**Project Development**

**Documents for**

**OneStop Menu**

**Application**

**Contents**

* 1. **Scope……………………………………………………......…….……3**
  2. **Identification…………………………………………………….….…3**
  3. **System overview………......………………………………………....3**
  4. **Document overview…………………………...…………………......3**

**2.0 References……………………………………………………………..4**

**3.0 Requirements…………………………………………………...……..5**

**4.0 Design………………………………………………………………..…6**

**5.0 Test Plan………………………………………………………….…….7**

**Appendix 8**

1. **Scope**

The project will start once this introduction is submitted.

The menu application will display a simple menu to customers of OneStop Restaurant. This will automate a large portion of services offered by the restaurant and reduce wait time.

The project will be done by myself. The requirements will be first laid out and a design implemented within the first week. Final design and testing will be done within the second week. The application released on November 17, 2020.

* 1. **Identification**

Name of application is OneStop Menu. Version is 1.0. Published by Kacheef White.

* 1. **System overview**

This application is vital for customers to see the food variety of the restaurant. It will allow a fast process for workers to prepare food at a faster time.

* 1. **Document Overview**

This document will contain all the requirements for the application, the design used to fulfill these requirements, test results and user manual. This document will help future developers understand the application to help will updates.

**2.0 References**

https://beginnersbook.com/2014/06/vector-enumeration-example-in-java/

**3.0 Requirements**

1. The OneStop Menu shall contain only 12 menu items
2. The OneStop Menu shall allow an option to cancel order
3. The OneStop Menu shall respond appropriately for invalid entries
4. The items shall be selected using integer values
5. The OneStop Menu shall display the prices for each item
6. The OneStop Menu shall display the final order.
7. All incomplete order shall trigger a message and not be processed

**4.0 Design**

**Member Data**

int option;

Identifier for each option on the menu.

int choice;

Identifier for the item selected by the customer.

Vector order;

Stores all the item selected by the customer.

**Functions**

The function printMenu() constantly displays options of 3 items for 4 different categories. These categories are meat, bread, cheese and drink. The menu will also direct the user how to cancel an order. Switch statements will be used to separate the items in the 4 different categories. The function will take exactly one argument.

The function selectItem() will tell the user what has been selected and ensures a valid entry has been made. Customers will only select a number for their item choice. These items will be stored in a vector where the full order will be later shown. If the customer cancels the order, the order will be immediately erased. Switch statements will be used to prevent invalid entries. The function will take exactly 3 arguments.

**5.0 Test Plan**

For this application, unit test will be done for all test procedure. This will be done by the developer.

Testing will be done after requirement document is written. All addition requirements will be tested as soon as it is designed and coded.

Testing will be done using NetBeans IDE 8.2.

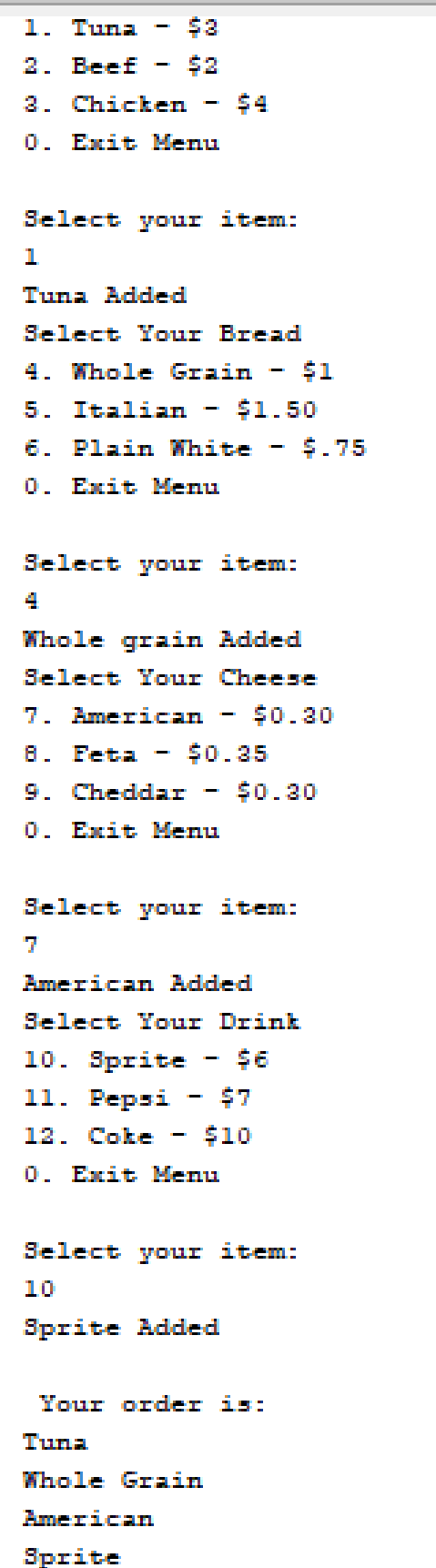
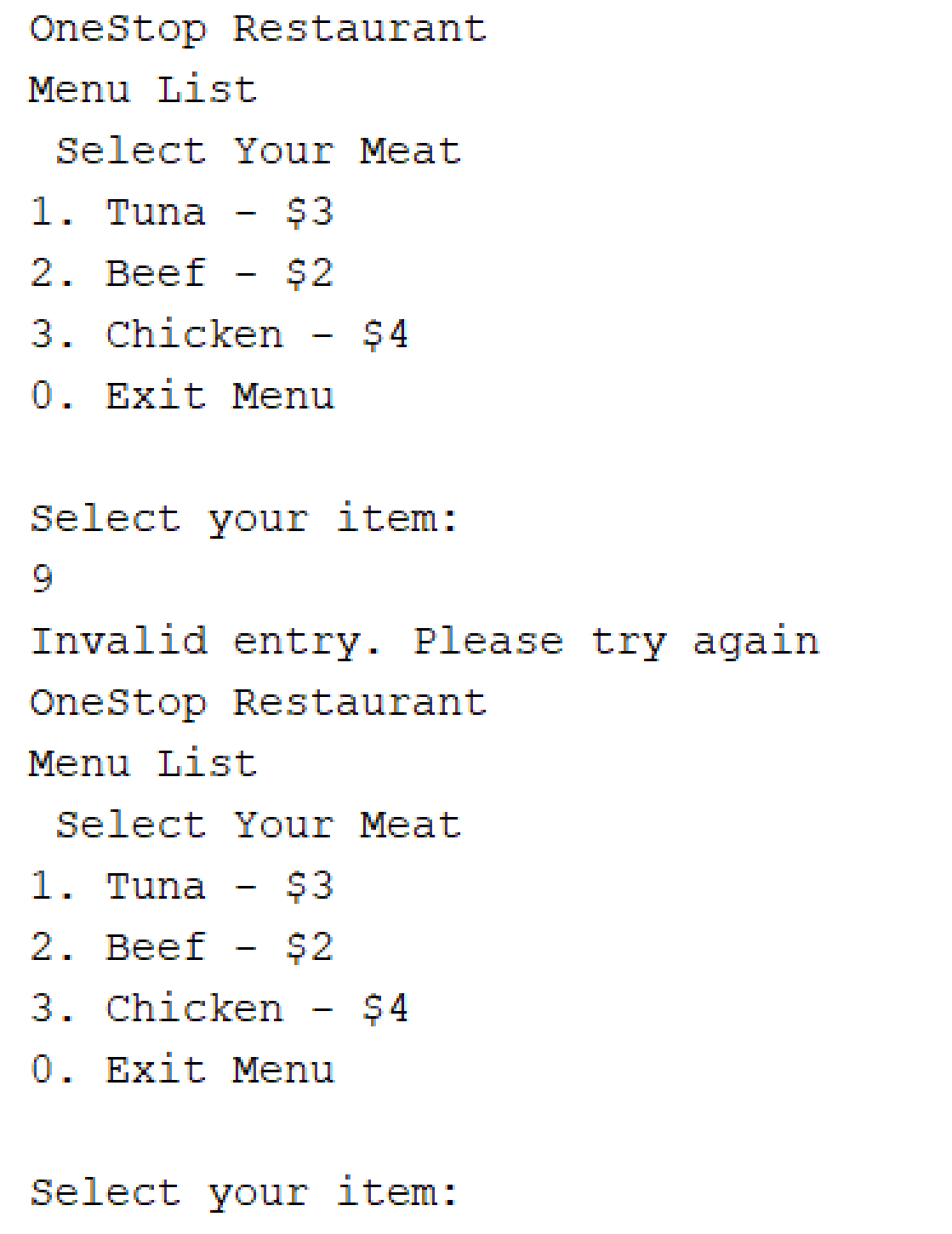
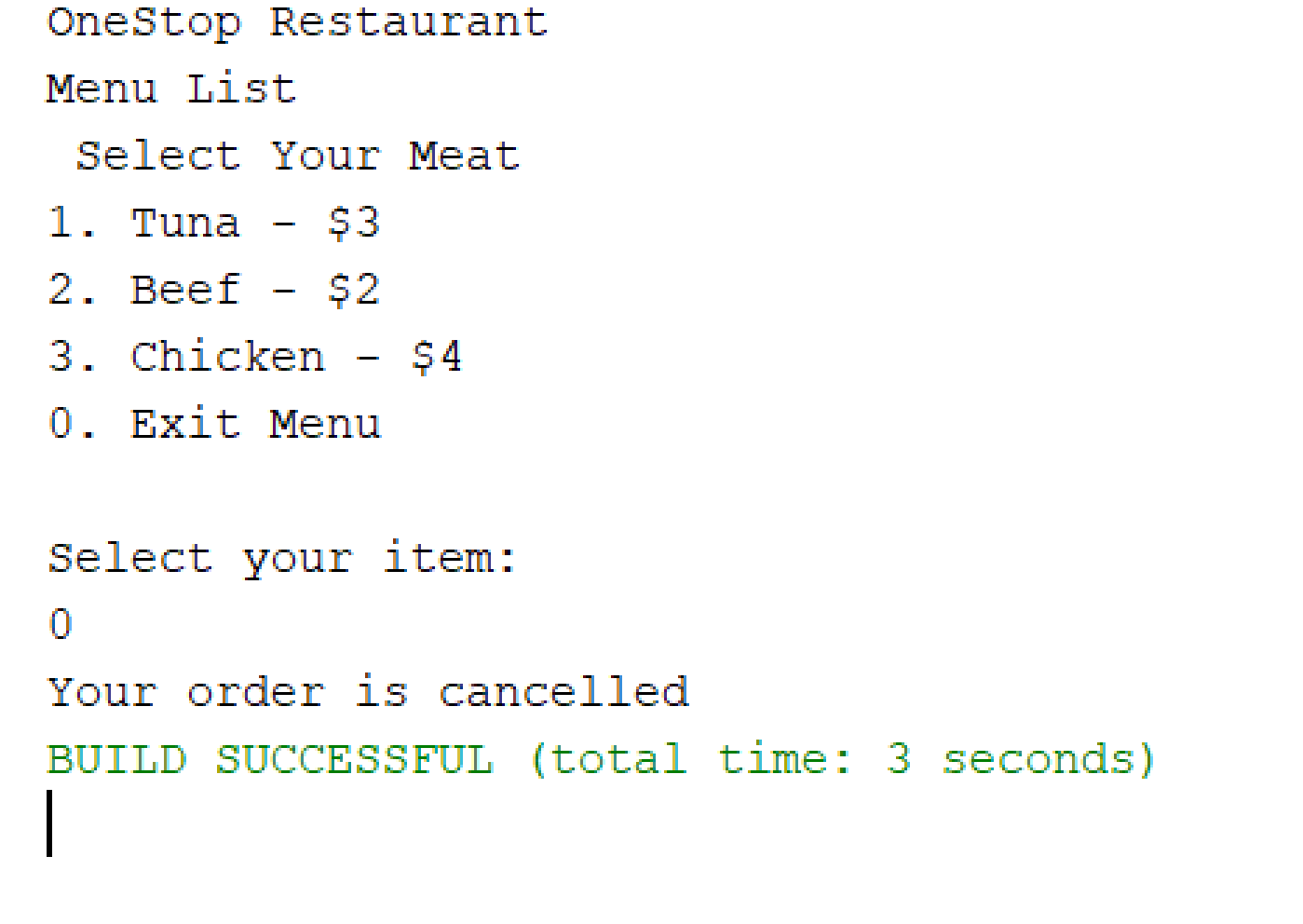
**5.1 Test Procedure**

Run the program. The output box should display the restaurant name, list of 3 meat options, the price of each item, an exit option and an area for the customer to input their choice.

1. Testing a completed order:
   1. Input a number between 1-3 for meat choice and press enter.
   2. Menu will display 3 choices of bread. Input a number between 4-6 and press enter.
   3. Menu will display 3 choices of cheese. Input a number between 7-9 and press enter.
   4. Menu will display 3 choices of drinks. Input a number between 10-12 and press enter. The menu will then display the correct order placed
2. Testing for a choice selected by customer that is not displayed on menu:
   1. Input a number other than 0,1,2,3 and press enter. An error should be displayed and the available options displayed again.
3. Testing for cancelled order:
   1. Input 0 and press enter. the order should display a message and cancel order

**Appendix**

Test Results

1.  A completed order
2.  Customer submit invalid entry
3.  Customer cancels order