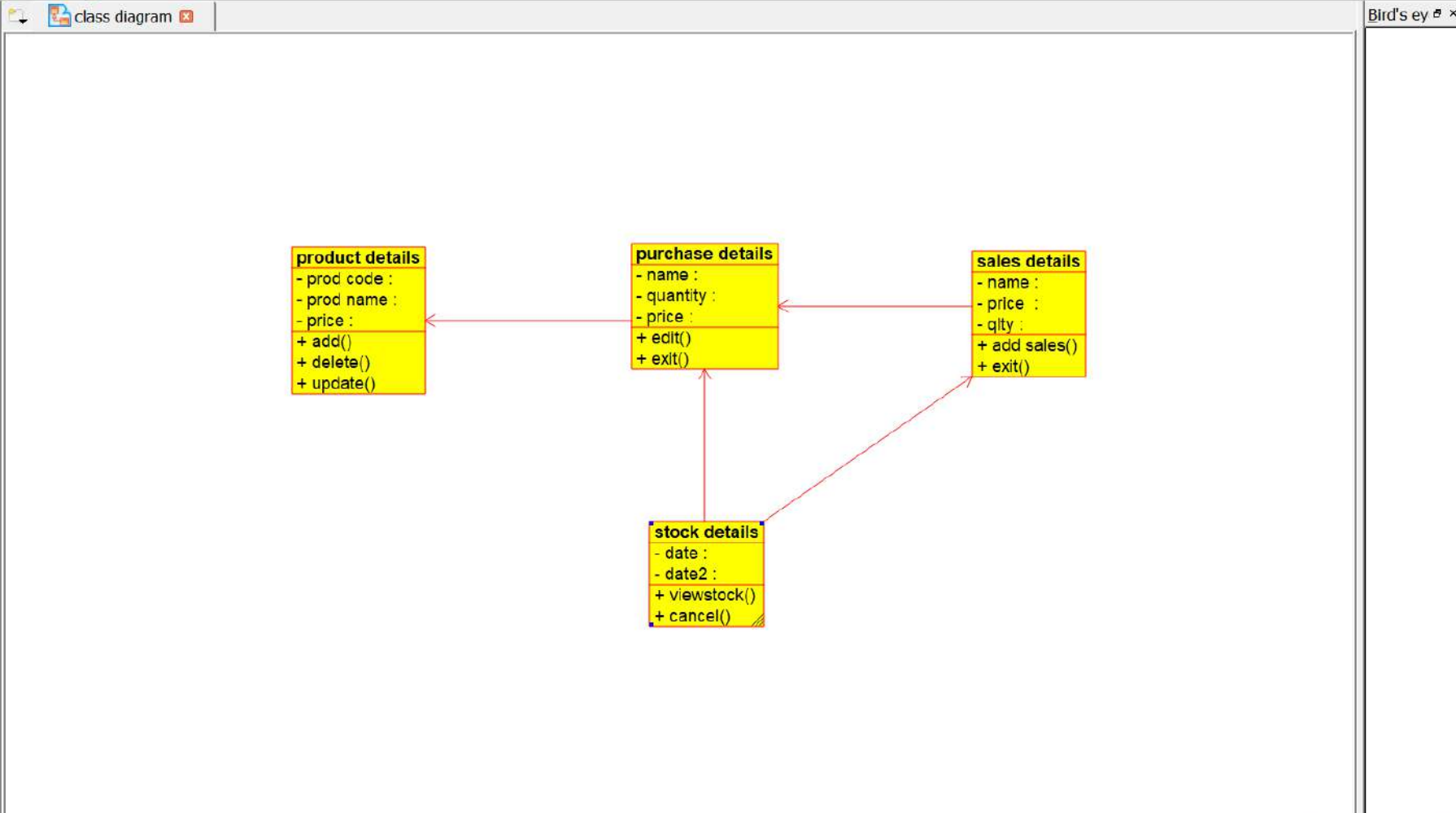
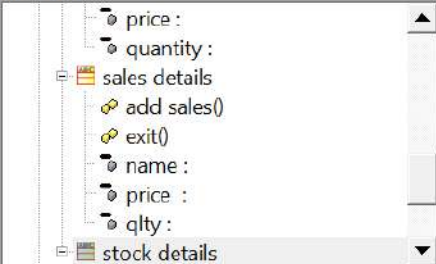


Name	Usage
folder	1
datatype	11



Stereotypes

Name	Usage
folder	1
datatype	11

Tree View

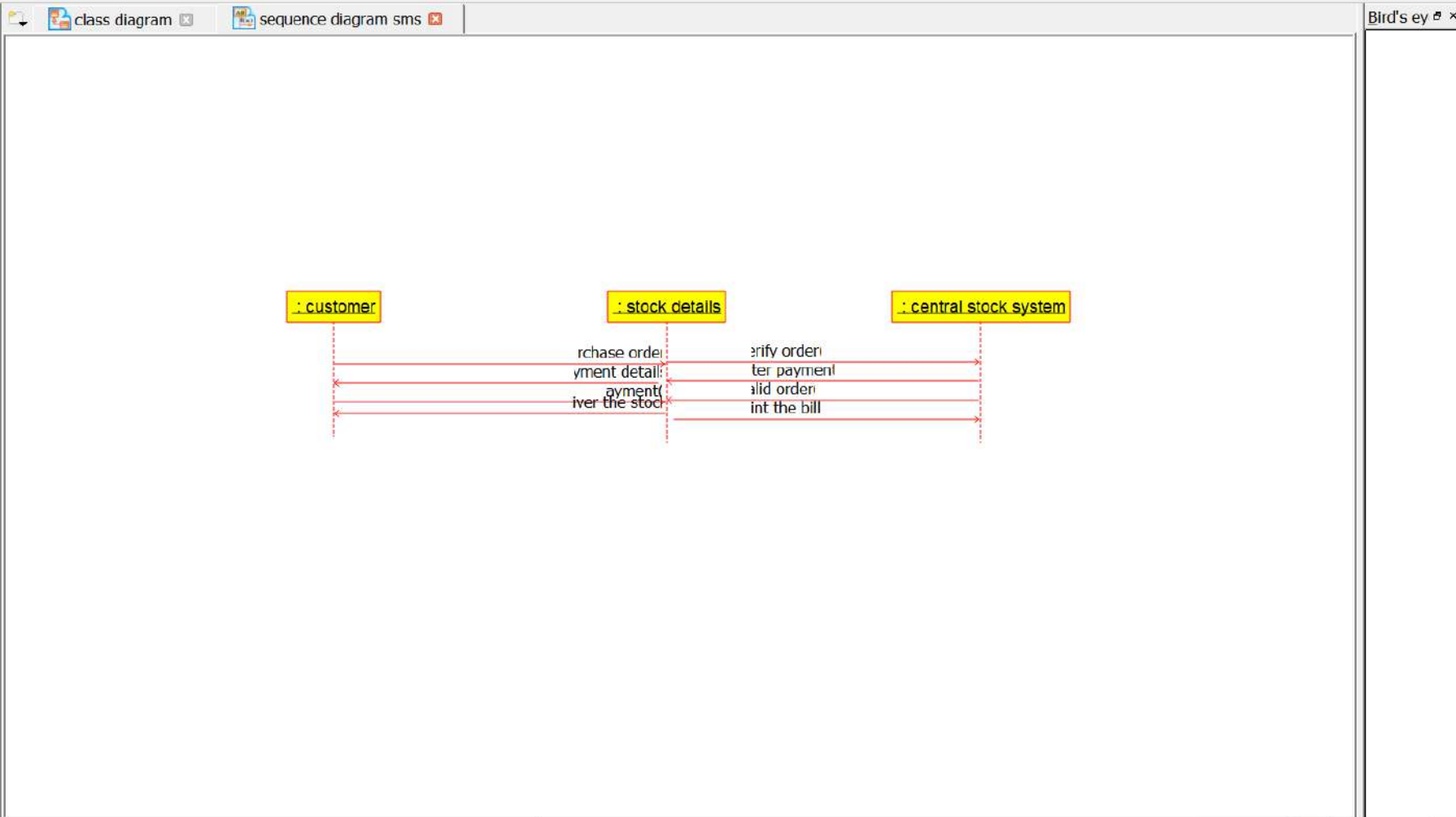
- print the bill()
- verify order()
- class diagram
- customer
 - deliver the stock()
 - payment details()
- Datatypes
- sequence diagram sms
- stock details

Documentation

sequence diagram sms

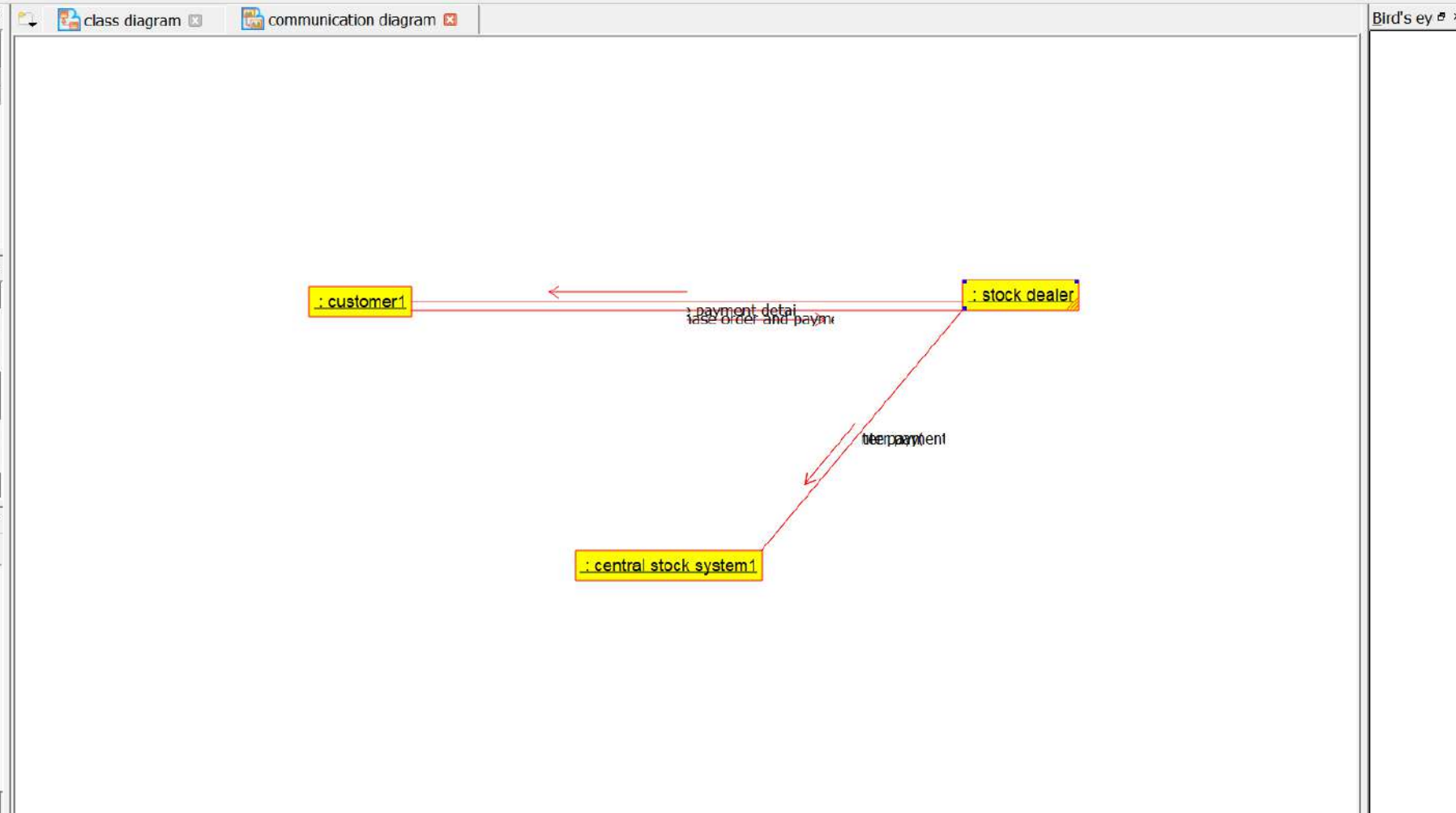
Documentation Command history Log

sequence diagram

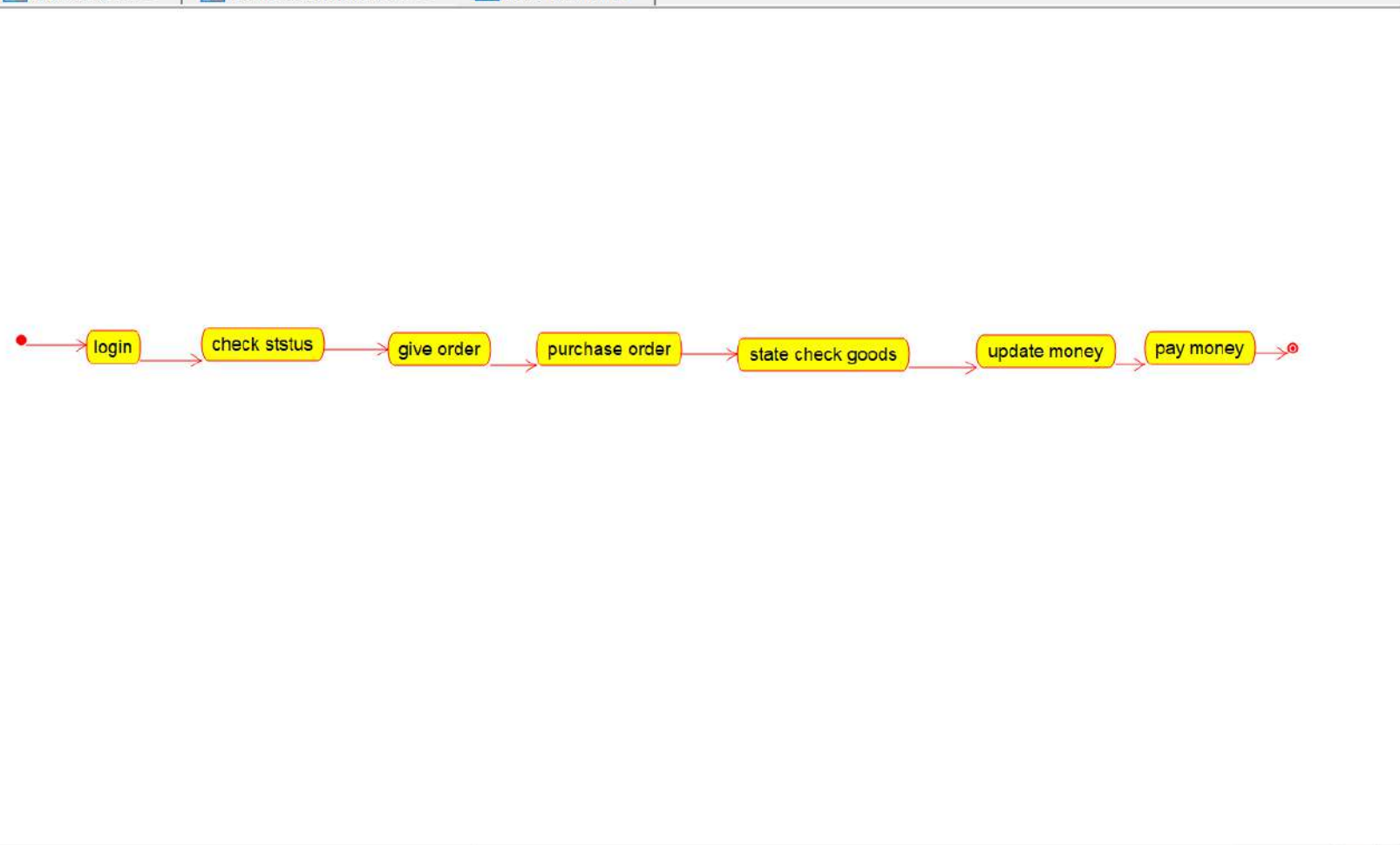
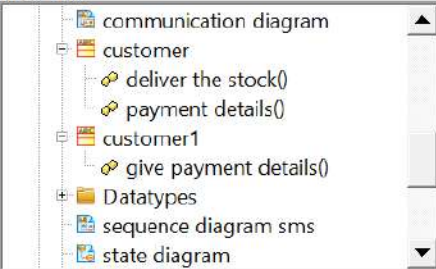


Name	Usage
folder	1
datatype	11

- new_operation()
- verify_order()
- class diagram
- communication diagram
- customer
 - deliver the stock()
 - payment details()
- customer1
 - give payment details()



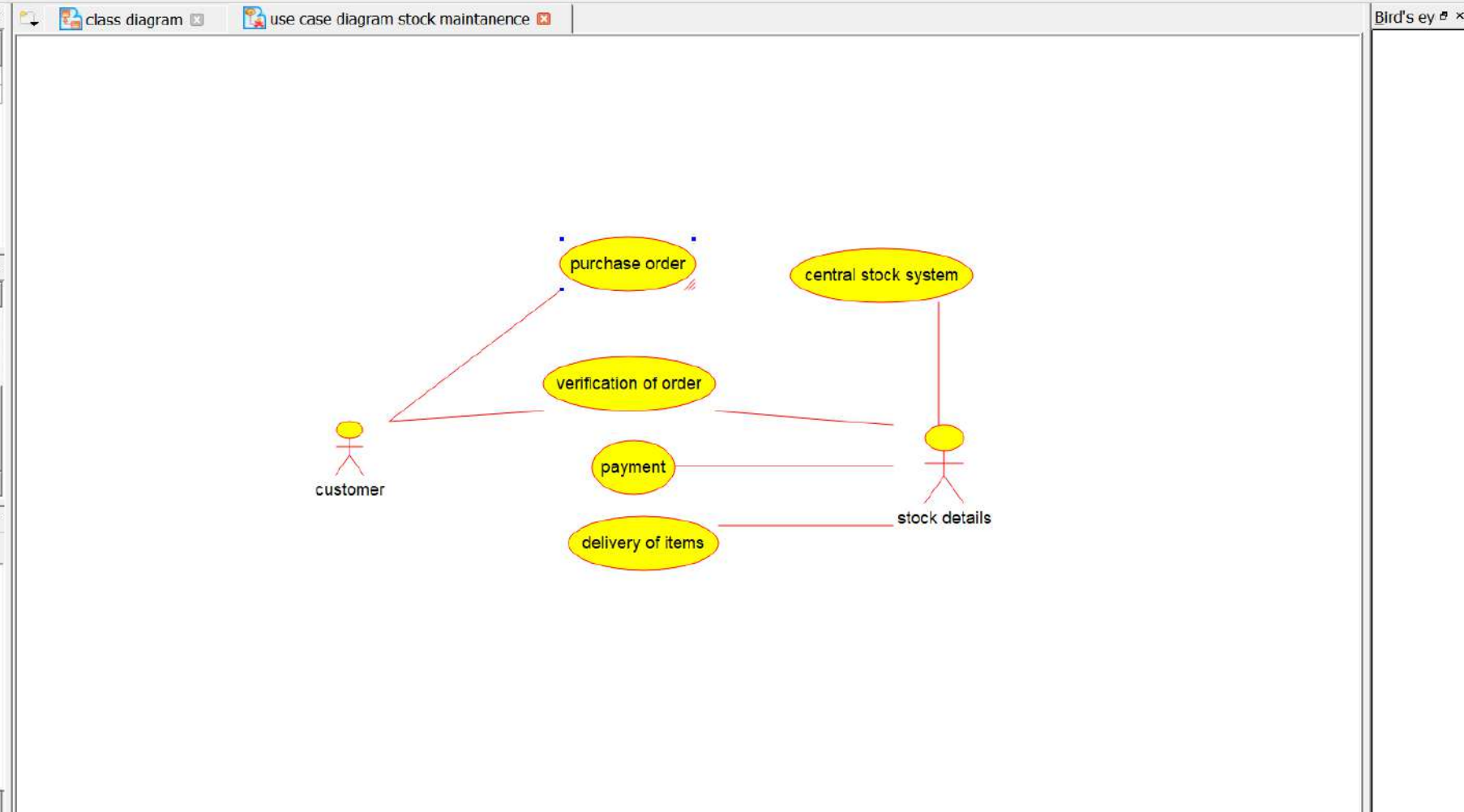
Name	Usage
folder	1
datatype	11



Name	Usage
folder	1
datatype	11

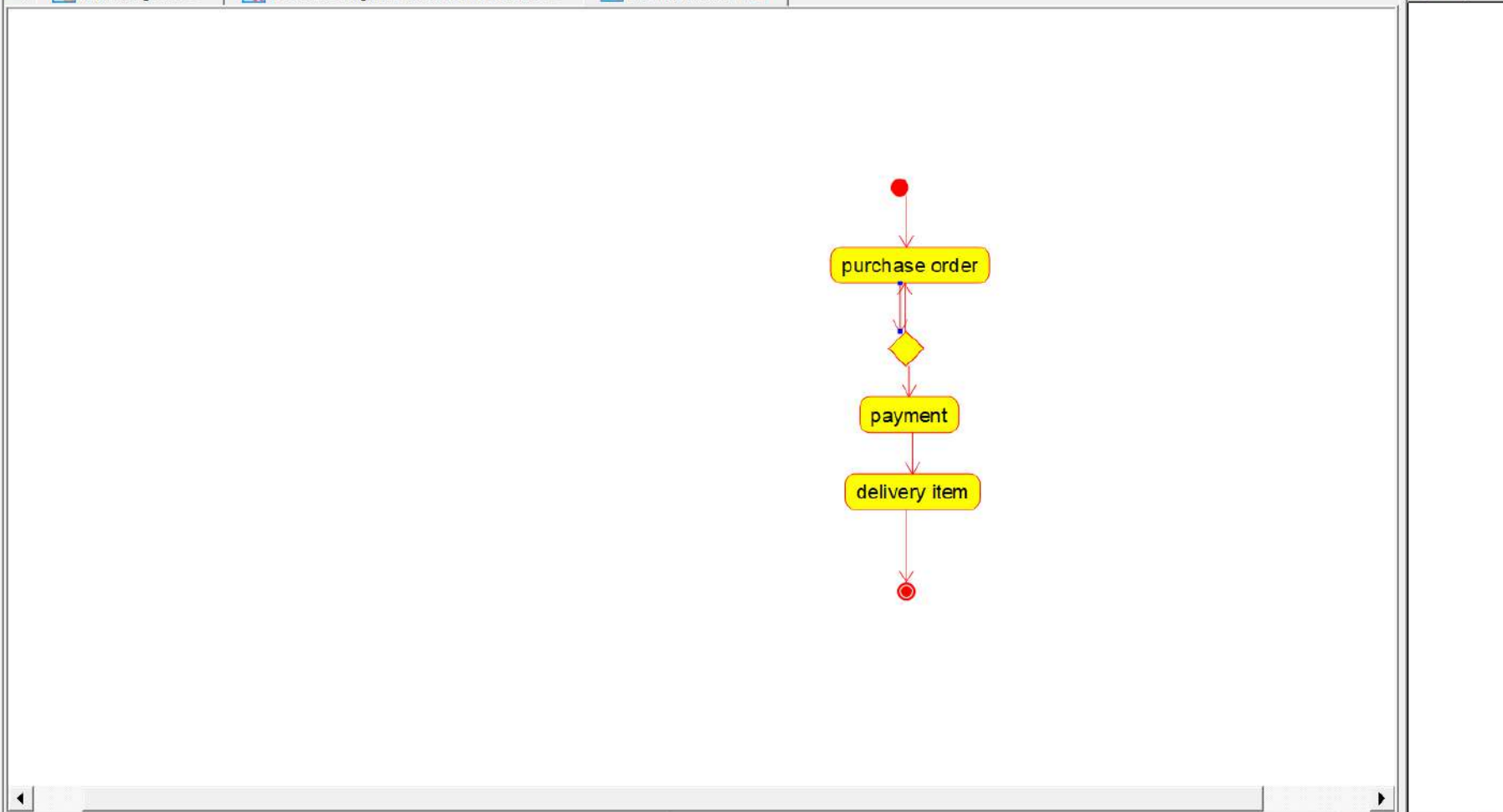
- Use Case View
 - central stock system
 - customer
 - delivery of items
 - payment
 - purchase order
 - stock details
 - use case diagram stock mainta...
 - verification of order

purchase order



Name	Usage
folder	1
datatype	11

- activity diagram
- class diagram
- Datatypes
- Use Case View
 - central stock system
 - customer
 - delivery of items
 - payment
 - purchase order



Stereotypes

Name	Usage
folder	1
datatype	11

Tree View

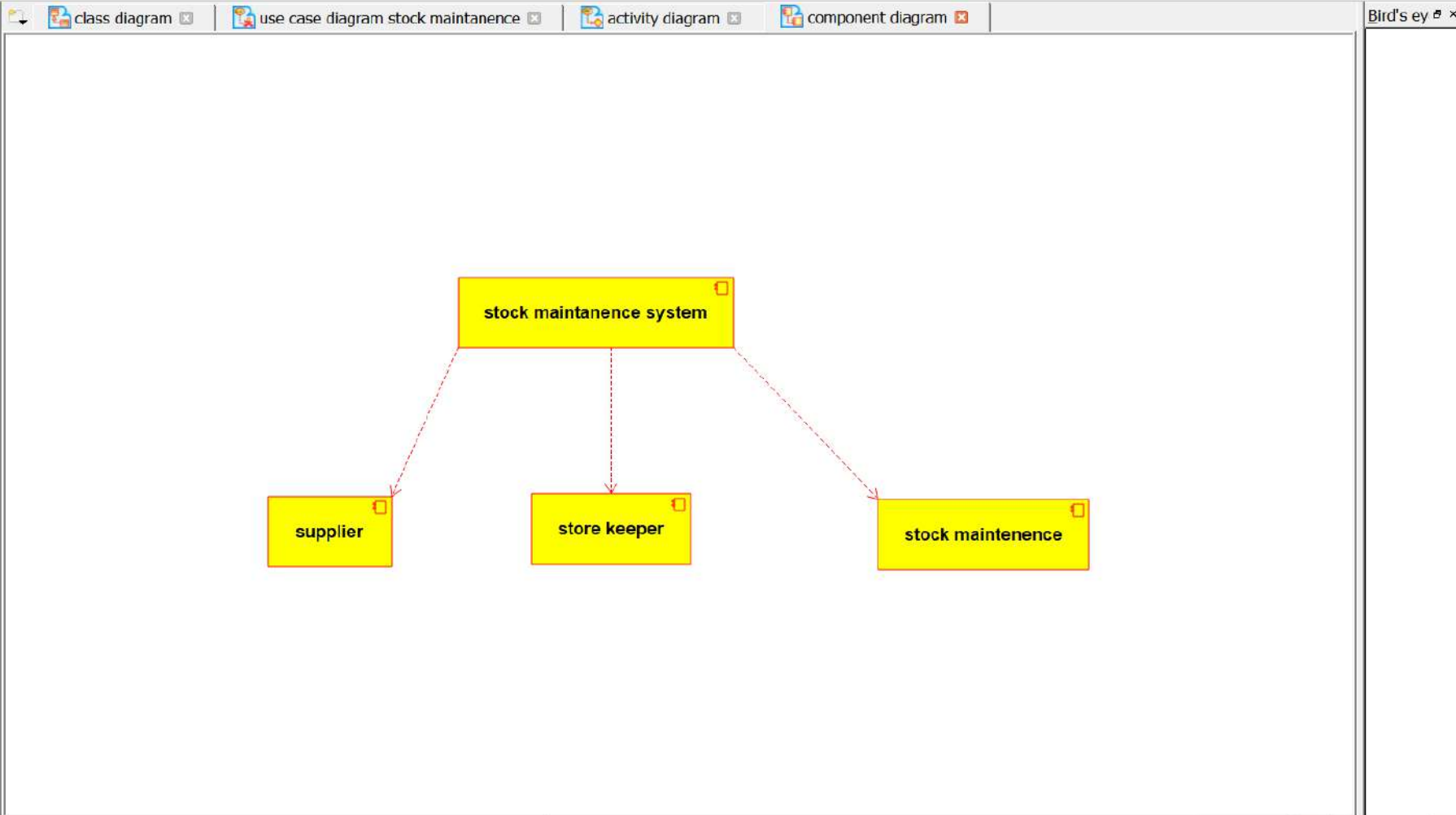
- component diagram
- stock maintenance system
- stock maintenance
- store keeper
- supplier
- Deployment View
- Entity Relationship Model
- Logical View
- activity diagram

Documentation

stock maintenance

Documentation Command history Log

component diagram



Stereotypes

Name	Usage
folder	1
datatype	11

Tree View

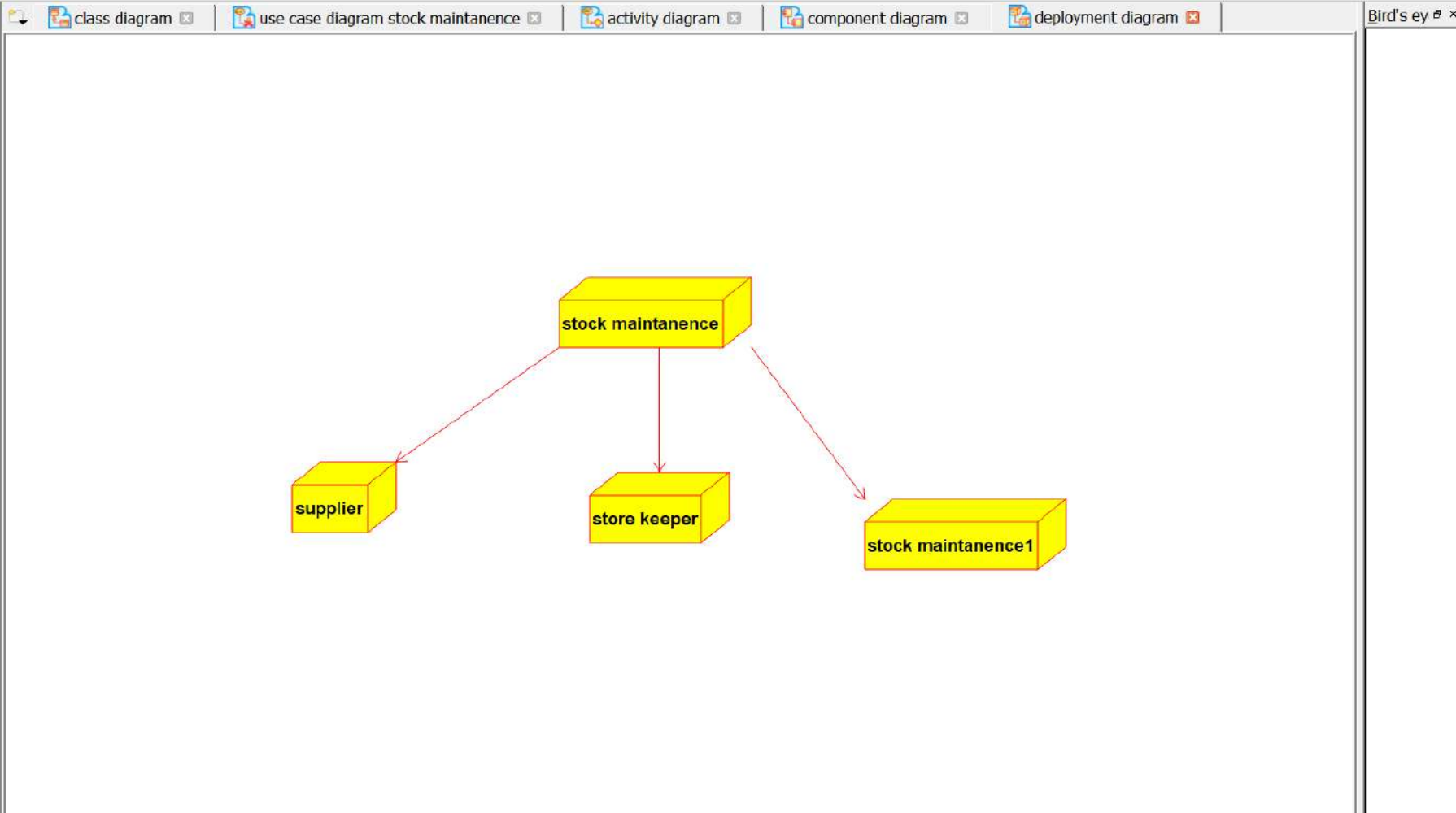
- component diagram
- stock maintenance system
- stock maintenance
- store keeper
- supplier
- Deployment View
- deployment diagram
- stock maintenance
- stock maintenance1

Documentation

stock maintenance1

Documentation Command history Log

deployment diagram



Stereotypes

Name	Usage
folder	1
datatype	11

Tree View

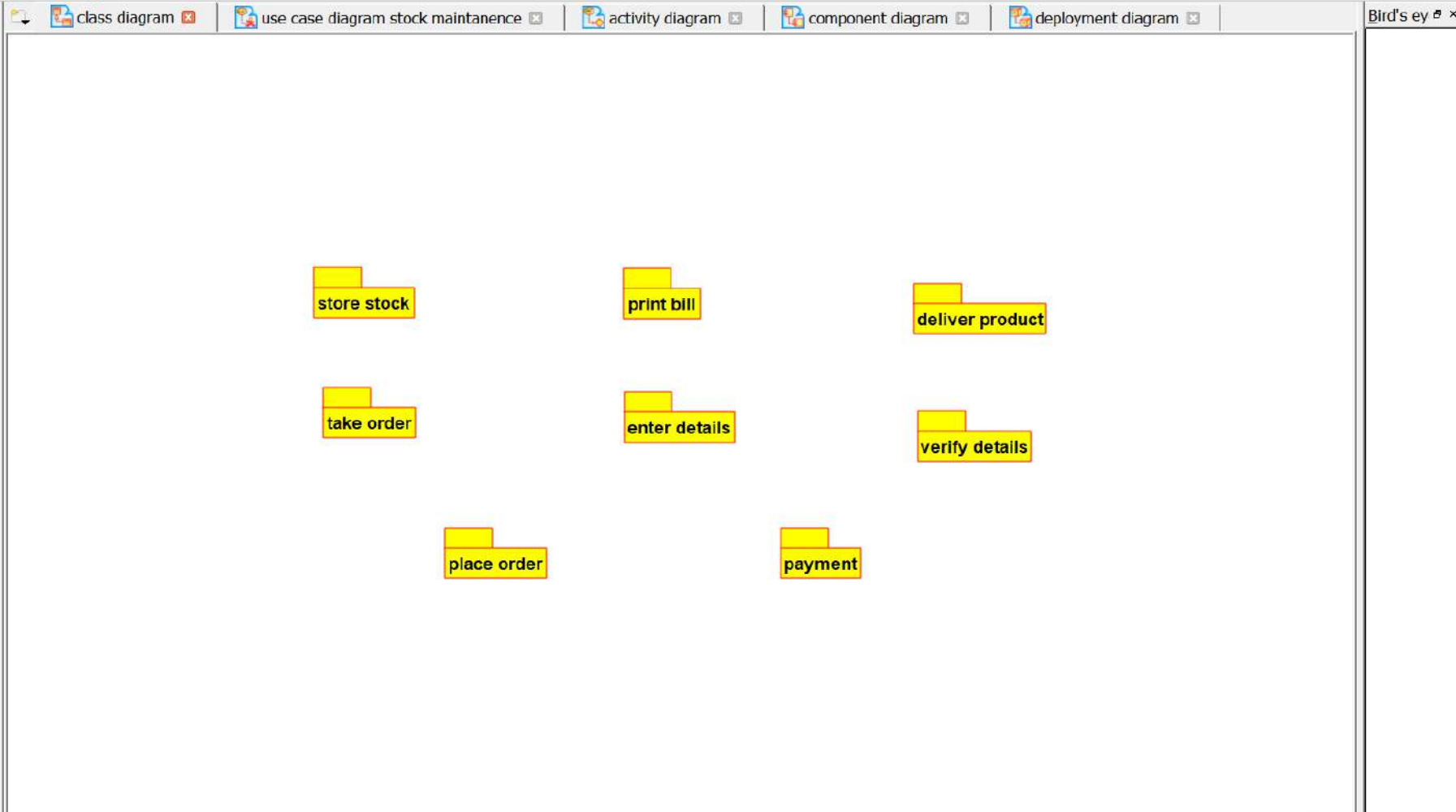
- activity diagram
- class diagram
- Datatypes
 - deliver product
 - enter details
 - payment
 - place order
 - print bill
 - store stock

Documentation

class diagram

Documentation Command history Log

class diagram



100% Fit 100%

C:\Users\Lenovo\uml-generated-code\product_details.h - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

IDM-GCC 4.9.2 64-bit Release

{globals}

Project Classes Debug

[*] product_details.h

```
1
2 #ifndef PRODUCT_DETAILS_H
3 #define PRODUCT_DETAILS_H
4 #include <string>
5 class product_details
6 {
7 public:
8     product_details ();
9     virtual ~product_details ();
10    void add ()
11    {
12    }
13    void delete ()
14    {
15    }
16    void update ()
17    {
18    }
19 protected:
20 public:
21 protected:
22 public:
23 protected:
24 private:
25     void prod_code;
26     void prod_name;
```

Compiler Resources Compile Log Debug Find Results

Line: 42 Col: 4 Sel: 0 Lines: 52 Length: 766 Insert Done parsing in 0.266 seconds

C:\Users\Lenovo\uml-generated-code\product_details.h - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

IDM-GCC 4.9.2 64-bit Release

{globals}

Project Classes Debug

[*] product_details.h

```
27     void price;
28 public:
29 private:
30 public:
31     void setProd_code (void new_var) {
32         prod_code = new_var;
33     }
34     void getProd_code () {
35         return prod_code;
36     }
37     void setProd_name (void new_var) {
38         prod_name = new_var;
39     }
40     void getProd_name () {
41         return prod_name;
42     }
43     void setPrice (void new_var) {
44         price = new_var;
45     }
46     void getPrice () {
47         return price;
48     }
49 private:
50     void initAttributes ();
51 };
52 #endif
```

Compiler Resources Compile Log Debug Find Results

Line: 42 Col: 4 Sel: 0 Lines: 52 Length: 766 Insert Done parsing in 0.266 seconds

Type here to search



32°C Mostly cloudy

ENG

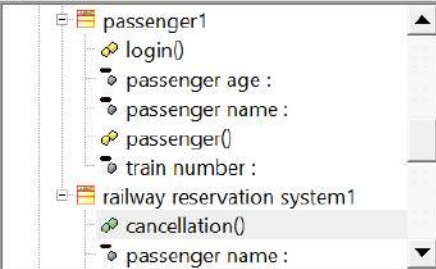
20:00
22-08-2023

1

Stereotypes

Name	Usage
folder	1
datatype	11

Tree View

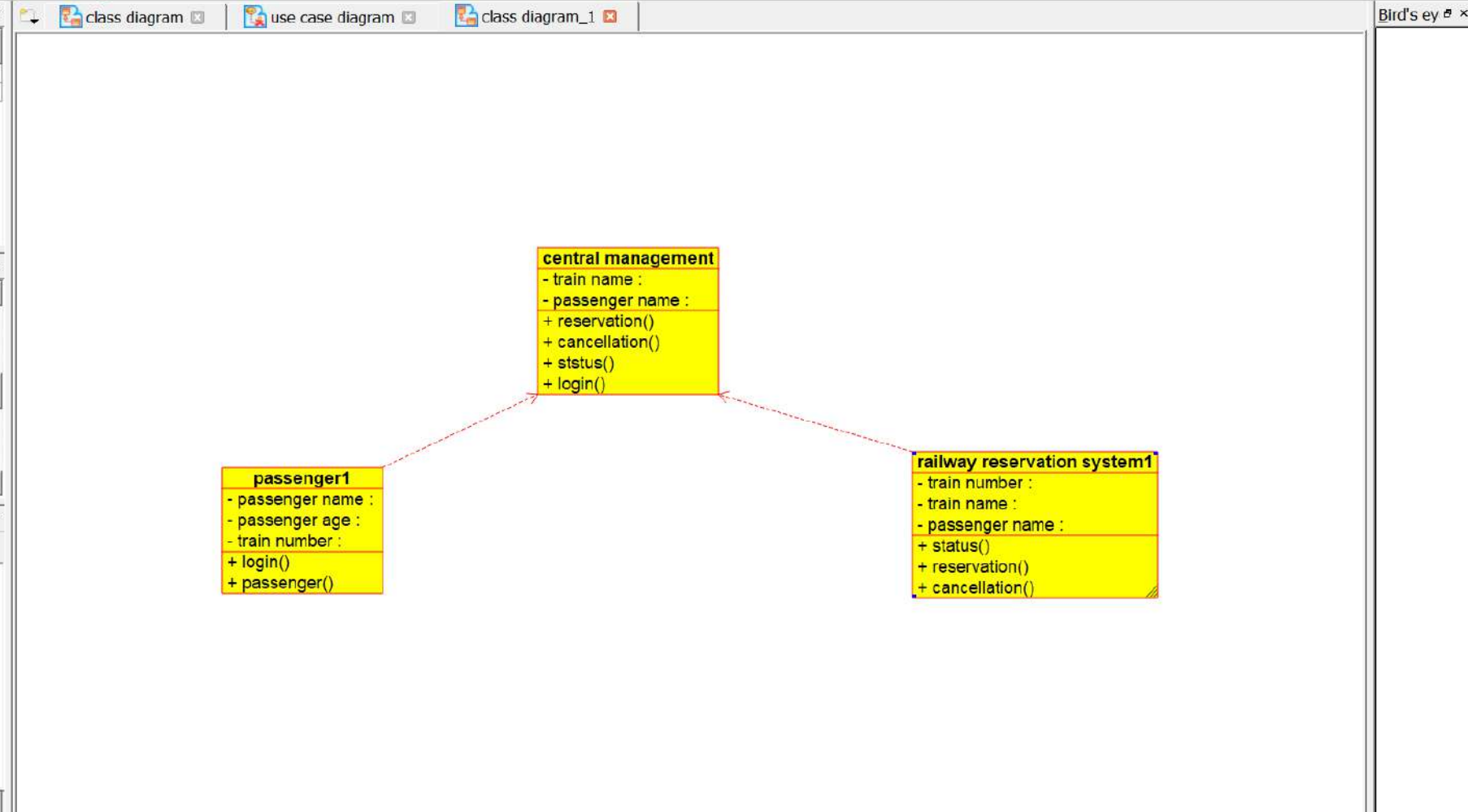


Documentation

central management

Documentation Command history Log

class diagram_1



C:\Users\Lenovo\uml-generated-code\railway_reservation_system1.h - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

IDM-GCC 4.9.2 64-bit Release

{globals}

Project Classes Debug [*] product_details.h [*] railway_reservation_system1.h

```
1
2 #ifndef RAILWAY_RESERVATION_SYSTEM1_H
3 #define RAILWAY_RESERVATION_SYSTEM1_H
4 #include <string>
5 class railway_reservation_system1
6 {
7 public:
8     virtual ~railway_reservation_system1 ();
9     void status ()
10 {
11 }
12     void reservation ()
13 {
14 }
15     void cancellation ()
16 {
17 }
18 protected:
19 public:
20 protected:
21 public:
22 protected:
23 private:
24     void train_number;
25     void train_name;
26     void passenger_name;
```

Compiler Resources Compile Log Debug Find Results

Line: 51 Col: 1 Sel: 0 Lines: 51 Length: 856 Insert Done parsing in 0.016 seconds

C:\Users\Lenovo\uml-generated-code\railway_reservation_system1.h - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

IDM-GCC 4.9.2 64-bit Release



{globals}

Project Classes Debug [*] product_details.h [*] railway_reservation_system1.h

```
26     void passenger_name;
27 public:
28 private:
29 public:
30     void setTrain_number (void new_var) {
31         train_number = new_var;
32     }
33     void getTrain_number () {
34         return train_number;
35     }
36     void setTrain_name (void new_var) {
37         train_name = new_var;
38     }
39     void getTrain_name () {
40         return train_n
41     }
42     void setPassenger_name (void new_var) {
43         passenger_name = new_var;
44     }
45     void getPassenger_name () {
46         return passenger_name;
47     }
48 private:
49     void initAttributes ();
50 };
51 #endif
```

Compiler Resources Compile Log Debug Find Results

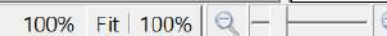
Line: 51 Col: 1 Sel: 0 Lines: 51 Length: 856 Insert Done parsing in 0.016 seconds

Bird's eye  

-

☐

activity diagram



Stereotypes

Name	Usage
folder	1
datatype	11

Tree View

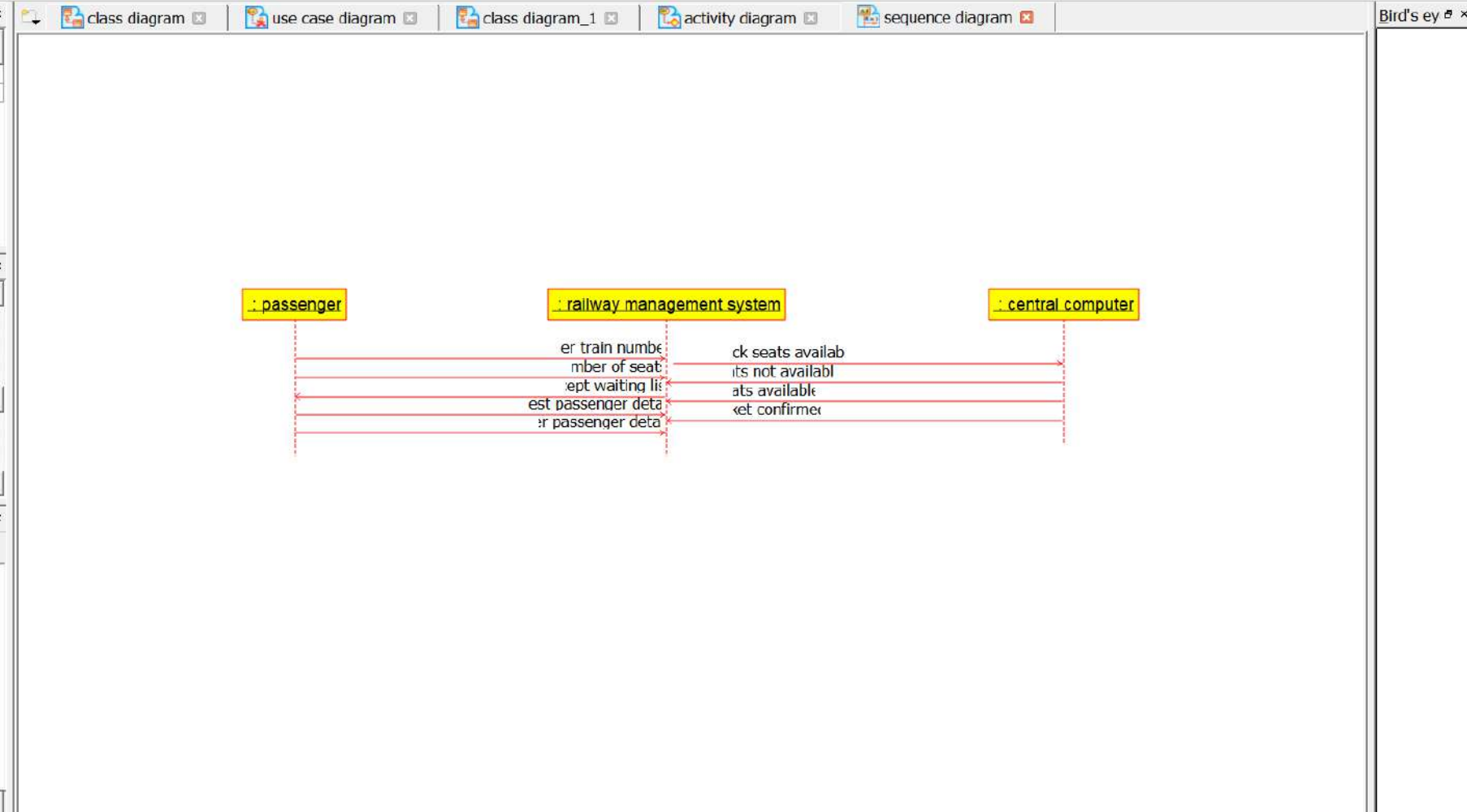
- train number :
- railway management system
 - enter passenger details()
 - enter train number()
 - number of seats()
 - request passenger details()
 - seats available()
 - seats not available()
 - ticket confirmed()

Documentation

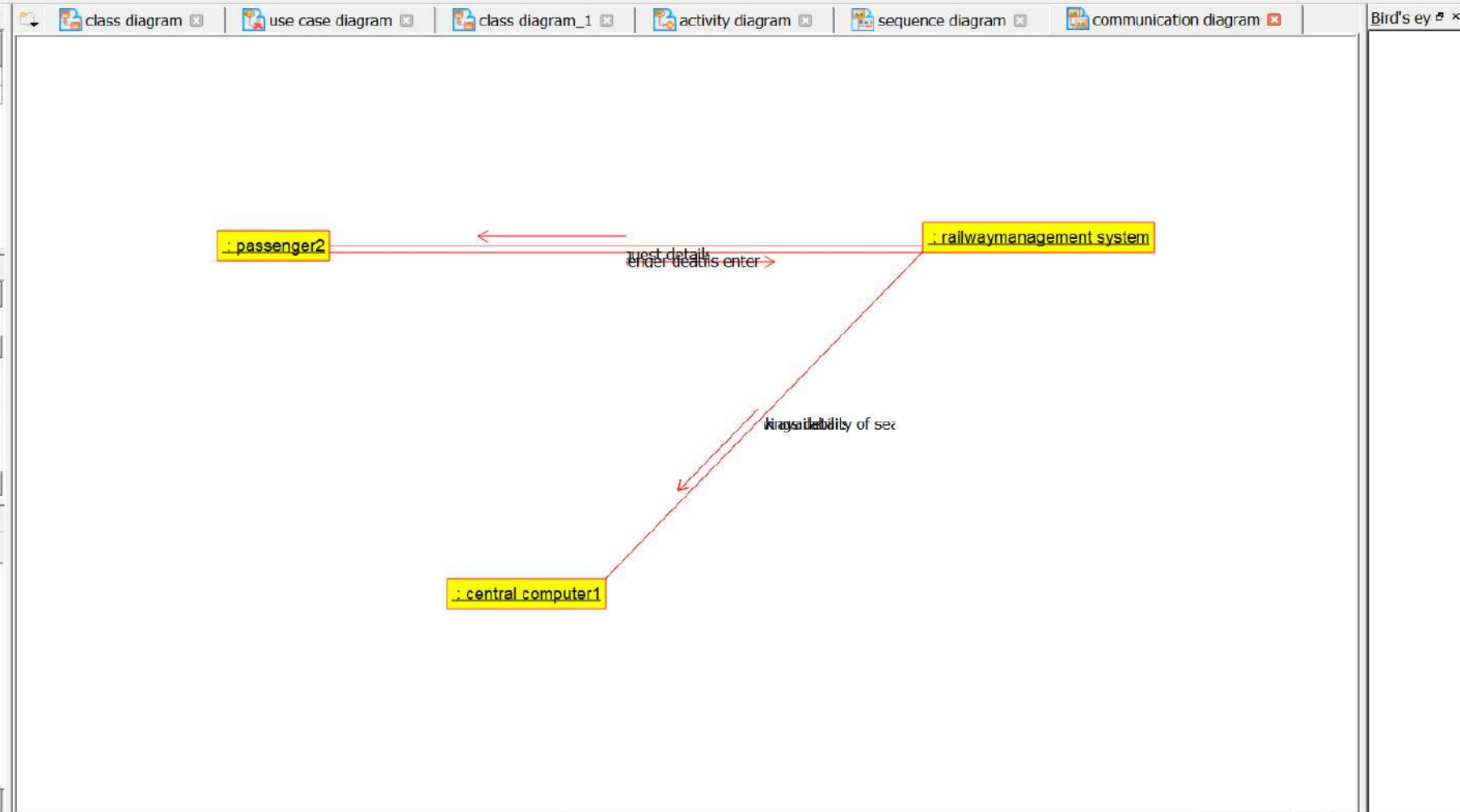
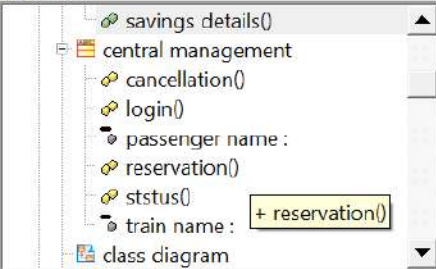
sequence diagram

Documentation Command history Log

sequence diagram



Name	Usage
folder	1
datatype	11



Stereotypes

Name	Usage
folder	1
datatype	11

Tree View

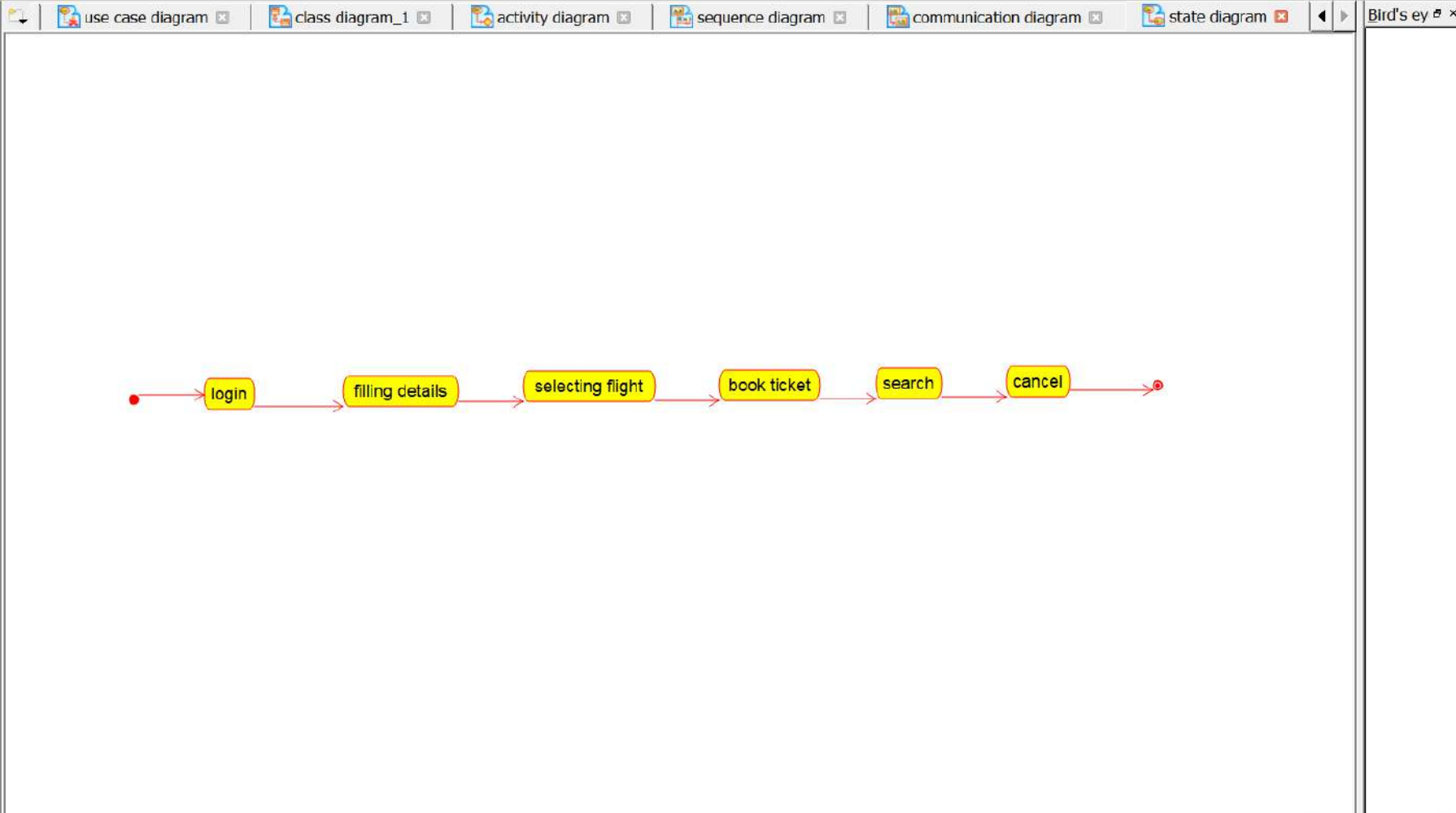
- passenger name :
 - reservation()
 - status()
- train name :
 - train number :
- railwaymanagement system
 - passenger deatils entered()
 - + passenger deatils entered()
 - state diagram

Documentation

State ☐

Documentation Command history Log

state diagram



Stereotypes

Name	Usage
folder	1
datatype	11

Tree View

- component diagram
- e tickrting
- filling details
- login
- selecting flight
- Deployment View
- Entity Relationship Model
- Logical View
- activity diagram

Documentation

selecting flight

Documentation Command history Log

component diagram

