

UML Covers...

- * Structure Diagrams

- * Class Diagram

- * Object Diagram

- * Package Diagram

- * Composite St

- * Component

- * Deployment

- * Profile Diagram

- * Behaviour Diagrams

- * Use Case Diagram

- * Information Flow Diagram

- * Activity Diagram

- * State Machine Diagram

- * Interaction Diagram

- * Sequence Diagram

- * And More!!!

Assignment Project Exam Help

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

We Will Use...

Assignment Project Exam Help

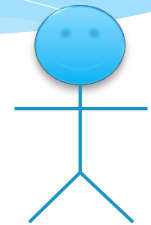
- * Use Case (M
- * Class Diagram <https://eduassistpro.github.io/>
- * Sequence Diagram

Add WeChat edu_assist_pro

Use Case Diagram

Assignment Project Exam Help

- * Actor – Represents a system.
The actor is <https://eduassistpro.github.io/>
- * Use Case – A usage scenario
Add WeChat edu_assist_pro
detailed scenario should be . Name
can be in bubble or below.
- * Relationship – Shows a logical link between actor and use case. Use line or arrow.



Use Case Diagram - Advanced

Assignment Project Exam Help

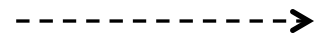
- * System Boundary – system domain limits between use cases
- * Each box represents use cases falling within that represented inside the box.
- * Inclusion – show relationships between use cases with common behaviour
- * Extension – show relationships with use cases that have the same goal but extra substeps



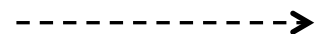
<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

<< include >>



<< extend >>



Class Diagram

Assignment Project Exam Help

- * Provide a de types and relationships <https://eduassistpro.github.io/>
- * It is independent of the impl
- * Allows software engineers to Add WeChat edu_assist_pro the design of the system before development

UML Class Diagram

Assignment Project Exam Help

- * Describes each class in the system

- * Class Name (

- * All attributes

- * All methods, including arguments and return types

- * Additional Details

- * + public member

- * # protected member

- * - private member

- * static member

- * *abstract operation*

ClassName
attributeName:type
methodName():returnType

<https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

Student
+ studentID: Integer
- password: String
enroll(Course course): Boolean

Abstract Classes and Interfaces

Assignment Project Exam Help

- * Abstract Class *could be italic*
- * Interface class <https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

<i>ClassName</i>

InterfaceName

Extension and Implementation

Assignment Project Exam Help

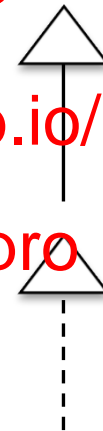
- * Classes that

- * Show inher <https://eduassistpro.github.io/>

Add WeChat edu_assist_pro

- * Classes the implement anot

- * Show interfaces and classes



Multiplicity

Assignment Project Exam Help

* 0..1

* 1

* 0..*

* 1..*

* [Int]

* [Int]..[Int]

<https://eduassistpro.github.io/>

Add WeChat ^Zedu_assist_pro
O

Exactly a Number

Exactly a Range (inclusive)

UML Sequence Diagram

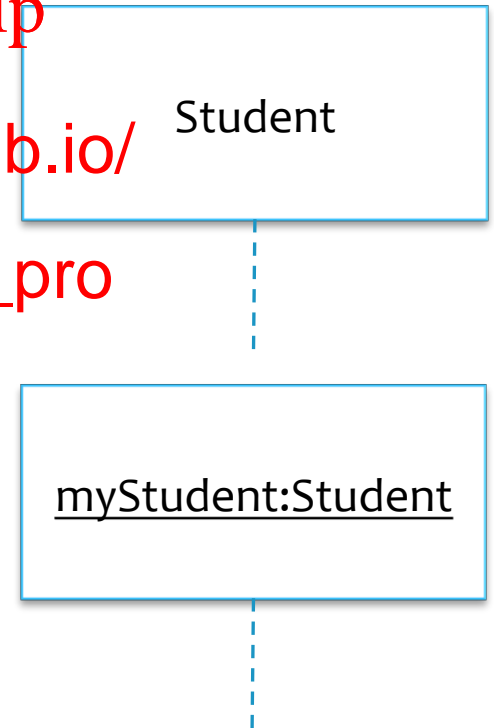
Assignment Project Exam Help

- * Shows the objects that should occur and the relationships during execution
<https://eduassistpro.github.io/>
- * Shows communication between objects, which can aid in updating software and refactoring
Add WeChat edu_assist_pro

UML Sequence Diagrams



* Lifeline

- * This can represent an instance of a class. It can be underlined.
- * This distinction is only needed if you have multiple named objects of the same type.



UML Sequence Diagrams

Assignment Project Exam Help

- * Messages – show function calls between objects
 - * Synchronous 
 - * Asynchronous 
- * The message represents a method in the receiving object
 - * The method signature should be noted above the message line
- * Return value is noted using a dotted line, with the value type/name written above the line

