FOOD ORDER SYSTEM:

USECASE DIAGRAM:

procurement manager

customers

prospective customer

create profile

expendite order

view menu

place order

change order

view reports

update prices

change menu
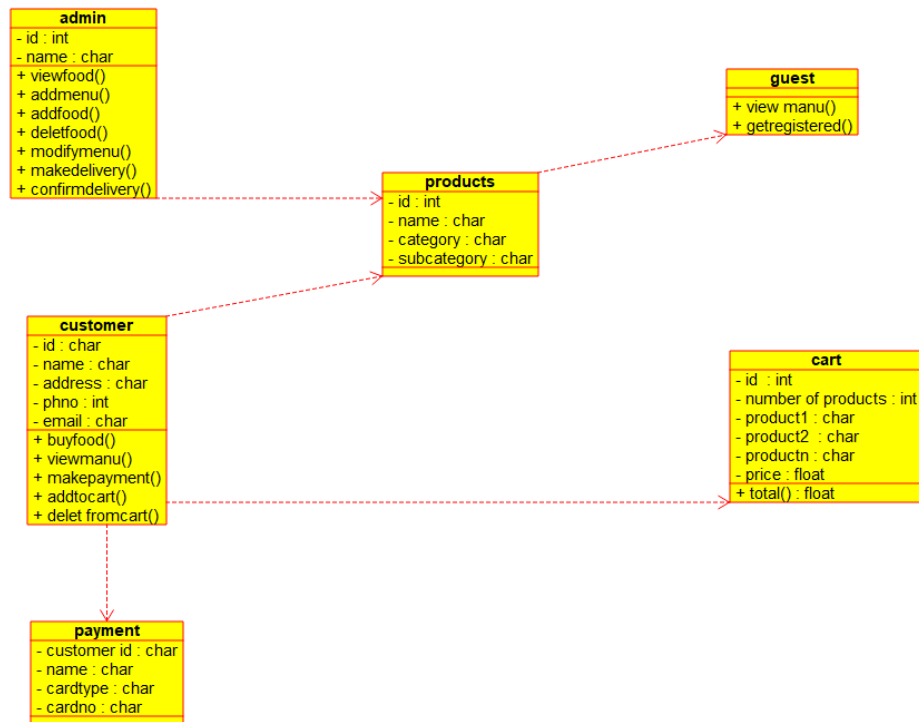
get ingredient request

service representative

meal manager

CLASS DIAGRAM:

CODING:

```
#ifndef ADMIN_H

#define ADMIN_H

#include <string>

/**

 * class admin

 *

 */

class admin

{

public:

 // Constructors/Destructors

 //

 /**

  * Empty Constructor
```

```cpp
 */
admin ();
/**
 * Empty Destructor
 */
virtual ~admin ();
// Static Public attributes
//
// Public attributes
//
// Public attribute accessor methods
//
// Public attribute accessor methods
//
/**
 */
void viewfood ()
{
}
/**
 */
void addmenu ()
{
}
/**
 */
void addfood ()
{
}
/**
 */
void deletfood ()
{
```

```
    }
    /**
     */
    void modifymenu ()
    {
    }
    /**
     */
    void makedelivery ()
    {
    }
    /**
     */
    void confirmdelivery ()
    {
    }
protected:
    // Static Protected attributes
    //
    // Protected attributes
    //
public:
    // Protected attribute accessor methods
    //
protected:
public:
    // Protected attribute accessor methods
    //
protected:
private:
    // Static Private attributes
    //
    // Private attributes
```

```cpp
 //
 int id;
 char name;
public:
 // Private attribute accessor methods
 //
private:
public:
 // Private attribute accessor methods
 //
 /**
  * Set the value of id
  * @param new_var the new value of id
  */
 void setId (int new_var)  {
    id = new_var;
 }
 /**
  * Get the value of id
  * @return the value of id
  */
 int getId ()  {
  return id;
 }
 /**
  * Set the value of name
  * @param new_var the new value of name
  */
 void setName (char new_var)  {
    name = new_var;
 }
 /**
  * Get the value of name
```

```cpp
     * @return the value of name
     */
    char getName ()   {
        return name;
    }
private:
    void initAttributes () ;
};
#endif // ADMIN_H
```