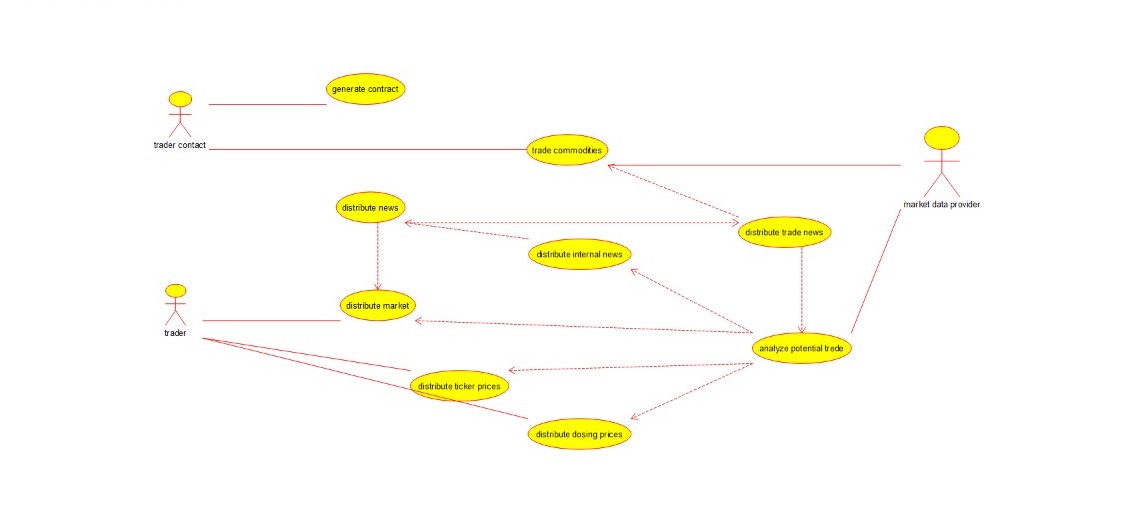
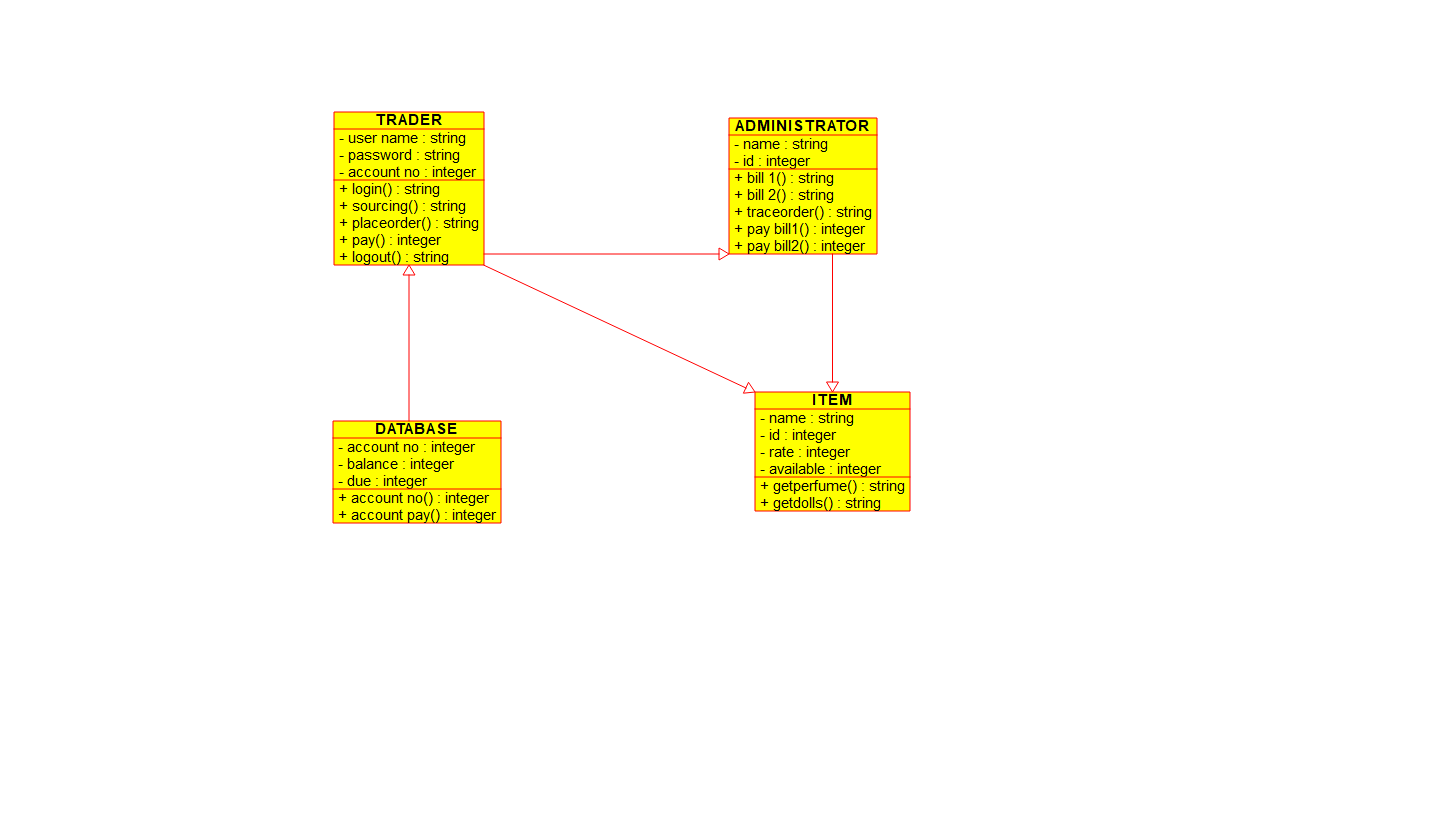
12.Foreign trading system

Use case diagram:



Class diagram:



Source code:

#ifndef ADMINISTRATOR\_H

#define ADMINISTRATOR\_H

#include "ITEM.h"

#include <string>

/\*\*

\* class ADMINISTRATOR

\*

\*/

class ADMINISTRATOR : 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 bill\_1 ()

{

}

/\*\*

\* @return string

\*/

string bill\_2 ()

{

}

/\*\*

\* @return string

\*/

string traceorder ()

{

}

/\*\*

\* @return integer

\*/

integer pay\_bill1 ()

{

}

/\*\*

\* @return integer

\*/

integer pay\_bill2 ()

{

}

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 // ADMINISTRATOR\_H