. The Focused Sequence diagram is displayed, portraying an activity from the Activity diagram. Lastly, pseudocode is provided for the implementation of the selected method from the Class Model diagram.

APPENDIX

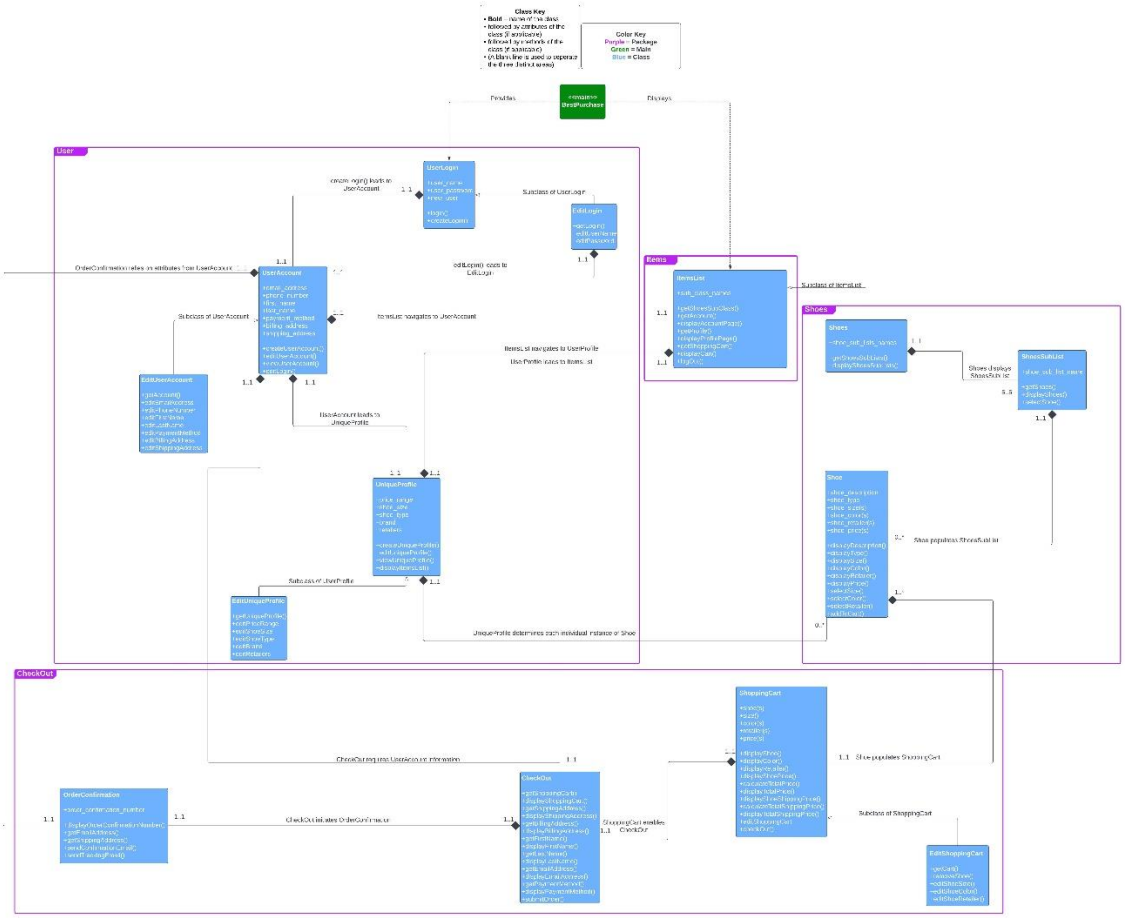
Appendix 1:

Mission Statement from Part 1

BestPurchase is a desktop and mobile hybrid shoe recommendation, sourcing, and purchasing application. It allows account holders to receive tailored shoe recommendations based off their unique customer requirements. Through the application, users will stay abreast of shoes available, both new shoe releases and prior releases that fit the users' criteria. BestPurchase allows for the purchase and order tracking of shoes directly through its application.

(Williams, 2023)

Appendix 2:



("Module 4," 2023; Williams, 2023)

REFERENCES

[1] Module 1: Introduction and Process. (2023). In D. Williams (Ed.), *Introduction to Systems Analysis Methodology – Part 2* (pp. 9). Boston University Metropolitan College.

[2] Module 4: Object-Oriented Design and UML Modeling. (2023). In D. Williams (Ed.), *Review of Class Diagram Components* (pp. 16). Boston University Metropolitan College.

[3] Module 4: Object-Oriented Design and UML Modeling. (2023). In D. Williams (Ed.), *Review of Sequence Diagram Components* (pp. 19). Boston University Metropolitan College.

[4] Williams, D. (2023, January 15). *Supplementary Live Session Week 3* [PowerPoint slides]. Boston University Metropolitan College.

[5] Williams, D. (2023, January 22). *Supplementary Live Session Week 4* [PowerPoint slides]. Boston University Metropolitan College.