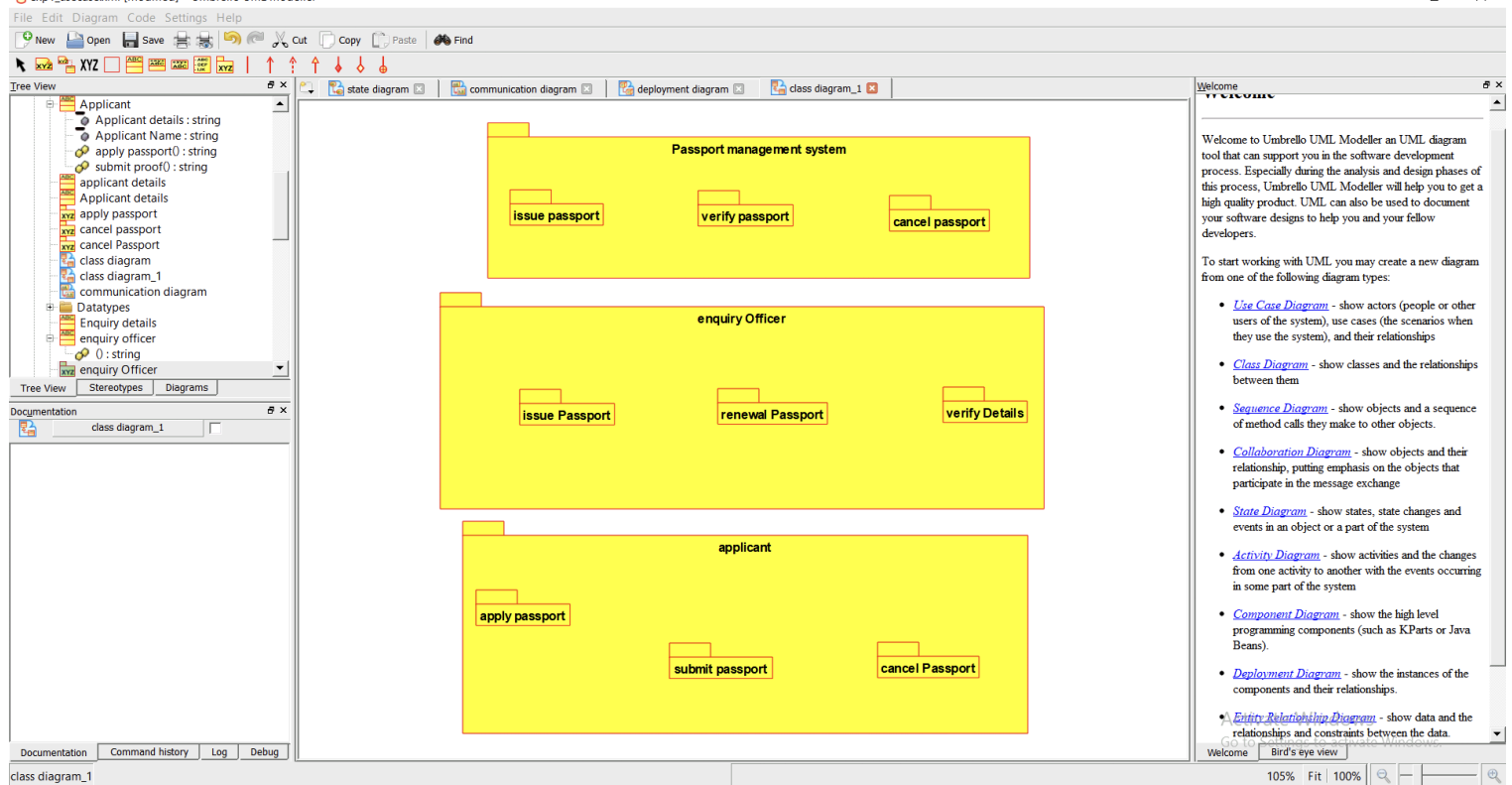


exp1_usecase.xmi [modified] - Umbrello UML Modeller



exp1_usecase.xmi [modified] – Umbrello UML Modeller

File Edit Diagram Code Settings Help

New Open Save Print Undo Redo Cut Copy Paste Find

XYZ

Tree View

Datatypes

Enquiry details

enquiry officer

0 : string

Enquiry officer

IssueOfPassport() : string

RenewalOfPassport() : string

verifydetails : string

Passport management

passport management system

Passport Management System

cancelPass() : bool

IssueOfPassport() : string

store proof : string

verify details : string

VerifyPassport() : string

sequence diagram

state diagram

Tree View Stereotypes Diagrams

Documentation

issue passport

Documentation Command history Log Debug

class diagram communication diagram state diagram activity diagram

```
graph LR; Start(( )) --> A[applicant details]; A --> B[applicant proof]; B --> C[verification of proof]; C --> D[issue passport]; D --> End((( )));
```

Welcome

Welcome to Umbrello UML Modeller an UML diagram tool that can support you in the software development process. Especially during the analysis and design phases of this process, Umbrello UML Modeller will help you to get a high quality product. UML can also be used to document your software designs to help you and your fellow developers.

To start working with UML, you may create a new diagram from one of the following diagram types:

- [Use Case Diagram](#) - show actors (people or other users of the system), use cases (the scenarios when they use the system), and their relationships
- [Class Diagram](#) - show classes and the relationships between them
- [Sequence Diagram](#) - show objects and a sequence of method calls they make to other objects.
- [Collaboration Diagram](#) - show objects and their relationship, putting emphasis on the objects that participate in the message exchange
- [State Diagram](#) - show states, state changes and events in an object or a part of the system
- [Activity Diagram](#) - show activities and the changes from one activity to another with the events occurring in some part of the system
- [Component Diagram](#) - show the high level programming components (such as KParts or Java Beans).
- [Deployment Diagram](#) - show the instances of the components and their relationships.
- [Entity-Relationship Diagram](#) - show data and the relationships between them.

Activate Windows

Welcome Bird's eye view

125% Fit 100%

exp1_usecase.xmi [modified] – Umbrello UML Modeller

File Edit Diagram Code Settings Help

New Open Save Print Undo Cut Copy Paste Find

XYZ

class diagramstate diagrampassport_auto_syssequence diagramcommunication diagramcomponent diagram

Tree View

Settings

Views

Component View

Applicant details

Applicant proof

component diagram

issue of passport

passport management system

verification of proof

Deployment View

Entity Relationship Model

Logical View

activity diagram

Applicant

Applicant details : string

Applicant Name : string

apply passport() : string

submit proof() : string

Tree View

Stereotypes

Diagrams

Documentation

passport management system

passport management system

Applicant details

Applicant proof

verification of proof

issue of passport

Welcome

Welcome

Welcome to Umbrello UML Modeller an UML diagram tool that can support you in the software development process. Especially during the analysis and design phases of this process, Umbrello UML Modeller will help you to get a high quality product. UML can also be used to document your software designs to help you and your fellow developers.

To start working with UML, you may create a new diagram from one of the following diagram types:

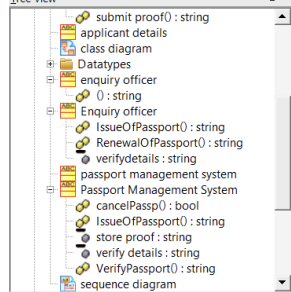
- [Use Case Diagram](#) - show actors (people or other users of the system), use cases (the scenarios when they use the system), and their relationships between them
- [Class Diagram](#) - show classes and the relationships between them
- [Sequence Diagram](#) - show objects and a sequence of method calls they make to other objects.
- [Collaboration Diagram](#) - show objects and their relationship, putting emphasis on the objects that participate in the message exchange
- [State Diagram](#) - show states, state changes and events in an object or a part of the system
- [Activity Diagram](#) - show activities and the changes from one activity to another with the events occurring in some part of the system
- [Component Diagram](#) - show the high level programming components (such as KParts or Java Beans).
- [Deployment Diagram](#) - show the instances of the components and their relationships.
- [Entity Relationship Diagram](#) - show data and the

Activate Windows

Welcome

Bird's eye view

component diagram100%Fit100%



applicant detail

enquiry office

passport management system

1: applicant details

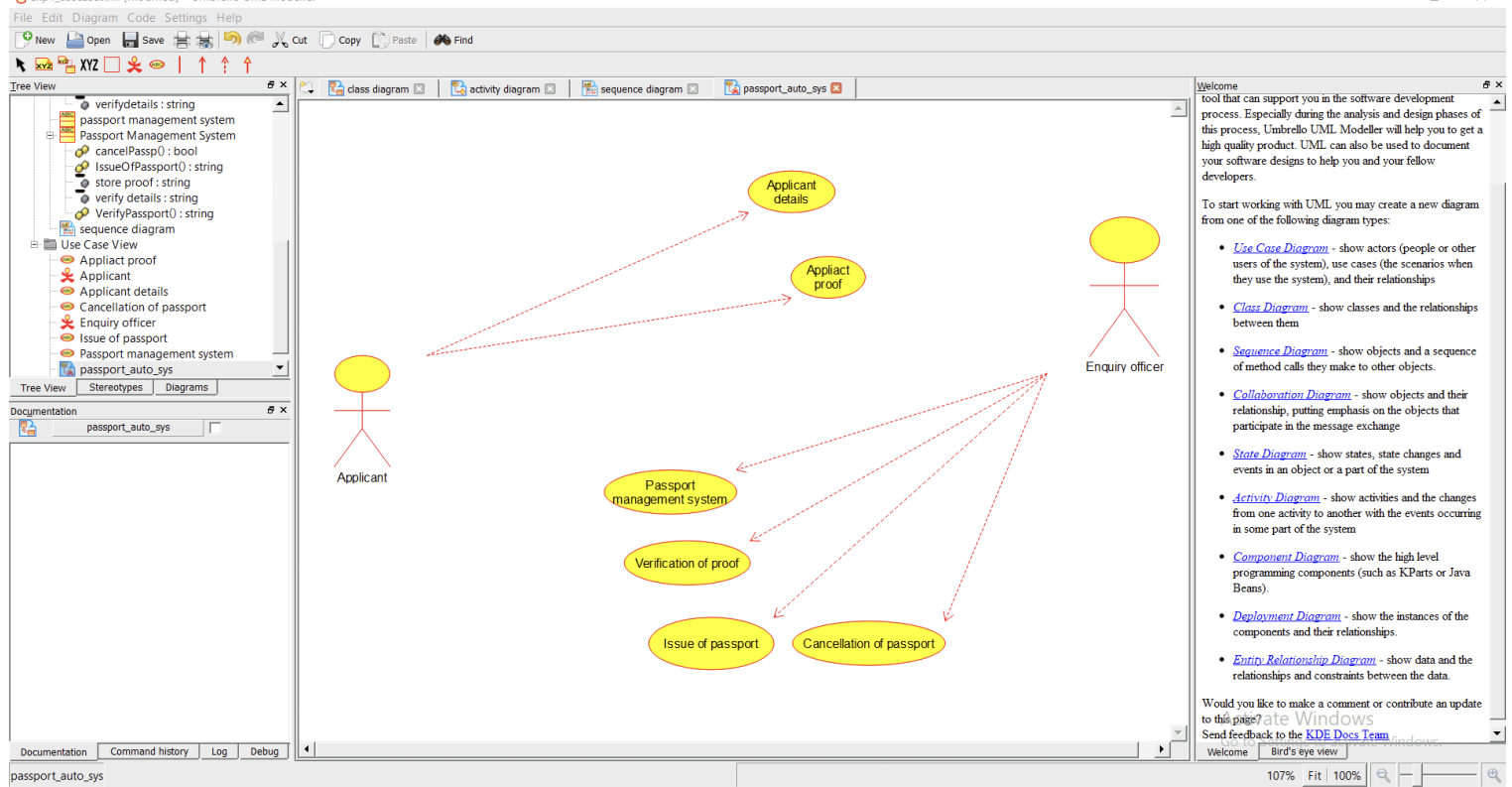
2: request for proofs

3: submission of proofs

4: check for valid proof

5: valid proof

6: issue passport



File Edit Diagram Code Settings Help

New Open Save Print Undo Cut Copy Paste Find

XYZ A B

Tree View

- communication diagram
- Datatypes
- Enquiry details
 - enquiry officer
 - 0 : string
 - Enquiry officer
 - IssueOfPassport0 : string
 - RenewalOfPassport0 : string
 - verifydetails : string
 - Passport management
 - passport management system
 - Passport Management System
 - cancelPass0 : bool
 - IssueOfPassport0 : string
 - store proof : string
 - verify details : string
 - VerifyPassport0 : string
 - sequence diagram

Tree View Stereotypes Diagrams

Documentation

communication diagram

Documentation Command history Log Debug

class diagram activity diagram sequence diagram passport_auto_sys communication diagram

1: Applicant details
3: Submission of proofs
2: request of passport
6: issue passport
4: check for valid proof
5: valid proof

..Applicant details ..Enquiry details ..Passport management

m8 m90

Welcome

tool that can support you in the software development process. Especially during the analysis and design phases of this process, Umbrello UML Modeller will help you to get a high quality product. UML can also be used to document your software designs to help you and your fellow developers.

To start working with UML you may create a new diagram from one of the following diagram types:

- [Use Case Diagram](#) - show actors (people or other users of the system), use cases (the scenarios when they use the system), and their relationships
- [Class Diagram](#) - show classes and the relationships between them
- [Sequence Diagram](#) - show objects and a sequence of method calls they make to other objects.
- [Collaboration Diagram](#) - show objects and their relationship, putting emphasis on the objects that participate in the message exchange
- [State Diagram](#) - show states, state changes and events in an object or a part of the system
- [Activity Diagram](#) - show activities and the changes from one activity to another with the events occurring in some part of the system
- [Component Diagram](#) - show the high level programming components (such as KParts or Java Beans).
- [Deployment Diagram](#) - show the instances of the components and their relationships.
- [Entity Relationship Diagram](#) - show data and the relationships and constraints between the data.

Would you like to make a comment or contribute an update to this page?

Send feedback to the [KDE Docs Team](#)

Welcome Bird's eye view

100% Fit 100%

exp1_usecase.xmi [modified] – Umbrello UML Modeller

FileEditDiagramCodeSettingsHelp

NewOpenSavePrintUndoRedoCutCopyPasteFind

XYZ

class diagramactivity diagramsequence diagrampassport_auto_sys

Tree View

class diagram

Datatypes

enquiry officer

Enquiry officer

IssueOfPassport() : string

RenewalOfPassport() : string

verifydetails : string

passport management system

Passport Management System

cancelPass() : bool

IssueOfPassport() : string

store proof : string

verify details : string

VerifyPassport() : string

sequence diagram

Use Case View

Appliact proof

Tree ViewStereotypesDiagrams

Documentation

class diagram

DocumentationCommand historyLogDebug

class diagram

Enquiry officer

- verifydetails : string

+ IssueOfPassport() : string

+ RenewalOfPassport() : string

Passport Management System

- verify details : string

- store proof : string

+ IssueOfPassport() : string

+ VerifyPassport() : string

+ cancelPass() : bool

Applicant

- Applicant Name : string

- Applicant details : string

+ apply passport() : string

+ submit proof() : string

Welcome

tool that can support you in the software development process. Especially during the analysis and design phases of this process, Umbrello UML Modeller will help you to get a high quality product. UML can also be used to document your software designs to help you and your fellow developers.

To start working with UML you may create a new diagram from one of the following diagram types:

- [Use Case Diagram](#) - show actors (people or other users of the system), use cases (the scenarios when they use the system), and their relationships
- [Class Diagram](#) - show classes and the relationships between them
- [Sequence Diagram](#) - show objects and a sequence of method calls they make to other objects.
- [Collaboration Diagram](#) - show objects and their relationship, putting emphasis on the objects that participate in the message exchange
- [State Diagram](#) - show states, state changes and events in an object or a part of the system
- [Activity Diagram](#) - show activities and the changes from one activity to another with the events occurring in some part of the system
- [Component Diagram](#) - show the high level programming components (such as KParts or Java Beans).
- [Deployment Diagram](#) - show the instances of the components and their relationships.
- [Entity Relationship Diagram](#) - show data and the relationships and constraints between the data.

Would you like to make a comment or contribute an update to this page?[Create Windows](#)

Send feedback to the [KDE Docs Team](#)

WelcomeBird's eye view

class diagram93%Fit100%

exp1_usecase.xmi [modified] - Umbrello UML Modeller

File Edit Diagram Code Settings Help

New Open Save Print Cut Copy Paste Find

XYZ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Tree View

- applicant details
- applicant proof
- deployment diagram
- issue of passport
- passport management system
- verification of proof
- Entity Relationship Model
- Logical View
- activity diagram
- Applicant
 - Applicant details : string
 - Applicant Name : string
 - apply passport() : string
 - submit proof() : string
- applicant details
- applicant details
- class diagram
- communication diagram

Tree View Stereotypes Diagrams

Documentation

deployment diagram

passport_auto_sys state diagram communication diagram deployment diagram

Welcome

Welcome

Welcome to Umbrello UML Modeller an UML diagram tool that can support you in the software development process. Especially during the analysis and design phases of this process, Umbrello UML Modeller will help you to get a high quality product. UML can also be used to document your software designs to help you and your fellow developers.

To start working with UML, you may create a new diagram from one of the following diagram types:

- [Use Case Diagram](#) - show actors (people or other users of the system), use cases (the scenarios when they use the system), and their relationships between them
- [Class Diagram](#) - show classes and the relationships between them
- [Sequence Diagram](#) - show objects and a sequence of method calls they make to other objects.
- [Collaboration Diagram](#) - show objects and their relationship, putting emphasis on the objects that participate in the message exchange
- [State Diagram](#) - show states, state changes and events in an object or a part of the system
- [Activity Diagram](#) - show activities and the changes from one activity to another with the events occurring in some part of the system
- [Component Diagram](#) - show the high level programming components (such as KParts or Java Beans).
- [Deployment Diagram](#) - show the instances of the components and their relationships.
- [Entity Relationship Diagram](#) - show data and the relationships between them.

Activate Windows

100% Fit 100%

Press Ctrl with left mouse click to delete a point

```
graph TD; A[exam registration system] -.-> B[registration]; A -.-> C[verification]; A -.-> D[search];
```

exp1_usecase.xmi [modified] – Umbrello UML Modeller

File Edit Diagram Code Settings Help

New Open Save Print Undo Cut Copy Paste Find

Tree View

applicant details

applicant proof

deployment diagram

issue of passport

passport management system

verification of proof

Entity Relationship Model

Logical View

activity diagram

Applicant

Applicant details : string

Applicant Name : string

apply passport() : string

submit proof() : string

applicant details

applicant details

class diagram

communication diagram

Tree View Stereotypes Diagrams

Documentation

deployment diagram

Documentation Command history Log Debug

passport_auto_sys state diagram communication diagram deployment diagram

```
graph LR; Start(( )) --> login[login]; login --> fee[fee payment and processing]; fee --> filling[filling form]; filling --> verification[form verification]; verification --> storing[storing to database]; storing --> registration[registration acknowledgement]; registration --> End((( )))
```

Welcome

Welcome

Welcome to Umbrello UML Modeller an UML diagram tool that can support you in the software development process. Especially during the analysis and design phases of this process, Umbrello UML Modeller will help you to get a high quality product. UML can also be used to document your software designs to help you and your fellow developers.

To start working with UML, you may create a new diagram from one of the following diagram types:

- [Use Case Diagram](#) - show actors (people or other users of the system), use cases (the scenarios when they use the system), and their relationships
- [Class Diagram](#) - show classes and the relationships between them
- [Sequence Diagram](#) - show objects and a sequence of method calls they make to other objects.
- [Collaboration Diagram](#) - show objects and their relationship, putting emphasis on the objects that participate in the message exchange
- [State Diagram](#) - show states, state changes and events in an object or a part of the system
- [Activity Diagram](#) - show activities and the changes from one activity to another with the events occurring in some part of the system
- [Component Diagram](#) - show the high level programming components (such as KParts or Java Beans).
- [Deployment Diagram](#) - show the instances of the components and their relationships.
- [Entity Relationship Diagram](#) - show data and the

Welcome Bird's eye view

100% Fit 100%

Press Ctrl with left mouse click to delete a point

File Edit Diagram Code Settings Help

New Open Save Print Undo Cut Copy Paste Find

XYZ A.B C.D E.F G.H I.J K.L M.N O.P Q.R S.T U.V W.X Y.Z

Tree View

- applicant details
- applicant proof
- deployment diagram
- issue of passport
- passport management system
- verification of proof
- Entity Relationship Model
- Logical View
- activity diagram
- Applicant
 - Applicant details : string
 - Applicant Name : string
 - apply passport() : string
 - submit proof() : string
- applicant details
- applicant details
- class diagram
- communication diagram

Tree View Stereotypes Diagrams

Documentation

deployment diagram

passport_auto_sys state diagram communication diagram deployment diagram

Welcome

Welcome

Welcome to Umbrello UML Modeller an UML diagram tool that can support you in the software development process. Especially during the analysis and design phases of this process, Umbrello UML Modeller will help you to get a high quality product. UML can also be used to document your software designs to help you and your fellow developers.

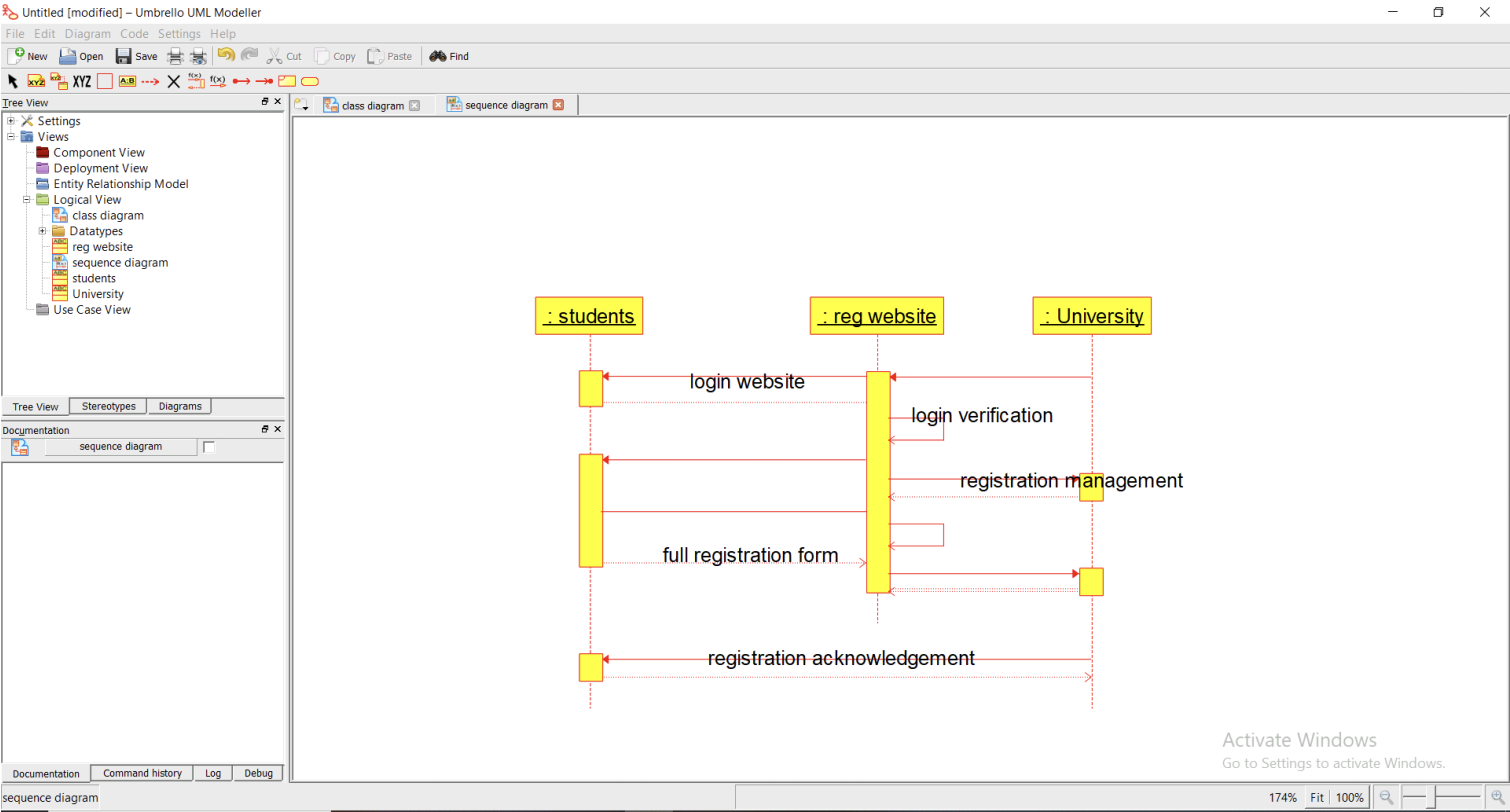
To start working with UML, you may create a new diagram from one of the following diagram types:

- [Use Case Diagram](#) - show actors (people or other users of the system), use cases (the scenarios when they use the system), and their relationships
- [Class Diagram](#) - show classes and the relationships between them
- [Sequence Diagram](#) - show objects and a sequence of method calls they make to other objects.
- [Collaboration Diagram](#) - show objects and their relationship, putting emphasis on the objects that participate in the message exchange
- [State Diagram](#) - show states, state changes and events in an object or a part of the system
- [Activity Diagram](#) - show activities and the changes from one activity to another with the events occurring in some part of the system
- [Component Diagram](#) - show the high level programming components (such as KParts or Java Beans).
- [Deployment Diagram](#) - show the instances of the components and their relationships.
- [Entity Relationship Diagram](#) - show data and the

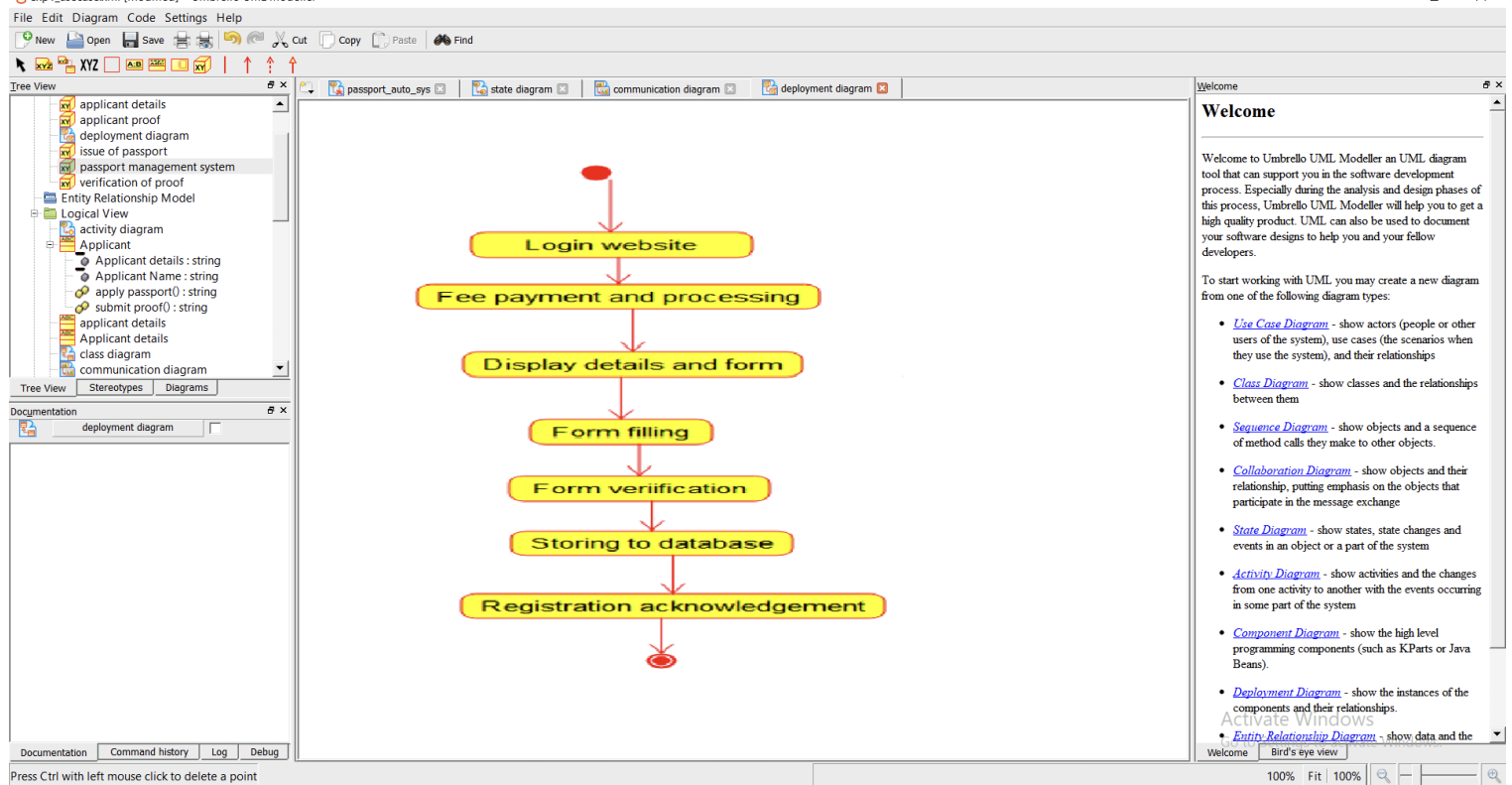
Activate Windows

100% Fit 100%

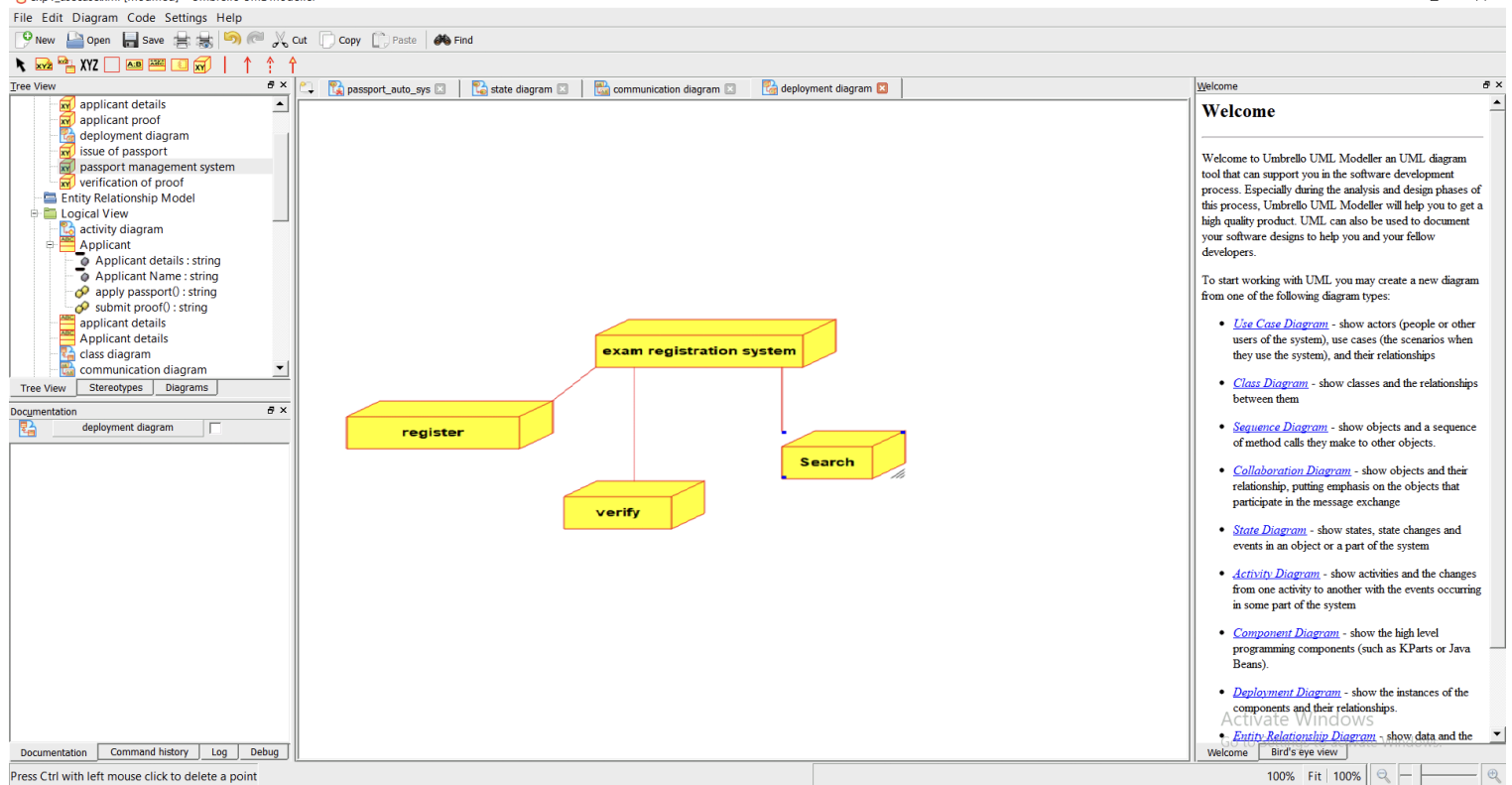
Press Ctrl with left mouse click to delete a point

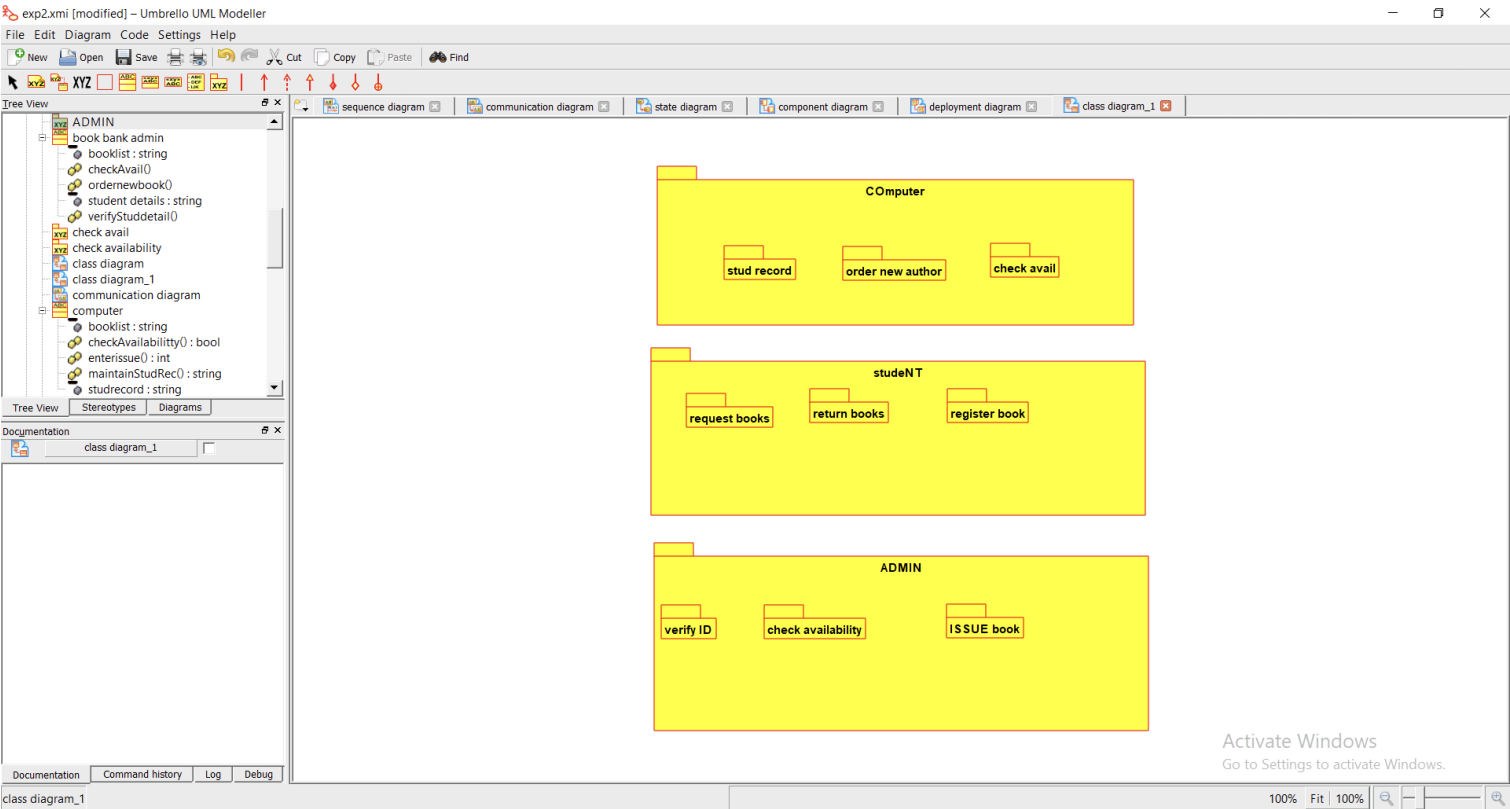


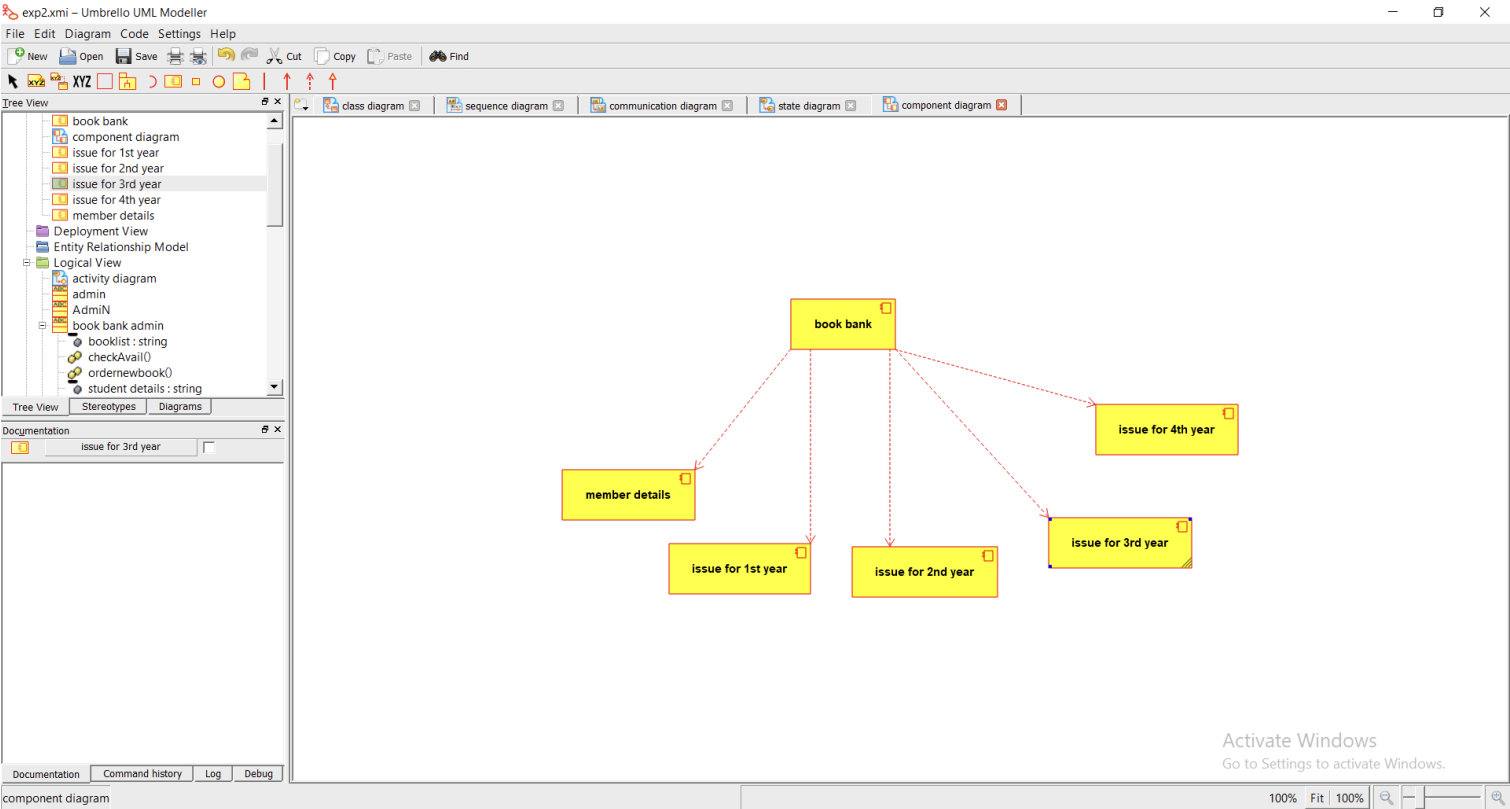
exp1_usecase.xmi [modified] - Umbrello UML Modeller

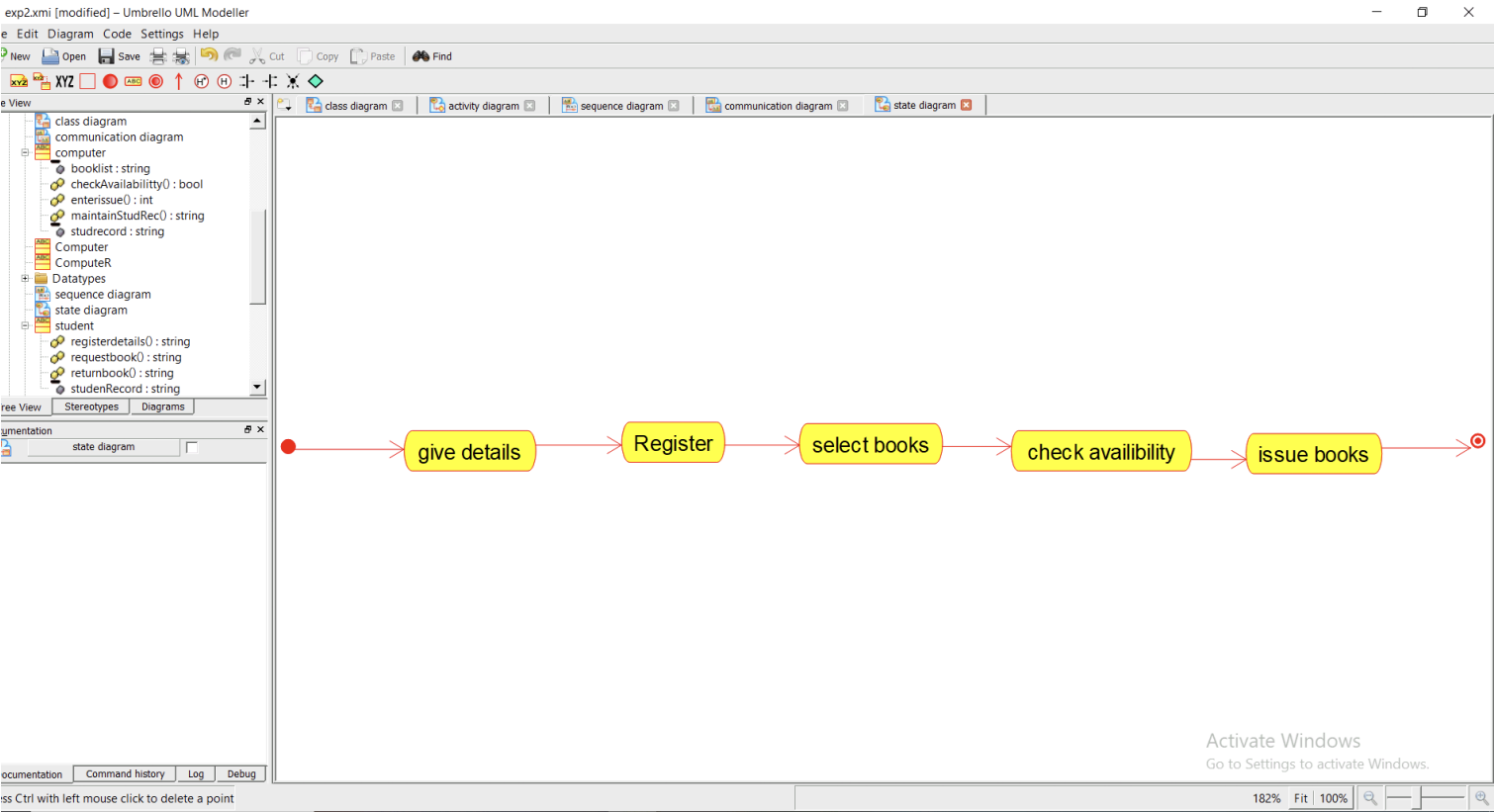


exp1_usecase.xmi [modified] - Umbrello UML Modeller









Untitled [modified] – Umbrello UML Modeller

File Edit Diagram Code Settings Help

New Open Save Print Undo Redo Cut Copy Paste Find

Tree View

Deployment View

Entity Relationship Model

Logical View

book bank admin

booklist : string

checkAvail()

ordernewbook()

student details : string

verifyStuddetail()

class diagram

computer

booklist : string

checkAvailability() : bool

enterissue() : int

maintainStudRec() : string

studrecord : string

Datatypes

student

Documentation

book bank admin

Documentation Command history Log Debug

use case diagram

class diagram

computer

- studrecord : string

- booklist : string

+ enterissue() : int

+ checkAvailability() : bool

+ maintainStudRec() : string

student

- studenRecord : string

+ requestbook() : string

+ returnbook() : string

+ registerdetails() : string

book bank admin

- student details : string

- booklist : string

+ verifyStuddetail()

+ checkAvail()

+ ordernewbook()

Welcome

Welcome

Welcome to Umbrello UML Modeller an UML diagram tool that can support you in the software development process. Especially during the analysis and design phases of this process, Umbrello UML Modeller will help you to get a high quality product. UML can also be used to document your software designs to help you and your fellow developers.

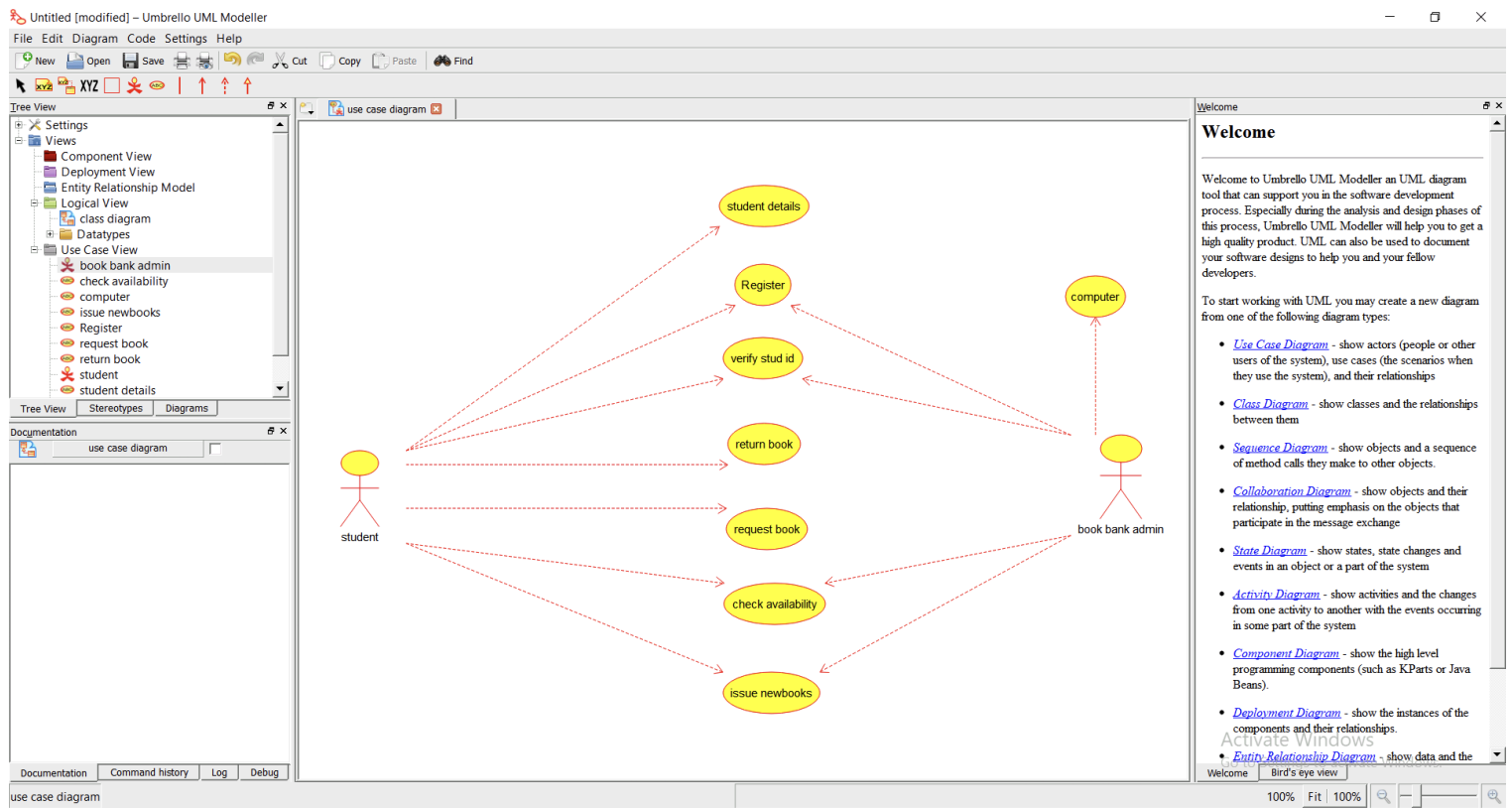
To start working with UML, you may create a new diagram from one of the following diagram types:

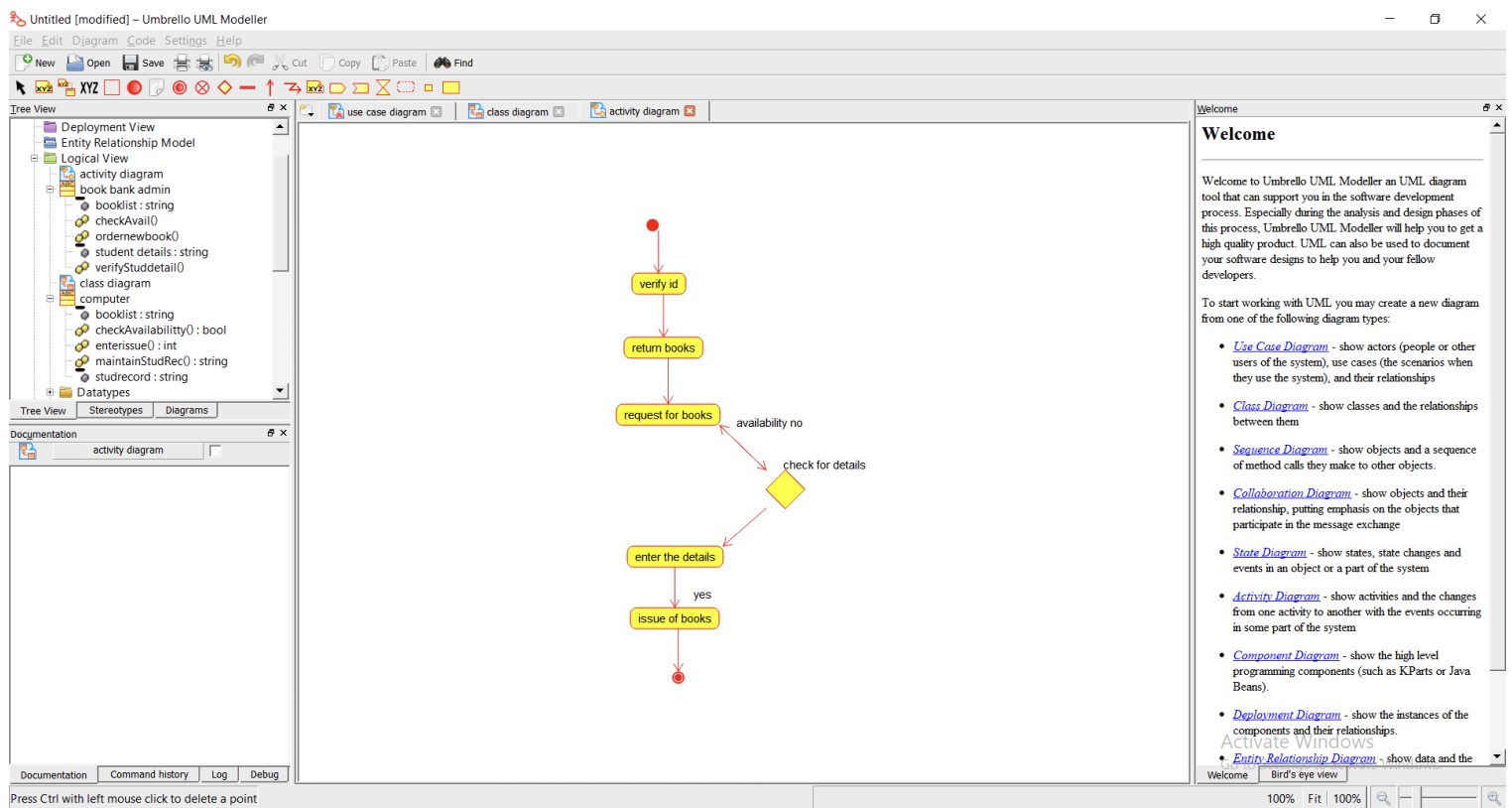
- [Use Case Diagram](#) - show actors (people or other users of the system), use cases (the scenarios when they use the system), and their relationships
- [Class Diagram](#) - show classes and the relationships between them
- [Sequence Diagram](#) - show objects and a sequence of method calls they make to other objects.
- [Collaboration Diagram](#) - show objects and their relationship, putting emphasis on the objects that participate in the message exchange
- [State Diagram](#) - show states, state changes and events in an object or a part of the system
- [Activity Diagram](#) - show activities and the changes from one activity to another with the events occurring in some part of the system
- [Component Diagram](#) - show the high level programming components (such as KParts or Java Beans).
- [Deployment Diagram](#) - show the instances of the components and their relationships.
- [Entity Relationship Diagram](#) - show data and the

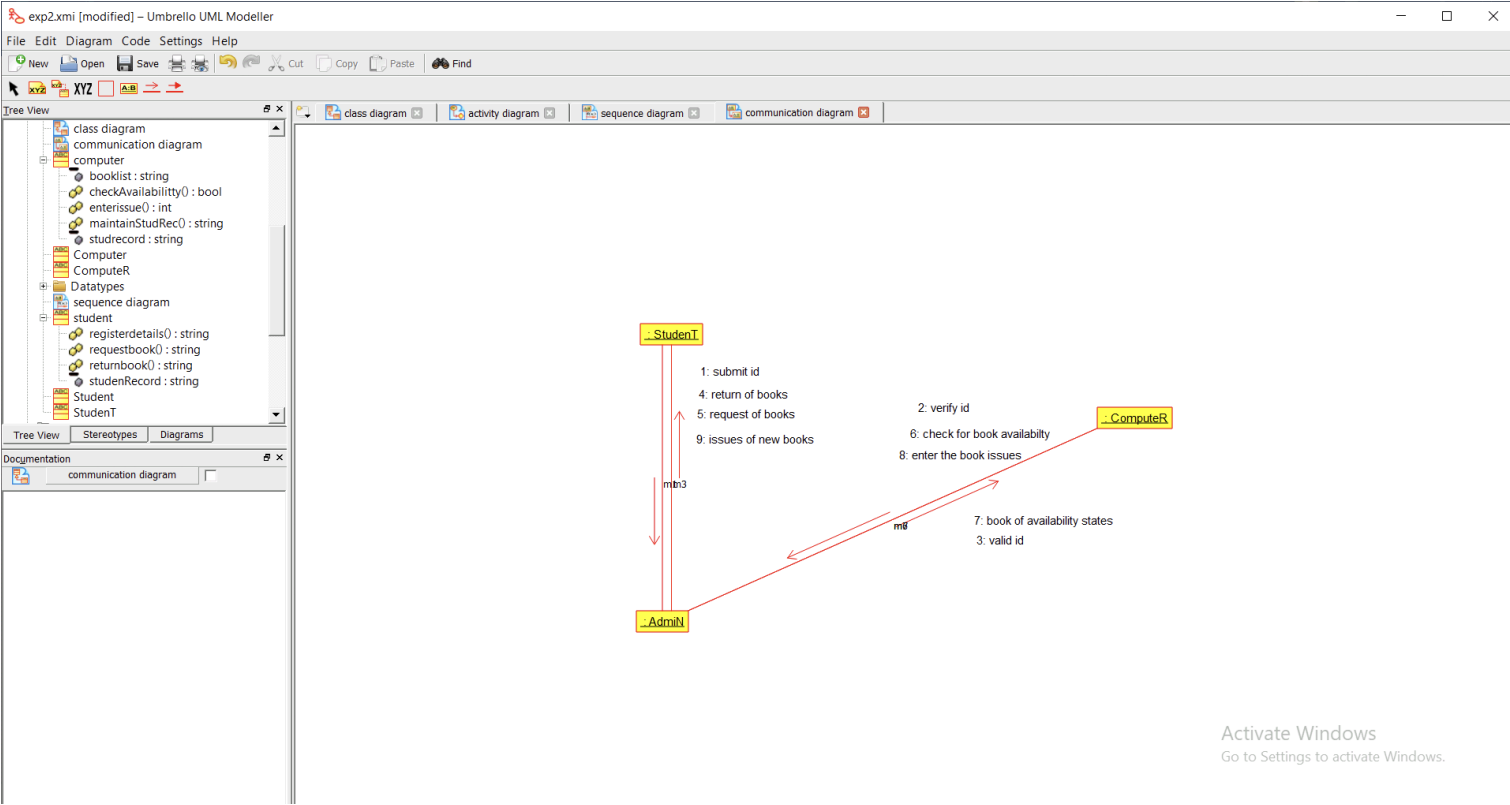
Activate Windows

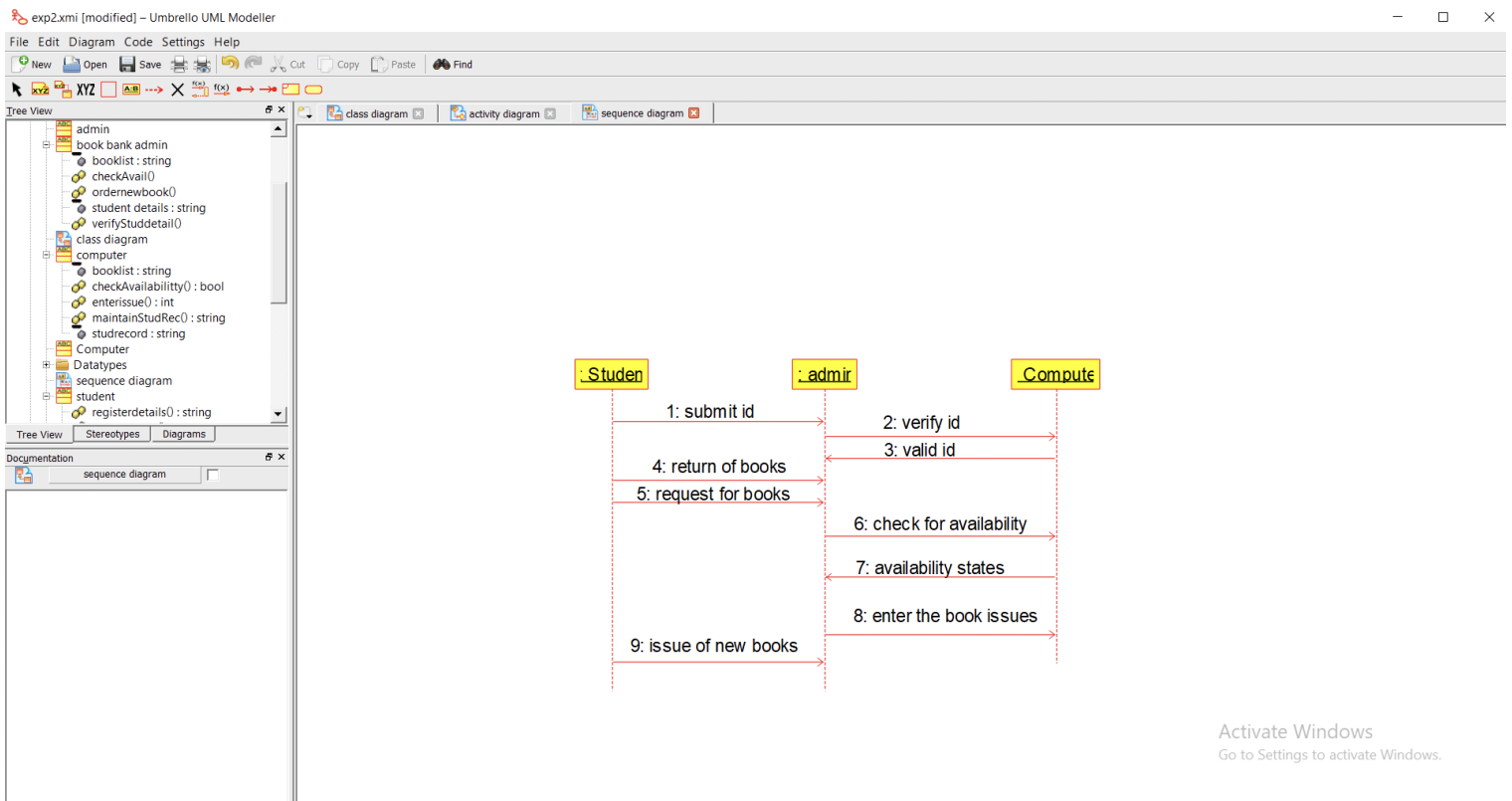
Welcome Bird's eye view

100% Fit 100%









exp1_usecase.xmi [modified] – Umbrello UML Modeller

File Edit Diagram Code Settings Help

New Open Save Print Cut Copy Paste Find

XYZ A.B C.D E.F

Tree View

applicant details
applicant proof
deployment diagram
issue of passport
passport management system
verification of proof
Entity Relationship Model
Logical View
activity diagram
Applicant
Applicant details : string
Applicant Name : string
apply passport() : string
submit proof() : string
applicant details
applicant details
class diagram
communication diagram

Tree View Stereotypes Diagrams

Documentation

deployment diagram

passport_auto_sys state diagram communication diagram deployment diagram

```
graph TD; PMS[passport management system] --> AD[applicant details]; PMS --> AP[applicant proof]; PMS --> VOP[verification of proof]; PMS --> IP[issue of passport];
```

Documentation Command history Log Debug

100% Fit 100%

Welcome

Welcome to Umbrello UML Modeller an UML diagram tool that can support you in the software development process. Especially during the analysis and design phases of this process, Umbrello UML Modeller will help you to get a high quality product. UML can also be used to document your software designs to help you and your fellow developers.

To start working with UML, you may create a new diagram from one of the following diagram types:

- [Use Case Diagram](#) - show actors (people or other users of the system), use cases (the scenarios when they use the system), and their relationships
- [Class Diagram](#) - show classes and the relationships between them
- [Sequence Diagram](#) - show objects and a sequence of method calls they make to other objects.
- [Collaboration Diagram](#) - show objects and their relationship, putting emphasis on the objects that participate in the message exchange
- [State Diagram](#) - show states, state changes and events in an object or a part of the system
- [Activity Diagram](#) - show activities and the changes from one activity to another with the events occurring in some part of the system
- [Component Diagram](#) - show the high level programming components (such as KParts or Java Beans).
- [Deployment Diagram](#) - show the instances of the components and their relationships.
- [Entity Relationship Diagram](#) - show data and the

Welcome Bird's eye view

Press Ctrl with left mouse click to delete a point

