Background:

My name is Mrs Pat Cake. I own a bakery called "To Pie For". I need to move with the times and therefore require an e-commerce web site to be built for me. This will provide my customers online shopping facilities from their homes and workplace. Very important for me during this COVID-19 period. Here are some of my requirements.

Any potential customer must be able to sign up for free. Upon registration they will need to enter their name, contact number, email address and a potential delivery address. Once signed up, they should be able to log in to their account and browse all my awesome, delicious products they may want to purchase as their interest is piqued.

Once on the site, a customer should be able to select a category of products they would like to browse. For instance, cookies, cakes and cupcakes, fresh breads, pastries, pies, and brownies. As the customer browses, they should be able to view prices and details of any selected items. These details must include any warnings (e.g.: contains nuts) as well as nutrient information and ingredients contained in the product.

Should the customer want to purchase a product, they will need to select it and then place it into their shopping cart. The customer must be able to continue to shop for more items after placing an item into the shopping cart. Once completed with their shopping, the customer should proceed to the 'checkout'.

At checkout time, all items placed in their shopping cart, along with the total amount to be paid is displayed as "order ready" to the customer. Should the customer not want to continue with the order and cancel the process (s)he should be able to do so. An option of being able to edit the order must be available before a customer makes a final acceptance.

If (s)he proceeds with accepting the order, more information is required to complete the transaction. The customer will be asked to fill an e-form providing information about the shipping address or confirm a delivery address previously used. They will also be required to select a payment option such as credit/debit card or EFT (incidentally, you do not have to implement a payment gateway, just create a 'stub' that will randomly accept and reject payments).

Once an order is placed, an email notification must be sent to the customer acknowledging the order request. For verification, the user must be e-mailed a copy of the shopping receipt after a successful payment transaction.

Functionality:

- The system must provide a convenient interface for
  - User Registration,
  - Categories,
  - Item Search and
  - Payment
- Any guest to my web site must be able to view all my products.
- For any kind of shopping, the user must be registered and logged on to my system.
- My system must be able to show bakery products in their respective categories.
- My system must generate an invoice, which the customer must pay online.
- My system must provide a 'Password Recovery' facility through the customer's email.
- An administrator of my system must be able to assign different options for sales (e.g.: 10%, 15%, and 20% discounts) on various bakery items.
- If an item is not available ('out-of-stock') and a customer selects it, an alert should be shown to the customer that this item is 'out-of-stock'. At present I do not want any orders to be placed on 'out-of-stock' items. My system must manage my stock efficiently.
- The customer must be able to view all their orders they placed.
- The administrator must be able to trace any order placed.
- My system must have a convenient 'search' allowing for the searching of any bakery item.
- Every order must be allocated a unique identifier (ORDER_ID).
- Likewise, each invoice must be associated with an order, and it too must have a unique identifier.
- My system must have both an administrator and customer role.
- The customer must be able to receive notifications via their email account.

Reports:

Reports, *inter alia* I require include the following:

1. <u>Orders placed:</u> This list can be generated alphabetically, or by category. I must be able to enter a single day or a range of days.
2. <u>Orders outstanding:</u> Keeping track of which orders have not been delivered. I must be able to enter a single day or a range of days, as well as a category.
3. <u>Orders delivered:</u> This list may be generated alphabetically or stipulated period.
4. <u>Ingredients in stock:</u> This list shows the ingredients in stock, in alphabetical order.
5. <u>Ingredients required to be ordered:</u> This list shows the ingredients, in alphabetical order I need to purchase as my stock is running low.

Other information:

"It is a requirement that I be able to change my prices when needed. I would also like to have the functionality of giving customers a discount, if deemed appropriate."

"The system should allow a customer and administrator to add/edit their profile details, and if necessary (not often) delete a customer. Customer details to be stored include, but are not limited to:
Name
Surname
Title
ID. Number
Delivery Address
Telephone Number (Mobile)
E-mail address"

"The system must allow me to add/edit all item details, and if necessary (not often) delete an item. Item details to be stored include, but are not limited to:

Item title
Item description
Item warnings if any
Nutrient information
Ingredients
Picture of my creation
Category item belongs to"

"The system must allow me to add/edit all category details, and if necessary (not often) delete a category."

"If possible, and time permitting I would really like my system must have a metrics/analytics section that is viewable by an administrator of my system (requiring very basic 'data mining'), which displays graphically:

- Number of orders placed per day, over a week (Monday – Sunday). This will provide me with information about which days are the most popular.
- Chart (pie) displaying the popularity of the category of orders.
- Chart displaying the popularity of items per category."

Performance Requirements:

- Ability to manage many customers connected to the website at the same time without crashing!
- Good performance and data transmission.
- Once logged in customers should be able to purchase items and view orders without re-entering information.

Quality Attributes:

- Maintain a user-friendly environment that is visually appealing.
- Easy to see and navigate.
- Maintain readable content.

Other requirements:

The software should be able to run on any sort of environment, regardless of operating system. The database must be either MySQL or PostgreSQL and the application server must be Tomcat or Glassfish.

Summary:

"These requirements are the minimum expected by me, Mrs Pat Cake for my system. Should you want to add more functionality, please feel free to discuss this with the stakeholder, whom I have appointed and is currently your lecturer. Remember; discuss this specification with her/him. Make sure you understand what is required. Please note you have been given a very specific time frame to work in. Make sure you do not spend too much time on 'frills'; **rather, complete a system for me that does the basics very well than something that has frills yet does not work properly**."

You and your group are required to design an appropriate system incorporating the knowledge you gained during Mecer Inter-Ed's Java fundamentals, programming, and enterprise courses you attended. You and your group need to present a fully integrated web application. You may incorporate other technologies you know, for instance HTML5, CSS3 and JavaScript or a front-end development framework such as Bootstrap.

Lastly - make sure you understand what is required. If you are uncertain about any issue regarding this project discuss it with your lecturer.

Note: you have been given a very specific time frame in which you must complete your work. Make sure you do not spend too much time on the 'frills'; **rather complete a system that does the basics very well than something that has frills yet does not work**.

Guidelines for your system project.

1. Choose a group name.
2. Choose a group member to liaise with your lecturer. This person is the group leader.
3. Set regular meetings with the lecturer to discuss progress, questions, and ideas. These meetings can prove helpful in making design and coding decisions as well as assist in managing conflict within your group. The lecturer will act in an advisory role. Marks may be deducted should the lecturer be required to repair any design issues or code.
4. Regularly peruse the specification to ascertain whether your group has deviated from it. It is no good presenting in a fancy looking project that does not do what the user requires. Whilst frills will be considered in a positive light, not adhering to the specification will be marked down heavily.
5. Be careful of being too flashy.
6. Communication between your group members is of paramount importance. Hold regular meetings to make sure you are all "going in the same direction" – i.e., true north as taught you in DevOps. Decide on a set of standards that all members will adhere to. For instance.
   a. Naming of variables and packages
   b. Using the same colour-schemes and look and feel
   c. Ensuring there is a consistent user-interface across all screens
7. Ensure code re-usability through abstraction and encapsulation. Write it once, write it well, and distribute it to all members.
8. Ensure you separate all view, process, and data access logic.
9. Remember to 'code to the interface'.
10. Try, if possible, to apply as many of the design patterns you have been taught.
11. All group members must use the same JDK 1.8 version.
12. All group members must adhere to recommendations and standards as specified by MECER Inter-Ed and the Java developers' guidelines.
13. Ensure the database data source name used is unique.
14. Ensure your group has enough time to 'put the project together'. Do not meet on the morning of the presentation and expect everybody's contributions to integrate seamlessly.
15. Make sure you have adequate time to test your system.
16. Test your application on different architectures.
17. If a particular topic, item, or component is not officially part of the curriculum, but you as a group would like to implement it in your project, you may after discussing it with you lecturer.
18. Have fun!! You will be learning a lot.

Agreement

We, the group named _____ consisting of the following four members:

_____ signed: _____

_____ signed: _____

_____ signed: _____

_____ signed: _____

agree to abide by the following:

1. We understand that every member of our group will report to MECER Inter-Ed each day to develop the project together.
2. We understand that it is the duty of our group to report without delay anybody not contributing to the success of the project.
3. We agree that conflict resolution within our group is part and parcel of the "group" experience.
4. We agree that should a group not be able to resolve a conflict within the group, the lecturer will make the final decision to which all members of the group must abide, otherwise the group will be penalised.
5. We agree that no criticism of any member of our group will be made after the due date of the project.
6. We agree that all members of our group will be awarded the same mark unless determined by the lecturer and after discussion with MECER Inter-Ed management.
7. No person will be allowed to change groups, unless deemed necessary by the lecturer.
8. Presentations will be delivered as determined by MECER Inter-Ed management in line with the date specified on the course schedule.
9. We understand that **NO EXTENSION** to this date will be granted.
10. Each group will present their project on Friday 23rd February 2024.