PC BUILDER

Lorenzo
Brandon
Ethan

Introduction of the Project

Project Description

- A Java program that allows a Customer to have a shopping experience as they custom build a personal PC from an existing inventory of PC Components.

Minimum Viable Product

- As a guest customer I would like to navigate to the Main Menu from any Submenu
- As a guest customer I want to be able to see my current build progress and current price of each component.
- As a guest customer I want to be able to update my customer information.
- As a guest customer I want to be able to add my build to my shopping cart.
- As a guest customer I want to be able to view the items in my shopping cart.
- As a guest customer I want to be able to remove a build from my shopping cart.

- As a Customer I would like to be able to view and confirm my Order details in my current Session
- As a guest Customer I would like to exit my current Session from the Main Menu
- As a guest Customer I would like to add and update Payment Information
- As a guest Customer I would like to navigate to a Submenu where I can update my previous name and email

-

Project Repository

GitHub Page:

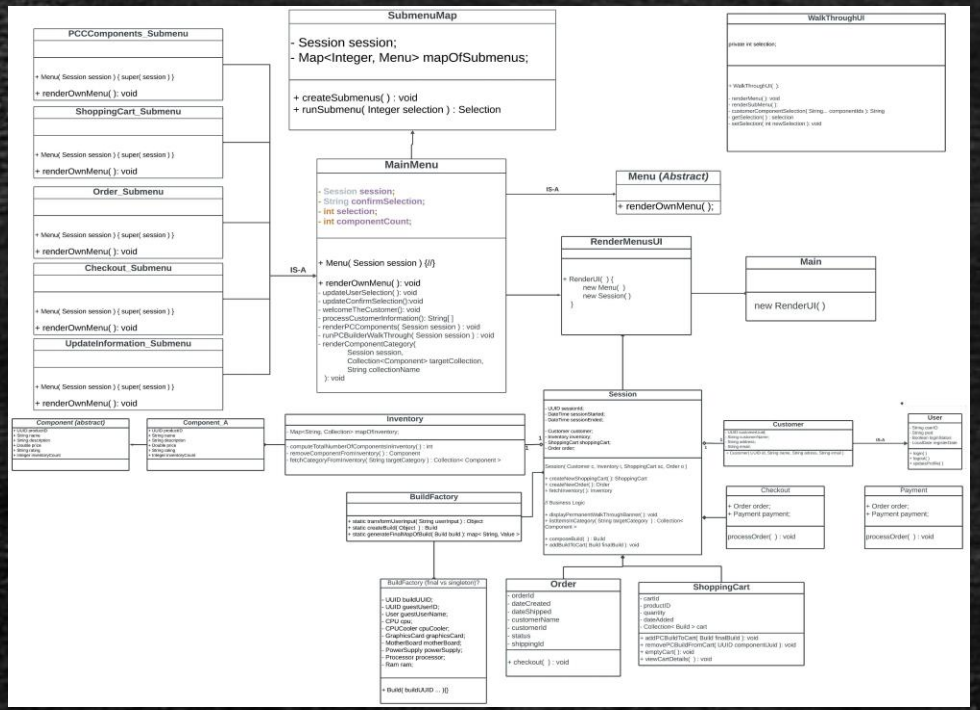
https://github.com/brandonyrigan/pcbuilder

Languages and Tools Used

- Java
- GitHub
- Git
- IntelliJ
- Microsoft Teams

Demo Time

UML diagram



LucidChart UML

Initial Planning

- Voted on the Best Idea
- Wanted the challenge of creating custom interface for a user
- We created a simple UML for how we could create the interface
- Outlined the Steps needed for the customer to complete a Purchase
- Implemented an initial proof of concept using the console

Code Fragments we are Proud of

Refactoring Switch Statements using a Map to render only the Target Submenu (gh code)

Displaying the current build progress and component prices

Extra Features

Prettify / Aesthetic Aspects (System output)

Explore and Add a DB for Persistence

Implemented a User Class

Add a GUI frame work like AWT or Swing

Team Experience in Coordination

- What worked well and what did not
- Readability of our code
- Clean code practices we implemented
- Technical Challenges We came across

Design Principles and Patterns Used

OOP Concepts

- Encapsulation
- Composition
  - Session Class has the Order, ShoppingCart, Customer, Checkout…
- Abstraction
  - Using Abstract Menus abstract method
  - Allows subclasses to render their own Menu when needed

Mistakes Made and Lessons Learned

Victims of our own Ambition

Spent too long on one problem

Simple mistakes

Learning how to use IntelliJs diff UI

Running into Gituations