Vegan Pizzeria

Customer

|  |  |  |
| --- | --- | --- |
| Use Cases | Task Owner | Notes |
| Register as a customer or sign in | Vic | Do we want an admin user to create employee users? |
| Place an order | Vic | * view menu (name, desc, price, pic?) * name, mobile, address? (GPS - check distance? delivery flat rate?) * <https://developer.paypal.com/docs/api/payments/v1/> |
| Check status of order | Jane | * automatically timed? When the initial order is placed the system should record the time. The other statuses should be set by the employees as they actually happen. |

Baker/Chef/Delivery person

|  |  |  |
| --- | --- | --- |
| Use Cases | Task Owner | Notes |
| List of orders with chosen status with function to change status. Possible statuses: ‘cooking’,  ‘ready for pickup’, ‘picked up’ or ‘delivered’ | Jane | This could be just one webpage with buttons on the top so other orders are hidden?   * Once set, have an automatic timer? (potentially different for each pizza type)   Dropdown for status? |

|  |  |  |
| --- | --- | --- |
| Databases required | Task Owner | Notes |
| Orders |  |  |
| Customers |  | GDPR - don’t ask for more info than we need. |
| Employees |  | It would be good to have a superclass for users that we extend for employees and customers so we probably want to have some common fields between customers and employees? |
| Pizzas | Vic | One of us needs to create the table but then we should probably both add to the table. |
| Animal Stories |  | One of us needs to create the table but then we should probably both add to the table. Might not need to be a table. Maybe use an API instead? <https://apislist.com/api/3/cat-facts>  <https://dukengn.github.io/Dog-facts-API/?ref=apislist.com> |

|  |  |  |
| --- | --- | --- |
| Other tasks | Task Owner | Notes |
| Presentation | Vic and Jane |  |
| Animal Stories and puns | Vic and Jane |  |
| Animal pictures | Vic and Jane |  |
| Random generation of stories |  |  |
| Rescue stories |  |  |
| Environmental impact |  |  |
| Mythbusters |  |  |

Started off with a separate baker and delivery person. Combined after Eugene’s suggestion.

Max 6 web pages including landing page.

**Sprint 1**

|  |  |  |
| --- | --- | --- |
|  | Vic | Jane |
| **Aim for the sprint** | **Customer places an order** | **Customer and Employees can check status of order** |
| MySQL Database | Create database  Create order, pizza and customer tables. | Create employee and orderlines tables.  Put in some sample data. |
| Create repo and packages.  Add this document and SQL scripts to repo. | Complete together on Zoom call on 28/5/2023.  Zoom Meeting link:  <https://crick.zoom.us/j/2464479323?pwd=dFpmblpWQlJLa09aQTRpYU5Pcnd2QT09> | |
| RowMappers | Customers  OrderLines  Pizzas | Orders  Employees  Login |
| Entities | Customers  OrderLines  Pizzas | Orders  Employees  Login |
| DAO interfaces and implementation classes | Customers  OrderLines  Pizzas  CRUD plus  Retrieve customer by email address | Orders  Employees  Login  CRUD plus  Retrieve employee by email address |
| Controllers | Customers  OrderLines  Pizzas | Orders  Employees  Login |
| html files | orderPizza | checkOrderStatusCustomer  CheckOrderStatusEmployee |

index.html will be in the next sprint when we do login.

Payment API will be in another sprint.

Animal Facts API will be in another sprint.

An automatic timer for cooking pizzas will be in another sprint.

**Initial ERD**

A screenshot of a computer

Description automatically generated with low confidence

**Wireframes**

Landing page

A picture containing text, diagram, screenshot, line

Description automatically generated

Login page

A screen shot of a login form

Description automatically generated with medium confidence

Sign-up page

A screen shot of a login form

Description automatically generated with medium confidence

Menu page

A picture containing text, diagram, parallel, line

Description automatically generated

Homepage - customer

A picture containing text, screenshot, diagram, number

Description automatically generated

Homepage - manager

Homepage - employee - chef

Homepage - employee - delivery driver

Place an order - what do we want our page to look like? Drop-down boxes? Or let them choose from the menu page? e.g. like Domino’s website. specify quantity, add to basket (adds to lineOrder). But then would need an extra page for the basket to display information, which could add extra work.

A screenshot of a pizza

Description automatically generated with medium confidence

A screenshot of a pizza

Description automatically generated with medium confidence

Alternative (Pizza Hut) - instead of specifying quantity, just add as many times as wanted

Background image from : <https://londoncitypizza.ca/why-should-you-search-vegan-pizza-near-me.html>



Pizza descriptions

1. Plain: A classic vegan delight with a crispy crust, flavourful tomato sauce, fresh herbs, and a touch of sea salt.
2. BBQ Jackfruit: Tangy BBQ jackfruit on a crispy crust, topped with colourful bell peppers, red onions, and vegan cheese.
3. Mushroom: Robust and earthy, this pizza features a blend of cremini, shiitake, and oyster mushrooms with vegan cheese and fresh thyme.
4. Roasted Veg: Bursting with flavours of roasted zucchini, bell peppers, eggplant, and cherry tomatoes, finished with herb-infused olive oil.
5. Olive and Sundried Tomato: A Mediterranean-inspired delight with Kalamata olives, sundried tomatoes, vegan feta cheese, and a drizzle of olive oil.

**Future Work (Beyond the scope of our current project)**

1. Make customer web pages flex to mobile screen size as this would allow customers to have a better user experience if using a phone instead of a desktop to order. It would be necessary for delivery drivers as they won’t have their desktops on their backs.

<https://www.w3schools.com/w3css/w3css_web_tmp_pizza.asp>

1. Different pizza sizes (large, small)
2. Create/customise own pizza
3. Use location to determine delivery availability and cost
4. Allow user to order without signing in (i.e. check out as a guest)
5. Other items (not just pizza)
6. Order history (previous orders)

References

**Resources for collaborating on GitHub:**

https://www.youtube.com/watch?v=\_wQdY\_5Tb5Q

Resources for using ThymeLeaf, SQL and Java date formats:

<https://www.baeldung.com/dates-in-thymeleaf>

<https://medium.com/@pdouvitsas/global-localdate-format-in-spring-boot-and-thymeleaf-29ff83b8f4c8>

Resources for passing parameters to other mappings:

<https://dzone.com/articles/spring-boot-passing-parameters>

Resources for CSS:

<https://css-tricks.com/emoji-as-a-favicon/>

<https://www.w3schools.com/w3css/w3css_web_tmp_pizza.asp>