Order analyser in food court

1.research:

An order analyzer in a food canteen provides users with speed, convenience, order accuracy, and personalized service.

Key benefits for the user:

- Reduced Waiting Times: Users can place orders remotely (from their desks or homes) via a web or mobile application, completely eliminating the need to wait in long queues, especially during peak hours.
- Improved Order Accuracy: The digital ordering process allows users to select and customize their meals precisely, which minimizes the risk of human errors or miscommunication that can occur with manual or verbal ordering systems.
- Convenient Access to Menus: Users can browse an up-to-date e-menu at their leisure, complete with item descriptions and prices, before making a selection.
- Real-Time Order Tracking: The system provides transparency by offering real-time updates on the order status (e.g., confirmed, preparing, ready for pickup, delivered), which helps manage expectations and reduces anxiety.

Canteen Food Ordering System and Management

2.ideate:

The main idea behind this project is to reduce human efforts in a service canteen or restaurant.

- This idea helps the owner to analyze the order given by the customer easily.
- It can also help to manage work load at times of rush.
- The order can also be prepared faster using this project.

3.analysis:

In today's world people do not compromise with food which is the right point of view for them . Now the job is the supplier to give a good service to the customer and satisfy them with good feedback.

4.build:

```
#include <stdio.h>

struct food {
    int food_number;
    char food_name[50];
    float food_price;
};

int main() {
     int i, j,f;
    printf("enter no.of dish:\n");
    scanf("%d",&f);
    struct food si[f];

    struct food temp_food;
    int n = f;



    printf("\n--- Enter Food Dish Details ---\n");
    for (i = 0; i < n; i++) {
        printf("\n--- Dish %d ---\n", i + 1);
        printf("Enter dish number: ");
        scanf("%d", &si[i].food_number);
        printf("Enter dish name: ");
```

```
        scanf("%s", si[i].food_name);
        printf("Enter price of dish: ");
        scanf("%f", &si[i].food_price);
    }


    for (i = 0; i < n - 1; i++) {

        for (j = 0; j < n - i - 1; j++) {


            if (si[j].food_price > si[j + 1].food_price) {


                temp_food = si[j];
                si[j] = si[j + 1];
                si[j + 1] = temp_food;
            }
        }
    }


    printf("\n--- Menu Sorted by Price (Ascending) ---\n");
    printf("%-10s %-20s %s\n", "Number", "Name", "Price");
    printf("-----------------------------------------\n");
    for (i = 0; i < n; i++) {
        printf("%-10d %-20s $%.2f\n",
               si[i].food_number, si[i].food_name, si[i].food_price);
    }

    return 0;
}
```

5.testing:

```
enter no.of dish:3

--- Enter Food Dish Details ---

--- Dish 1 ---
Enter dish number: 12
Enter dish name: coke
Enter price of dish: 20

--- Dish 2 ---
Enter dish number: 23
Enter dish name: sandwich
Enter price of dish: 70

--- Dish 3 ---
Enter dish number: 34
Enter dish name: cake
Enter price of dish: 180

--- Menu Sorted by Price (Ascending) ---
Number     Name                Price
-----------------------------------------
12         coke                $20.00
23         sandwich            $70.00
34         cake                $180.00


=== Code Execution Successful ===
```

6.implimentation: