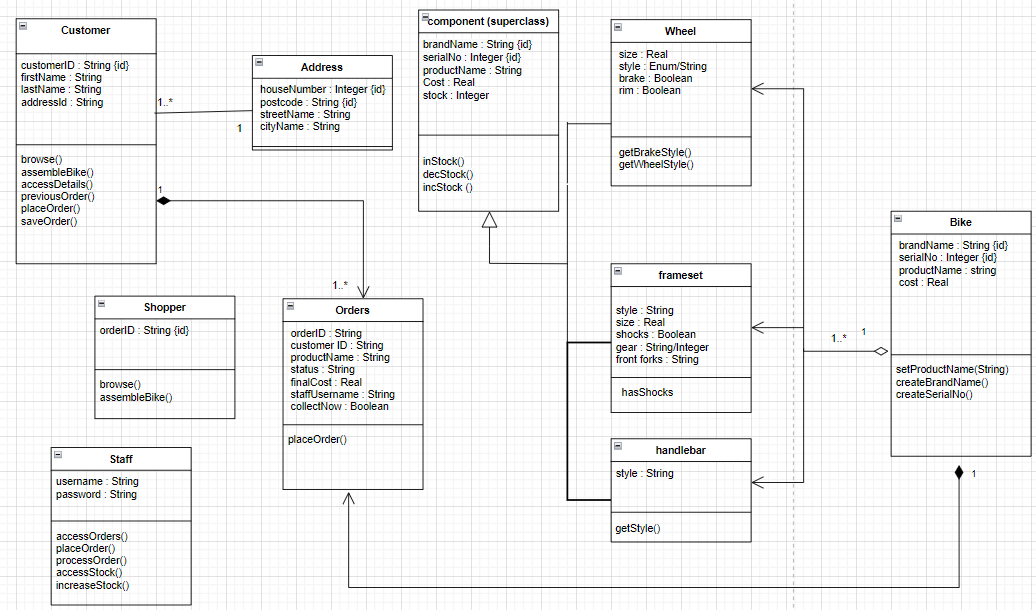
Introduction

We feel as though we have closely followed the brief to the best of our abilities, though a couple of assumptions were made.

* We have assumed that all bikes have an additional £10 assembly charge.
* We have assumed that a customer can’t have multiple bikes in one order, and if they want to order multiple bikes they will have to make multiple orders.
* We chose our own hash to use for the passwords: SHA
* When making the serial numbers for each component, to make sure it was random, we just hashed the brand name and product name.

UML Class Diagrams

These are our UML Class Diagrams. This is our initial analysis of the brief and of the system.



UML Class Diagram of the Normalised Database

These are the UML Class Diagrams of the normalised database. For each table primary keys have been identified, and foreign keys have been identified when needed.

Diagram, schematic

Description automatically generated

UML State Machine Diagram

These are our UML state machine diagrams. They show the different states of the system and what transitions can be performed in these states.

A picture containing text, sky, map

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Security Features

We have implemented various security features. One of them being that the staff passwords are stored in the database as hashes. This ensures that if unauthorised personnel ever did get a hold of the table they would not be able to view the passwords – and so would not be able to login to the system. And when a user logs in, their input string is hashed and compared against the hashes in the table to see if they match.

Another security feature that we implemented was that if anyone typed in SQL commands into the text boxes they would not be carried out. We achieved this by using prepared statements. This ensures that a user could not do something like delete the database or make any unauthorised changes.

Furthermore, to get onto certain parts of the system the staff need to input their username and password. This ensures an extra level of security and authentication.

Customer and staff don’t share pages, this means that all the parts of the system are separated. So, there is no chance of a customer accidentally viewing or changing anything that they aren’t supposed to.

* Allow customer to sign in with name and address, or with order number.
* Customer can see their orders (pending and previous)
* Customer can delete pending orders
* Staff can login
* Staff can view all orders

Working System Screenshots

Graphical user interface

Description automatically generatedMain menu Customer Login with details

Graphical user interface, application

Description automatically generated

Customer login with order number Staff login

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Browse with no options selected Browse with options selected

Graphical user interface, application

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

Graphical user interface, application

Description automatically generatedA picture containing text, sky, screenshot, day

Description automatically generatedCustomer main menu View Customer’s pending orders

Table

Description automatically generated with low confidenceAllow customer to view customer previous orders Allowing customers to edit their details

Graphical user interface, application, Teams

Description automatically generated

Graphical user interface

Description automatically generatedGraphical user interface, application

Description automatically generated with medium confidenceStaff main menu Staff view all orders

Changing the status of order Adding component

Graphical user interface, application, Word

Description automatically generated

Graphical user interface, application

Description automatically generated

Group Contributions

This table displays all the tasks that everyone did.

|  |  |
| --- | --- |
| **Name** | **Tasks** |
| Vernon Yankson | * Initial UML Information model * GUI design * Initial GUI prototype * Staff changing order status * Staff view all orders * Initial dummy code for staff and customer * GUI for staff login * Part Selection get and set methods |
| Zakariyah Yunis | * Initial UML Information model * Verifying the hash when logging in * Staff making orders * GUI looks * Created methods for edit customer details, edit order status, get order list * Initial dummy methods for components based on UML diagrams * GUI interface for staff login, view orders, staff menu and staff orders |
| Noor Elreidy | * Initial UML Information model * Created the database tables in MySQL * GUI looks * Allow staff to add new components * Allow customer to edit details * Bike product name generator * Allow shopper to place first order then register * Allow shopper to place order * Initial dummy methods for users based on UML diagrams |
| Noor Sharif | * Initial UML Information model * Encryption: used for password hash * GUI Looks * Allow shopper to browse through components and assemble bikes * Initial dummy methods for components based on UML diagrams * GUI interface for staff login, shopper and customer browsing * Report * UML database diagrams * State diagrams |

This table displays everyone’s effort levels and has been looked over by everyone and everyone agrees.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Vernon Yankson | Zakariyah Yunis | Noor Elreidy | Noor Sharif |
| Effort | 25 | 25 | 25 | 25 |

Vernon:

Zakariyah:

Noor E:

Noor S: