

COM2008 Report Build-A-Bike

Jonathan Ruffles
Kush Bharakhada
Kanghua Liu
Zhongjie Huang

Interpretations of the Brief

The brief explains that we are to design and implement a bicycle building application using Java Swing and JDBC. The application is to be used in-store on a computer of sorts, where a customer can create a bicycle and then pay at the counter.

A person is considered to be a shopper if they have not placed an order on the system. They are a potential customer and only become a customer when they have created a bicycle and saved it. Staff members have pre-existing logins.

When a shopper places an order (and becomes a customer), it becomes the responsibility of the staff member to process the order after said person has paid. The payment system is modelled separately to the system we are creating.

It is the responsibility of the customer to ensure that they do not lose their credentials. The only way a staff member can authenticate a customer is through their order ID, or personal details (first name, surname, address).

As a staff, it is their responsibility to manage all of the orders in the system, along with the warehouse stock. For example, if a new shipment has arrived to the warehouse of, for example, 15x 'Wheel X', then the staff is responsible for updating the quantity available to the customers through the staff dashboard.

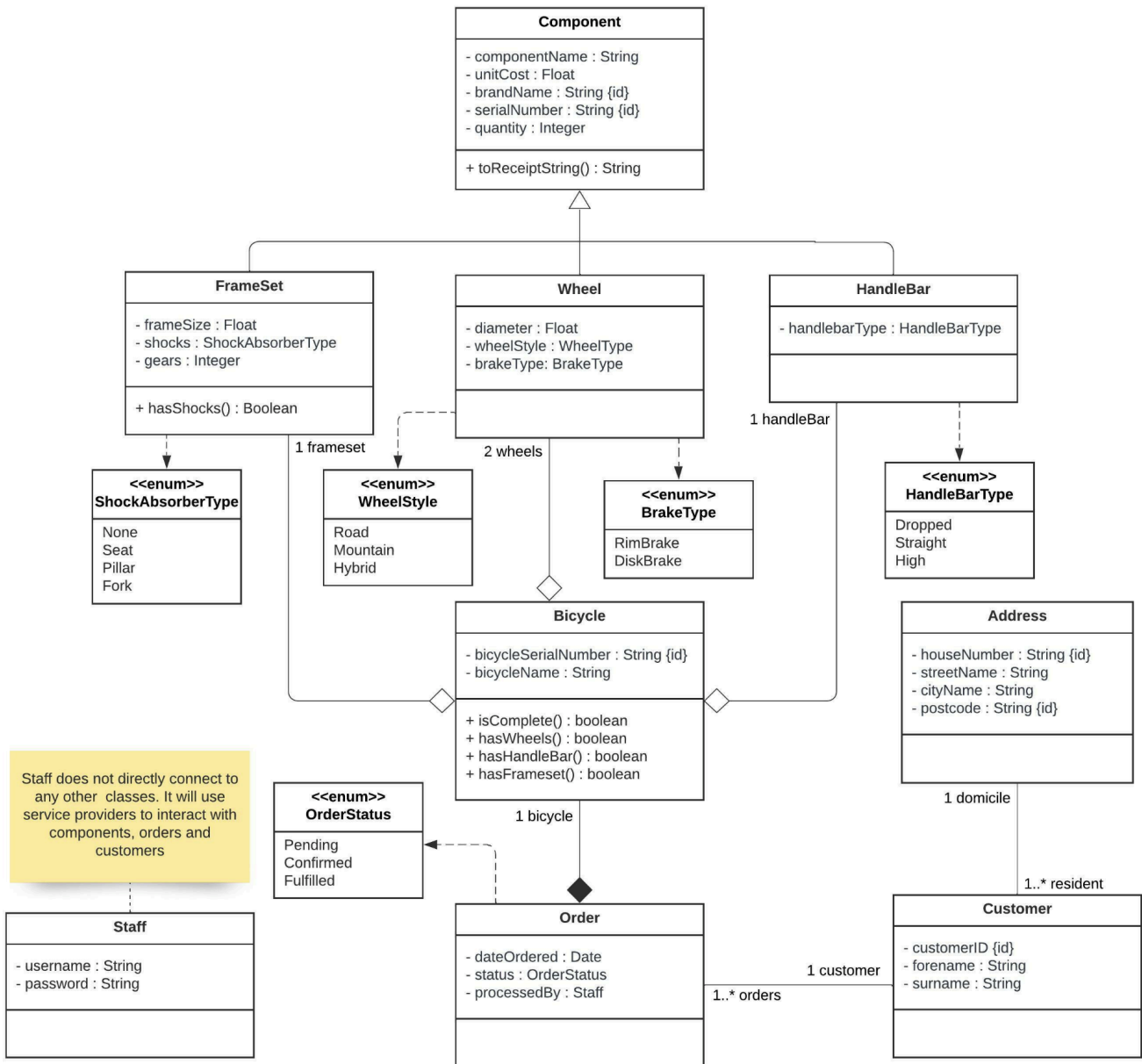
Additionally, a staff member cannot 'forget' their user credentials as the brief does not state any password recovery methods.

We also interpreted that the wheels on the bike are of the same type: a customer is only allowed to select one wheel type. This is because bicycles usually have the same front and back wheel type.

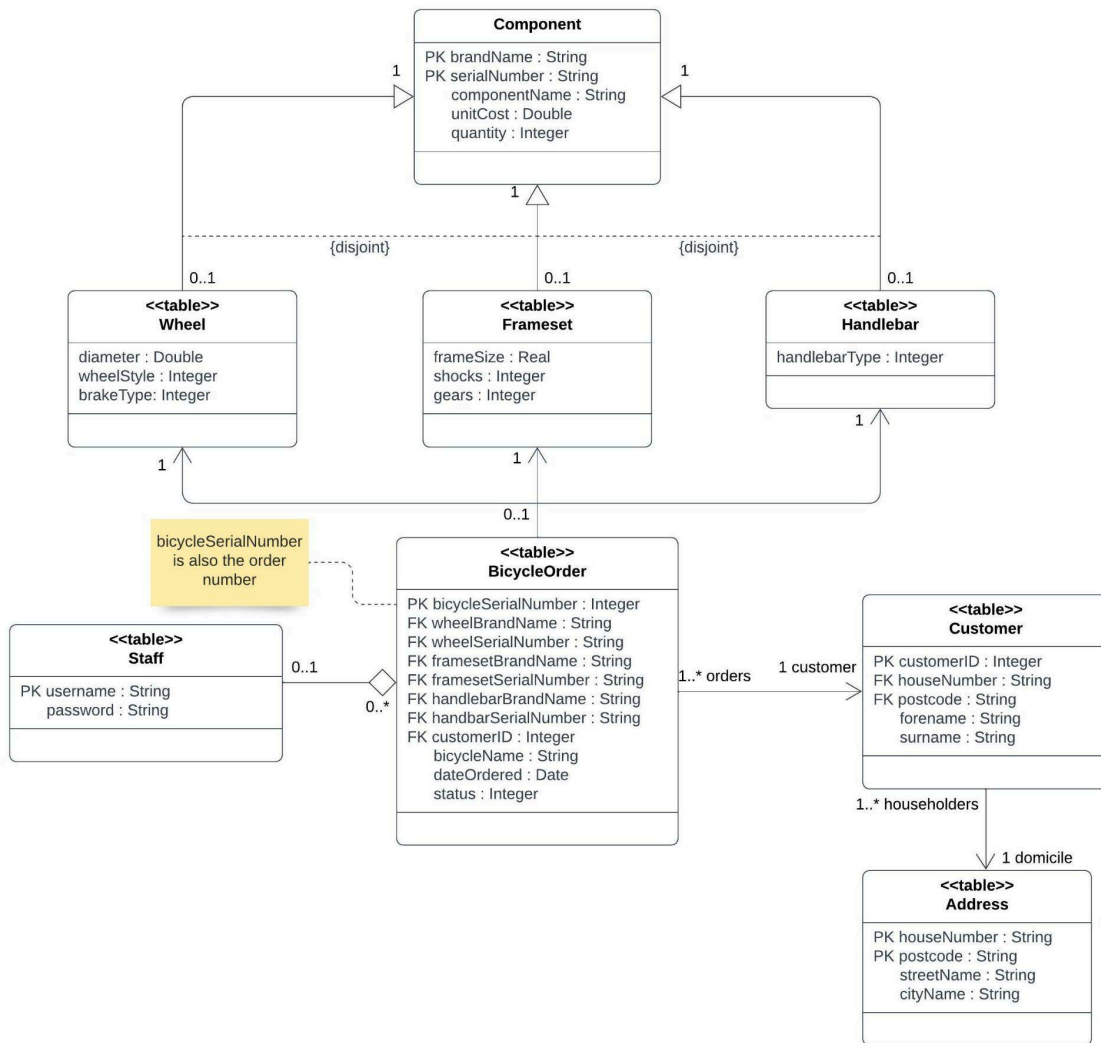
We also made the interpretation that the shock absorbers can be built on either the fork, pillar or seat (or an option for no shock absorbers).

We have interpreted the 'Pending' order status as a bicycle that has not yet been processed by a staff member yet. We have interpreted 'Confirmed' as a bicycle that has been paid for at the counter. We have interpreted 'Fulfilled' as when the bicycle has been constructed, and the quantity of the components are reduced.

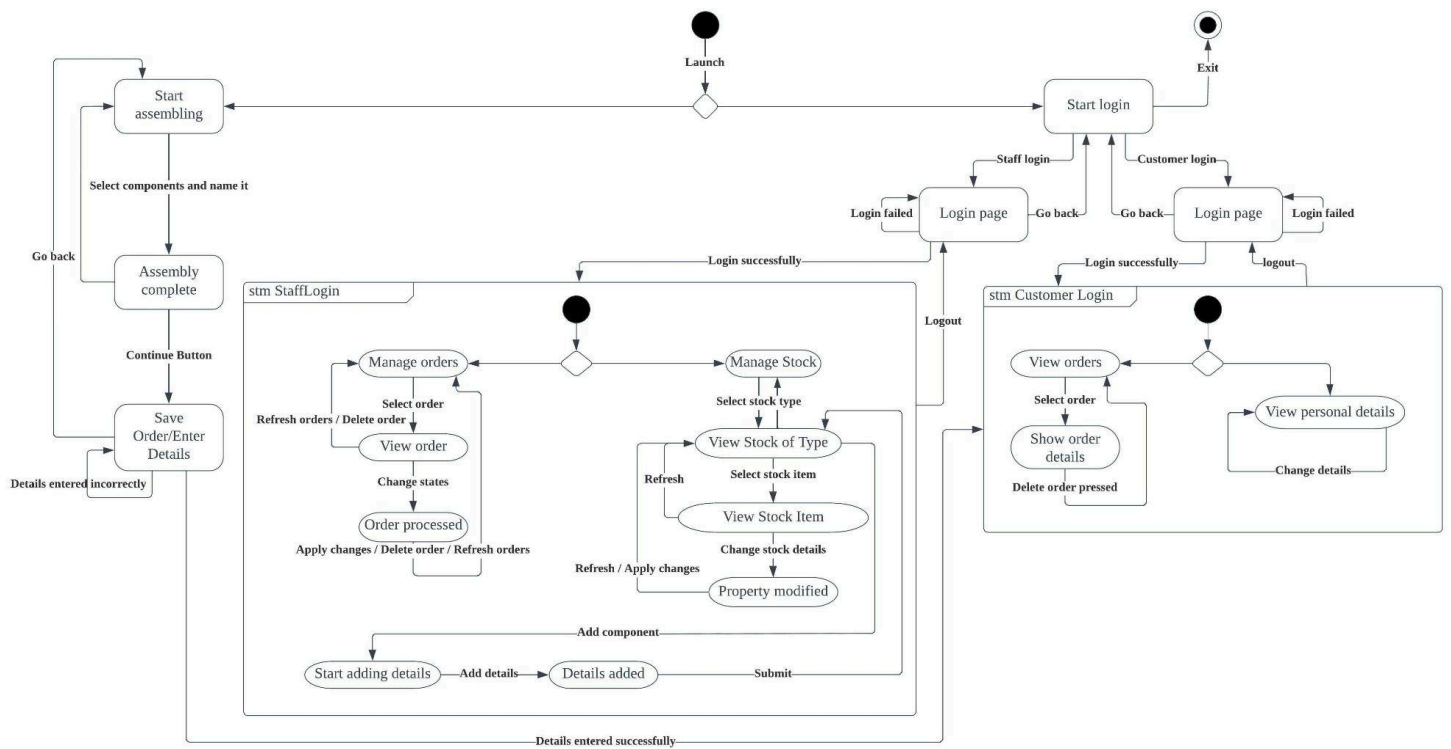
UML Class Diagram of the Initial Information Model



UML Class Diagram of the Database Model



UML Diagram of the State Machine



Screenshots of the System

Main Build Bicycle Page

WELCOME TO BUILD-A-BIKE LTD.

Customise Your Bicycle!

Click 'CHECKOUT' once your bicycle is complete

Bicycle Name

Wheel Type

Frameset

Handlebar Type

Login:

Items in Basket

Custom Bicycle Name: Example Bicycle Name
Bicycle Brand Name: raleighroad

Receipt

Product Details	Quantity	Unit Cost	Amount
Wheels Name: Mach 1 240 Rim, 8/9 Speed Cassette Brand: Raleigh Serial Number: 3 Wheel Style: ROAD Brake Type: RIM	2	£67.99	£135.98
Frameset Name: 700c Rim Brake QR Wheel Brand: Raleigh Serial Number: 1 Frame Size: 69.85cm Shock Absorbers: NONE	1	£31.99	£31.99
Handlebars Name: 02 Road Handlebar Brand: Deda Serial Number: 1 Type: STRAIGHT	1	£16.0	£16.0

Total: £183.97 + £10.00 Assembly Fee

End of Receipt

Continue

Search Order Menu

Order Number

Order Number:

Find My Order

My Details

Forename:

Surname:

House Number/Name:

Street Name:

Postcode:

City:

Find My Order

[Go Back to Building a Bike](#)

Order Overview

Order 1

Custom Bicycle Name: Example Bicycle Name
Bicycle Brand Name: argon 18mountain

Receipt

Product Details	Quantity	Unit Cost	Amount
Wheels Name: Crossmax XL 27.5 Wheel Brand: MAVIC Serial Number: 1 Wheel Style: MOUNTAIN Brake Type: DISK Frameset Name: 18 E-119 Rim Frameset 2020 Brand: ARGON 18 Serial Number: 1 Frame Size: 150.0cm Shock Absorbers: SEAT	2	£199.0	£398.0
Handlebars Name: 02 Road Handlebar Brand: DEDA Serial Number: 1 Type: STRAIGHT	1	£16.0	£16.0

Total: £3414.00 + £10.00 Assembly Fee

End of Receipt

Delete Order

Change Order Details

Order Id:
2

Date Ordered:
2022-11-27

Forename:
Kush

Surname:
Bha

House Number/Name:
89

Street Name:
Bobby Street

Postcode:
PO90 0P9

City:
Sheffield

Submit

Staff Order Management

Welcome Admin!

Logout

Manage Orders

Manage Stock

ID	Forename	Surname	Date	Address	Order Number	Status	Processed By
1	Kush	Bha	2022-11-27	89, Bobby Street, PO9...	1	PENDING	Nobody
1	Kush	Bha	2022-11-27	89, Bobby Street, PO9...	2	FULFILLED	admin
3	Jonathan	Ruffles	2022-11-27	56, Mulehouse Road, ...	3	FULFILLED	admin
3	Jonathan	Ruffles	2022-11-27	56, Mulehouse Road, ...	4	FULFILLED	admin

Change Status

Pending

Confirmed

Fulfilled

Refresh Table

Remove Order

Apply Changes

Search Order

Stock Management

Welcome Admin!

Logout

Manage Orders

Manage Stock

Wheels Properties

Framesets Properties

Handlebars Properties

Brand Name	Serial Number	Component Name	Unit Cost	Quantity	Diameter	Wheel Style	Brake Type
Raleigh	3	Mach 1 240 Rim, 8/9...	67.99	10	15.5	ROAD	RIM
Raleigh	4	26 Front Rim Brake N...	62.99	10	15.5	MOUNTAIN	DISK
Raleigh	5	Rr Wl 26 50 H 00	77.99	10	15.5	ROAD	DISK
Shimano	1	RS100 700C 9-11 S...	99.99	15	66.39	MOUNTAIN	DISK

Component Name

26 Front Rim Brake Nutted Wheel

Unit Cost (£)

62.99

Quantity

10

Wheel Diameter

15.50

Wheel Style

MOUNTAIN

Brake Type

DISK

Apply Chang

Refresh

Add Compon

Create a Wheel

Brand Name

Serial Number

Component Name

Unit Cost (£)

Start Quantity

Diameter

Wheel Style

ROAD

Brake Type

RIM

Submit

Discussion of the Security Features

We decided to use the AES encryption algorithm for our encryption method. This is because it is symmetric, so we can do encrypted SQL queries.

Using AES, we encrypted the customer's first name and forename in the table. By encrypting the customers credentials, any attackers that gain access to the database won't be able to manage an order maliciously.

For the staff database, we use hashing and salting (PBKDF2 with HMAC-SHA1) to store the passwords. Hashing is a one-way algorithm that means you cannot decrypt it. This means that once we have stored the password we cannot get it as plaintext again. When we attempt to login as a staff, we hash the inputted password and compare the result in the database. This means that if an attacker was to get access to the database, they would not be able to log in as a staff member.

When handling passwords in the GUI, we made sure to use the JPassword class. This handles the front-end for us by filling in the characters typed by the user with asterisks, making it impossible for people to snoop over the staff's shoulder to get the password. Additionally, the JPassword class handles the password as a char array instead of a String type. This means that we are less likely to accidentally pass the unencrypted password (wrongfully) into a display field or parameter we shouldn't.

Finally, we use JDBC's PreparedStatement class to prepare any SQL statements we send to the database. This is so that we prevent any SQL injection that an attacker may try on the system. We also sanitised user input fields to ensure the data was clean before being inserted into the database.

Work Distribution

Name	Factual Account	Contribution Points
Jonathan Ruffles	<p>I was the team leader for the second half. This role entailed making sure that we finished the project on time, and that the tasks assigned to everybody were completed in a timely manner. We communicated this through Discord and the Trello board (I set up a Trello account and invited all team members so we could have a central plan of what needs to be implemented or changed.).</p> <p>I conceptualised and wrote the backing ServiceProvider class that provides services to the views through a rudimentary Dependency-Injection style system.</p> <p>I wrote the SQL statements for OrderService and CustomerService. This includes joining data from other tables to make sure that the classes representing the table data are fully filled. The methods for each service were created on demand for what the team required at the time.</p> <p>I led the conceptualisation of the database model with the team, making sure it was normalised to the 3NF.</p> <p>I created the abstraction layer for most of the staff dashboard and the table models. This allowed for reusable and easily expandable dashboard panels that Kush could implement.</p> <p>I also helped Kanghua and Zhongjie refactor the AESUtil class to support encryption of strings to prevent duplicate code.</p>	30
Kush Bharakhada	<p>I was the team leader for the first half of the project to ensure everyone was delegated the task they were best at and ensure clear communication between everyone in the team. I organised regular weekly in-person meetings where everyone showed what they had completed and what they were working on.</p> <p>I did the staff login service in the back-end including the security for hashing staff credentials using PBKDF2 and HMAC-SHA1. I added some of the components into the database and created the staff logins. I sanitised user inputs and used regex to ensure data was clean before being inserted into the database.</p> <p>I was in charge of the front-end development therefore I created the majority of the front end using Java layout managers and no GUI builders to ensure the front and back-end were fully understood. With Jonathan's help, I connected the front-end to the back-end so inputs from the UI fed as expected into the database. Me and Jonathan regularly pair programmed to ensure we understood exactly how the front and back-end functioned.</p> <p>With the team I led the design of the information model and I refactored the information model to ensure it was of good quality. I also contributed to the database model with the team and refactored it with Jonathan. I checked Zhongjie's state diagram to ensure it matches the system requirements and gave feedback.</p>	30

Kanghua Liu	<p>I generally finished the work assigned by the team leader. I would discuss my query during the meeting to ensure good communication among team members.</p> <p>I did the backend work. I participated in the creation and modification of CustomerService, OrderService and ComponentService. With the conversation with the team members, we planned the appropriate methods in need based on the assignment brief for the whole bicycle system. Among these services, I coded methods that can create a customer account and add a new order etc.</p> <p>I did the AES encryption and decryption with Zhongjie. It is supposed to be the RSA, but in that case, the same data will be encrypted into different data each time, which is unwell for another class's development. So we choose AES instead. I wrote a class to generate the key automatically if the key does not exist and a util class, with the help of Jonathan, support encrypting and decrypting some essential data in the database.</p> <p>For the UML diagrams, I provided my thoughts on each diagram. I checked some of them and made some revisions according to the final construction of the bicycle system.</p>	20
Zhongjie Huang	<p>I was mainly responsible for back-end development and drawing in UML in the team, and tasks assigned to me after a conversation with the team.</p> <p>I drafted the initial model of the project, including some basic classes, instance variables and methods, and quickly implemented the development of the LoginService and CustomerService as well as the interaction between Java and MySQL, and made several changes in follow-up conversations with the team members. Each time a task was completed, I discussed it with the team and got their approval before formally submitting it, unless it was a very minor issue.</p> <p>I drafted ideas about the creation and modification of database tables in the Java code, and likewise I implemented the JDBC functionality. Although some disagreements occurred in subsequent discussions with the team members, we successfully reached an agreed solution through a friendly and collaborative approach.</p> <p>I discussed with Kanghua on the encryption and decryption of the data and communicated the problems encountered to the rest of the team and eventually adopted the AES encryption technology.</p> <p>In terms of UML, I worked with other team members to draw different versions of the information model and state machine diagram, and after receiving feedback from the team members, I made changes to help the team optimise the final version.</p>	20