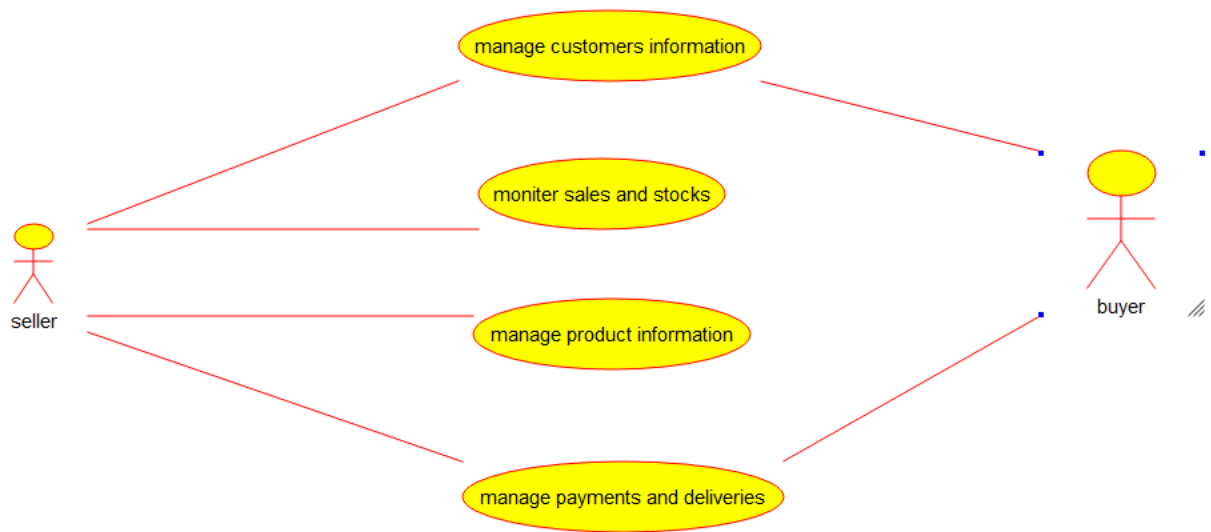
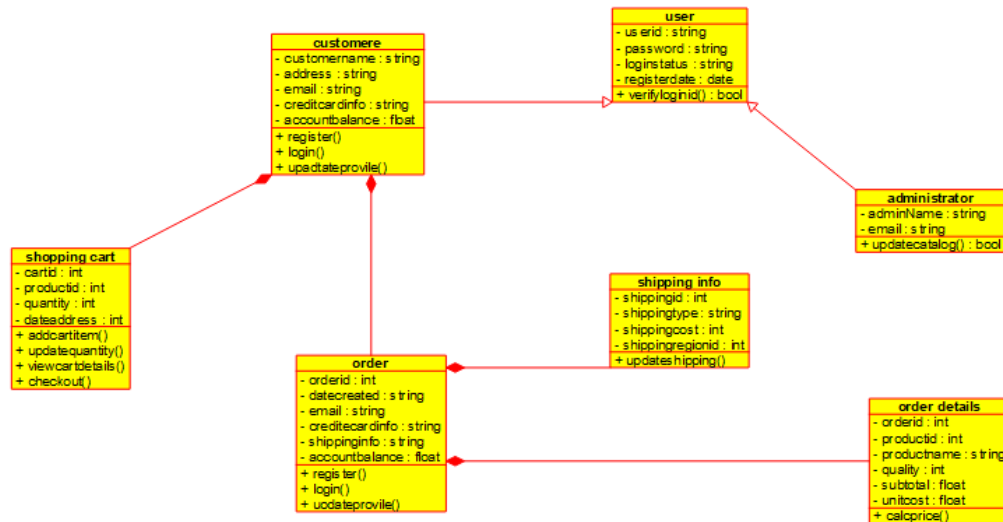


E-COMMERS SYSTEM

USE CASE DIAGRAM:



CLASS DIAGRAM:



CODING:

```

#ifndef ADMINISTRATOR_H
#define ADMINISTRATOR_H

#include "trade.h"
#include "item.h"
#include <string>

/**
 * class administrator
 *
 */

class administrator : public trade, public item

```

```

{
public:
    // Constructors/Destructors
    //
    /**
     * Empty Constructor
     */
    administrator ();
    /**
     * Empty Destructor
     */
    virtual ~administrator ();
    // Static Public attributes
    //
    // Public attributes
    //
    // Public attribute accessor methods
    //
    // Public attribute accessor methods
    //
    /**
     * @return string
     */
    string bill1 ()
    {
    }
    /**
     * @return string
     */
    string bill2 ()
    {
    }
    /**

```

```

    * @return string
    */
    string traceorder ()
    {
    }

    /**
    * @return integer
    */
    integer paybill1 ()
    {
    }

    /**
    * @return integer
    */
    integer paybill2 ()
    {
    }

protected:

    // Static Protected attributes
    //
    // Protected attributes
    //

public:

    // Protected attribute accessor methods
    //

protected:

public:

    // Protected attribute accessor methods
    //

protected:

private:

    // Static Private attributes

```

```

//
// Private attributes
//
string name;
integer id;

public:
// Private attribute accessor methods
//
private:
public:
// Private attribute accessor methods
//
/**
 * Set the value of name
 * @param new_var the new value of name
 */
void setName (string new_var) {
    name = new_var;
}
/**
 * Get the value of name
 * @return the value of name
 */
string getName () {
    return name;
}
/**
 * Set the value of id
 * @param new_var the new value of id
 */
void setId (integer new_var) {
    id = new_var;
}

```

```
/**
 * Get the value of id
 * @return the value of id
 */
integer getId () {
    return id;
}
private:
    void initAttributes ()
};
#endif // ADMINISTATOR_H
```