|  |  |
| --- | --- |
| **Testing/Evidence Log** | |
| Pre-coding development:  Before anything else, I set up a Trello board as a way to keep track of my project management. This was what it looked like to start with. Then I did some research on similar programs, which can be found in the Research folder. While I was researching aspects of these programs, I noticed that they ask the user to enter the time they want to pick up their pizza or get it delivered. I talked to the stakeholder about this, and it was decided that this feature would not be necessary in the program that I am making. The second screenshot shows my board once I had finished my research and planning documents. The final screenshot shows the calendar of due dates for certain aspects of this project. As you can see, I was now running ahead of schedule. |  |
| In version one, I focused on getting the menu of pizzas set up and being able to loop through and print them all out. I felt that this was a good place to start, because everything centres around the pizzas in the program and I needed to make sure that it worked before I started on anything else. |  |
| In version two I focused on getting the input for the user to order a pizza. The user will enter the number that corresponds to the pizza that they want to order, and the program will store this and output the price of the pizza and the id that they inputted. At the moment, it only takes in the order for one pizza, but that will change later.  I tested this with a few different pizzas to make sure it was outputting the right price based on the menu the pizza was from. |  |
| In version three I added the capability for multiple pizzas to be ordered at once, and then calculated the total price of the order. The user enters the number of pizzas that they want to order, and each time the menus print out and the user enters what pizza they want. However, upon testing this, I realised a few problems. One was that the user had to decide how many pizzas they wanted to order from the beginning, before even being able to see the menu. Also, the large output of the menu happens every single time, which could be annoying for the user if they don’t want to see the menu again. So, I decided to try another way of presenting the output and get the opinion of a tester. |  |
| Here are screenshots of my trello board and trello calendar at this point in the development. Time-wise, I was a little under half way, and this time is represented pretty well by the amount of tasks in the done column. Also, as seen on the calendar, I was running a few days ahead of schedule. |  |
| In version four, I changed the input method to solve the problems I identified in version three. I used a menu based input method, so that the user was in control of when they wanted to see the menu, whether they wanted to see their order so far, etc. This way they don’t have to decide how many pizzas they want to order at the start – if they see the menu and decide they want to order an extra pizza, that’s fine. I got a tester to try both versions of the program, and got their opinion on which one was better, and what needed to change in the future. The feedback that I got was that version three (where they start with entering how many pizzas they want) was overwhelming because the long menu printed out every time. However, they liked the choice that the menu-based program provided, and how the menu did not print out unless they wanted it to. They gave the advice that I should change the output at the end after they quit, as it looked exactly like when they output during the order process. I have noted this, and I will change it when I get to changing the output at the end of the program.  One other thing that I changed is that, whenever the program is deciding whether a pizza is from the regular or gourmet menu, the comparison it performs is pizza id < 7. However, if another pizza was added to the regular menu, this would no longer work. It needed to be smaller than the first id in the gourmet menu, whatever that may be. So I changed this to pizza id < gourmet\_menu[0][2] (the third item in the first list). |  |
| As I was adding the stuffed crust option to my program, I came across a problem. I entered an order with two of the same pizza, except one had stuffed crust and one did not. However, it repeated the first order of this pizza twice. The second screenshot is the code for the order output before I fixed this problem. To fix it, I changed the “for pizza in order” to “for pizza in range(len(order))”. So, while originally pizza would be the id of the pizza, now pizza was the position of the id of the pizza in this list. This helped me get around duplicates of the same id in the list, as their pizza value will be different. The third screenshot is the code after I made the fix explained above, and the final screenshot is the same order but with the improved code. |  |
| In version five, I added the option for users to add stuffed crust to any of their pizzas. After every pizza, they are prompted as to whether or not they want to add stuffed crust to that pizza for the price of 3 dollars. If they answer yes, this price is added to the price variable. Regardless of what they answer, their answer gets appended to a list that keeps track of whether or not each pizza has stuffed crust or not. This is included in the output of the user’s order.  These screenshots provided are of various testing examples of this version. There is not a screenshot of the code because that was included in the previous log entry. |  |
|  |  |