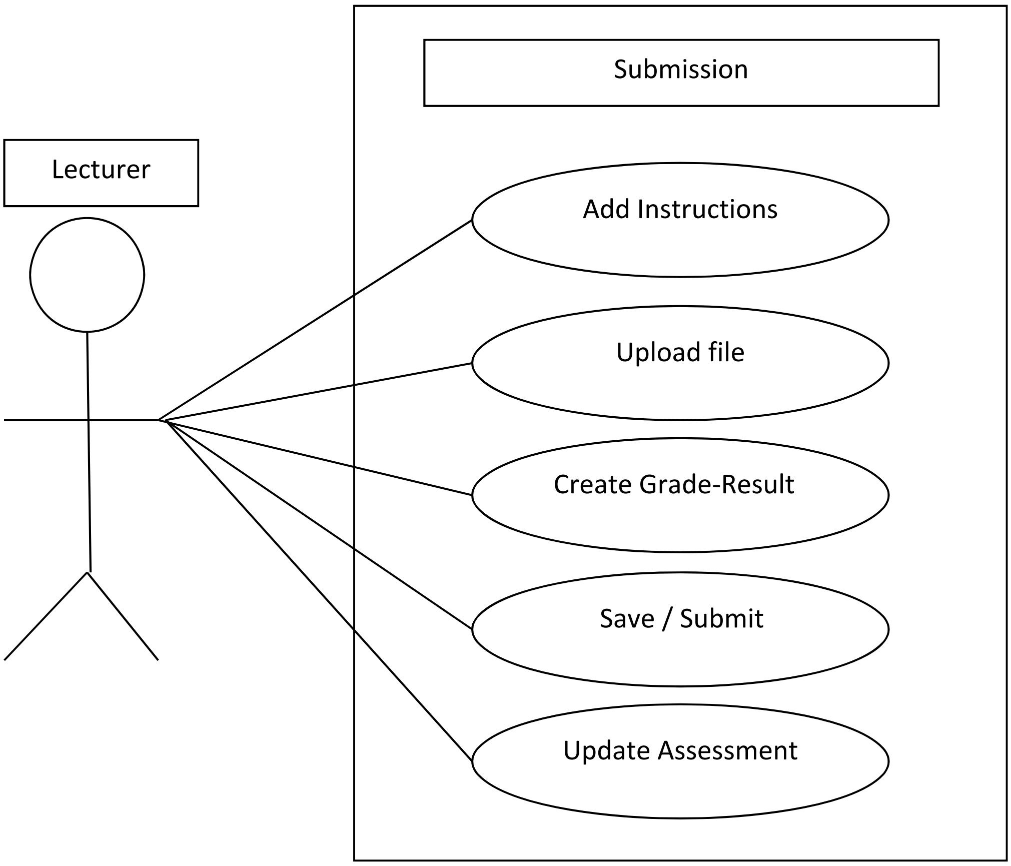
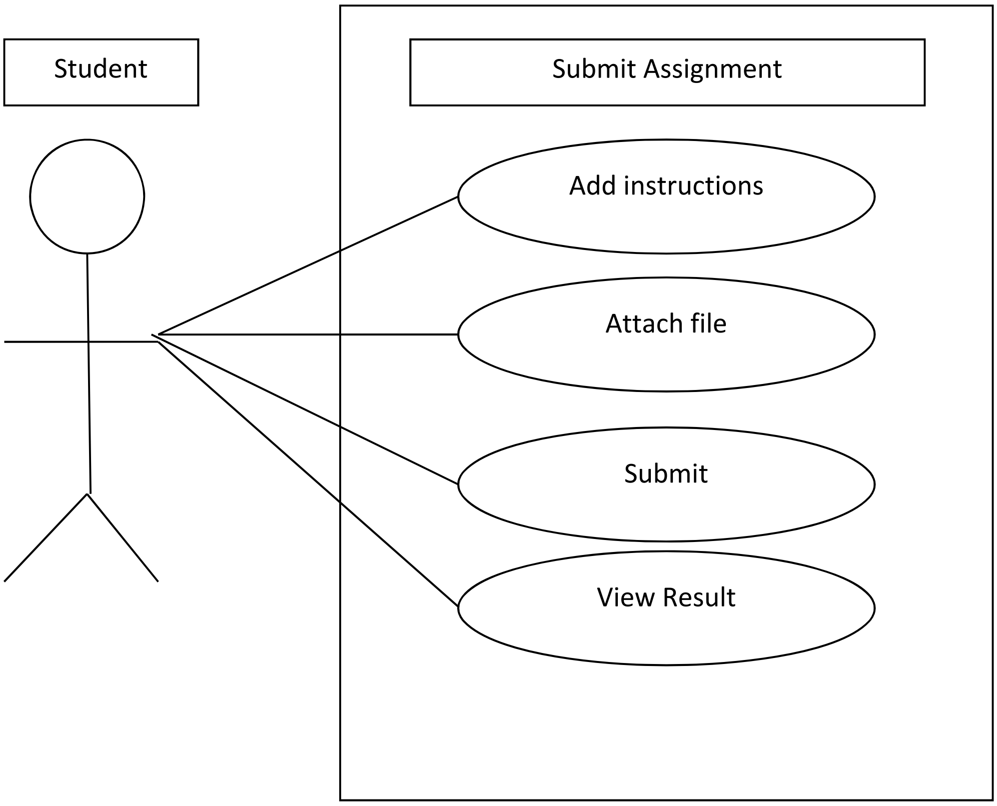
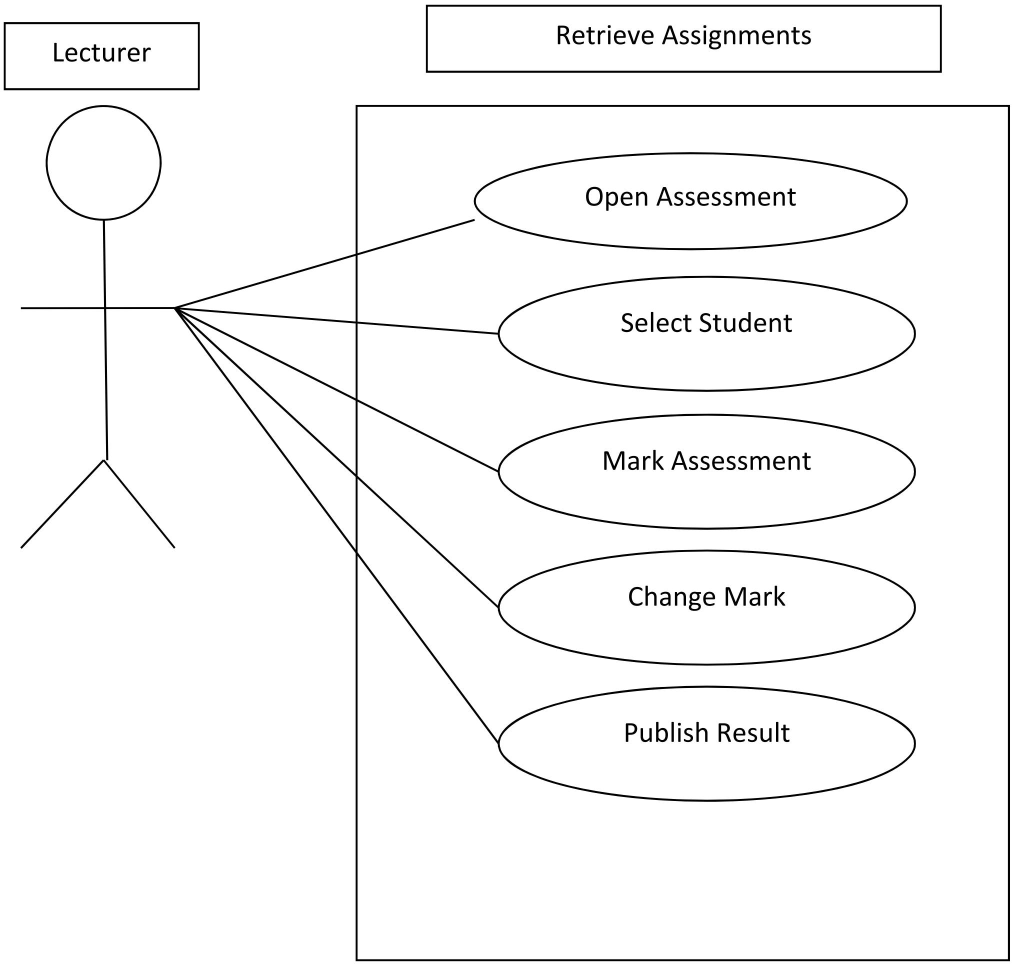
***Task 7.1:***

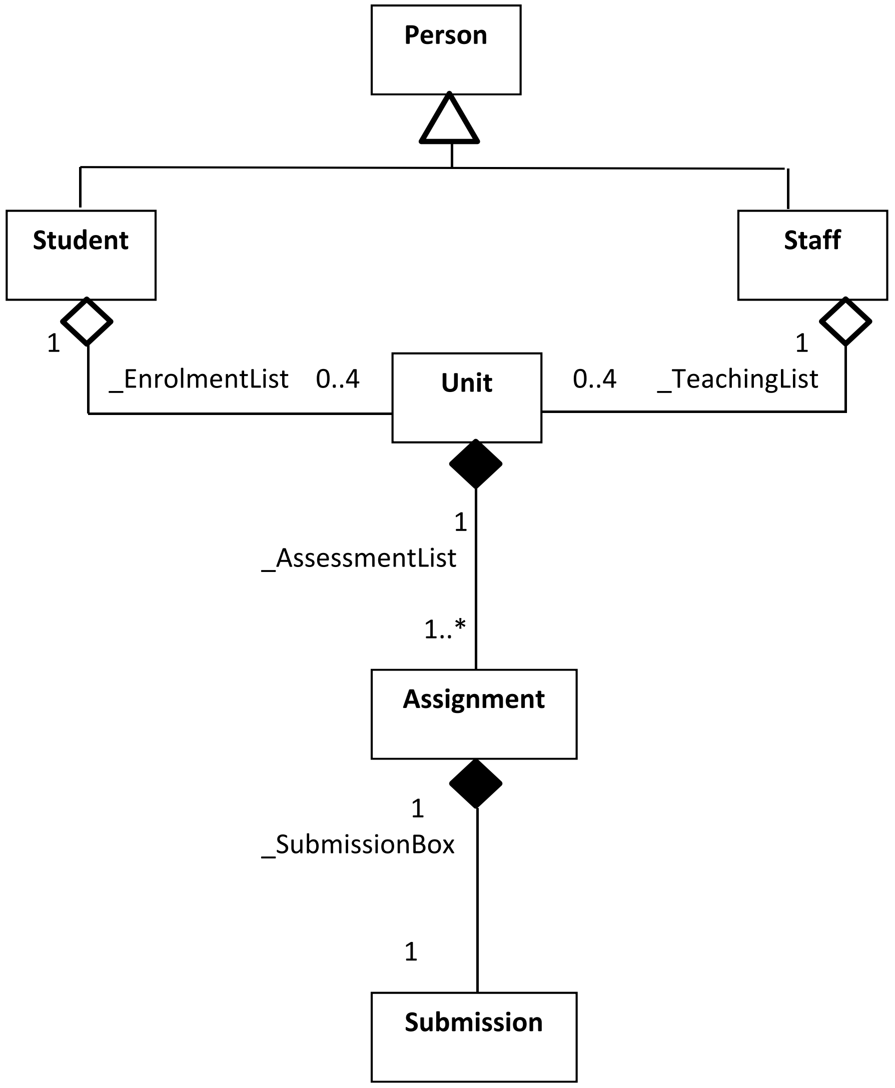

Submission
Lecturer
Add Instructions
Upload file
Create Grade-Result
Save / Submit
Update Assessment

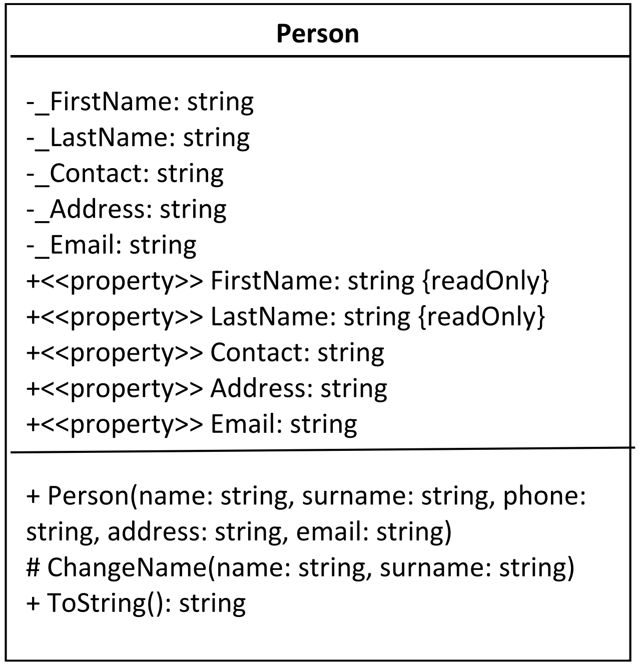

Student
Submit Assignment
Add instructions
Attach file
Submit
View Result



## *Task 7.2:*

see wk7t2.vsd

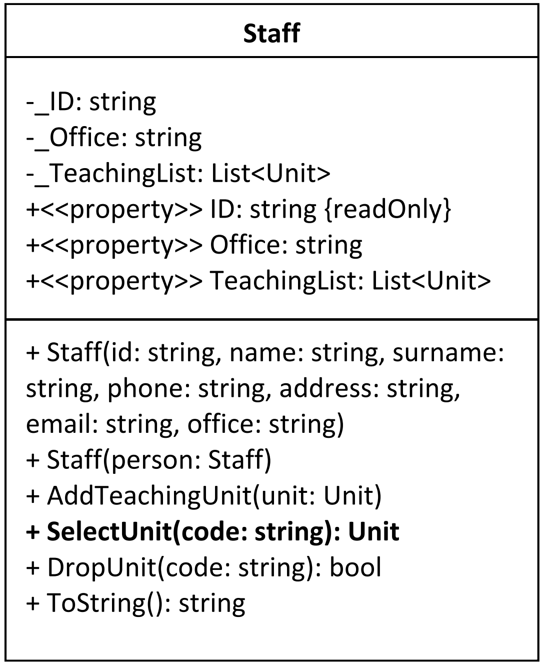

Person
Student
Staff
1
_EnrolmentList
0..4
Unit
0..4
_TeachingList
1
1
_AssessmentList
1..*
Assignment
1
_SubmissionBox
1
Submission



**Person**

-_FirstName: string
-_LastName: string
-_Contact: string
-_Address: string
-_Email: string
+<<property>> FirstName: string {readOnly}
+<<property>> LastName: string {readOnly}
+<<property>> Contact: string
+<<property>> Address: string
+<<property>> Email: string

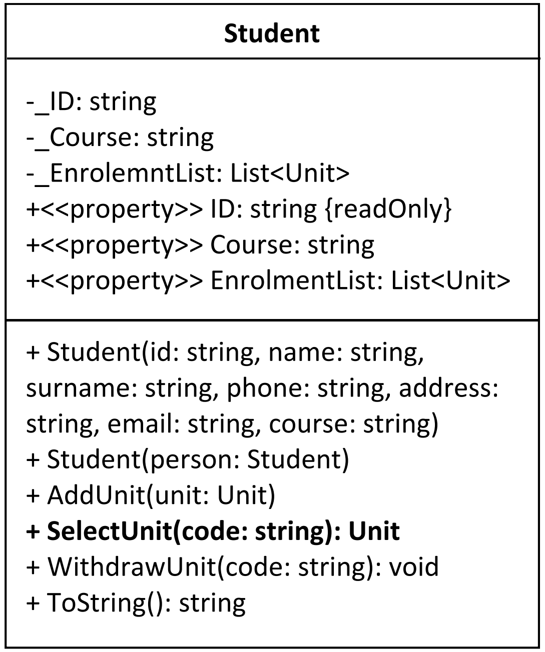
+ Person(name: string, surname: string, phone: string, address: string, email: string)
# ChangeName(name: string, surname: string)
+ ToString(): string



**Staff**

-_ID: string
-_Office: string
-_TeachingList: List<Unit>
+<<property>> ID: string {readOnly}
+<<property>> Office: string
+<<property>> TeachingList: List<Unit>

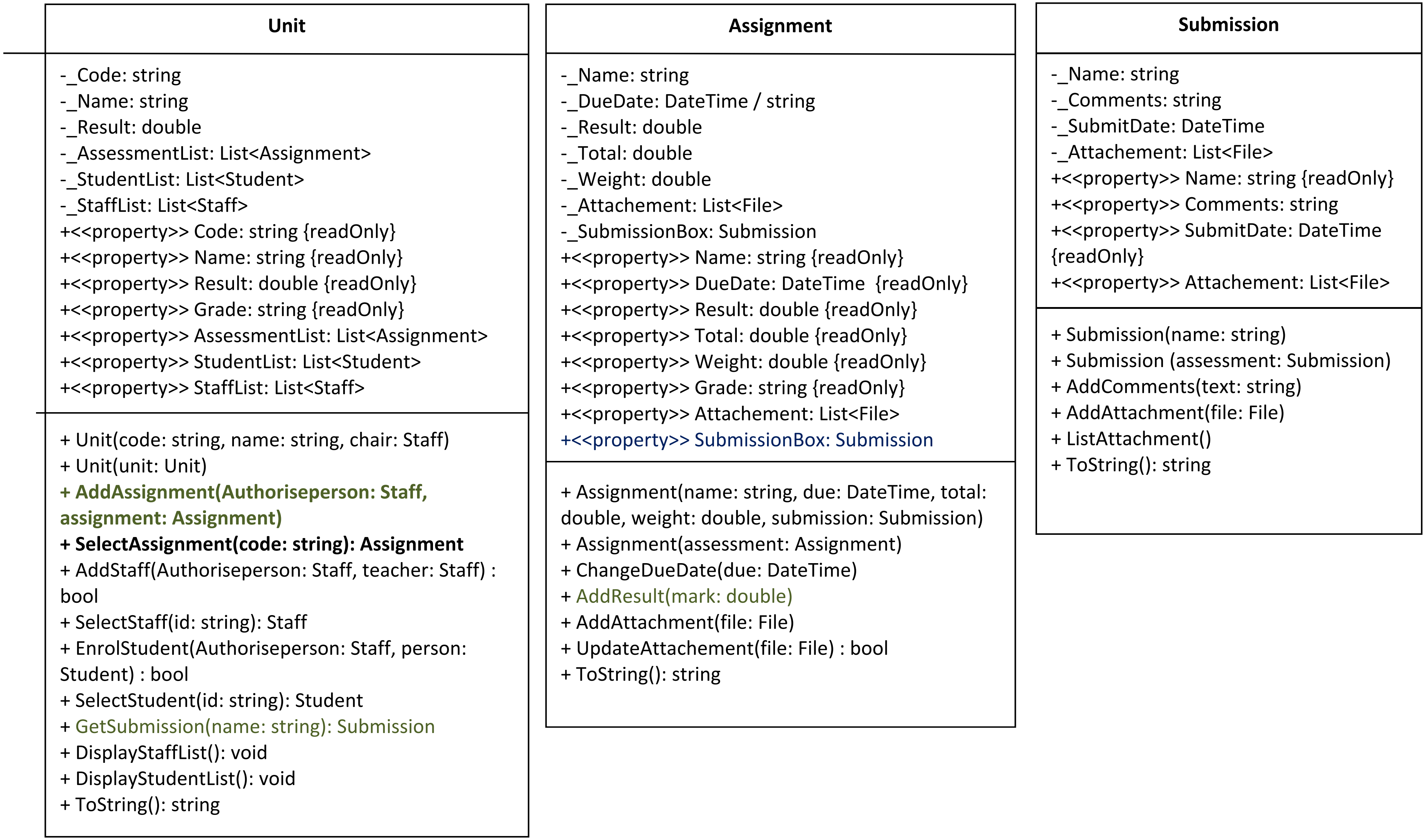
+ Staff(id: string, name: string, surname: string, phone: string, address: string, email: string, office: string)
+ Staff(person: Staff)
+ AddTeachingUnit(unit: Unit)
**+ SelectUnit(code: string): Unit**
+ DropUnit(code: string): bool
+ ToString(): string



**Student**

-_ID: string
-_Course: string
-_EnrolemntList: List<Unit>
+<<property>> ID: string {readOnly}
+<<property>> Course: string
+<<property>> EnrolmentList: List<Unit>

+ Student(id: string, name: string, surname: string, phone: string, address: string, email: string, course: string)
+ Student(person: Student)
+ AddUnit(unit: Unit)
**+ SelectUnit(code: string): Unit**
+ WithdrawUnit(code: string): void
+ ToString(): string



**Unit**

-_Code: string
-_Name: string
-_Result: double
-_AssessmentList: List<Assignment>
-_StudentList: List<Student>
-_StaffList: List<Staff>
+<<property>> Code: string {readOnly}
+<<property>> Name: string {readOnly}
+<<property>> Result: double {readOnly}
+<<property>> Grade: string {readOnly}
+<<property>> AssessmentList: List<Assignment>
+<<property>> StudentList: List<Student>
+<<property>> StaffList: List<Staff>

+ Unit(code: string, name: string, chair: Staff)
+ Unit(unit: Unit)
**+ AddAssignment(Authoriseperson: Staff, assignment: Assignment)**
**+ SelectAssignment(code: string): Assignment**
+ AddStaff(Authoriseperson: Staff, teacher: Staff) : bool
+ SelectStaff(id: string): Staff
+ EnrolStudent(Authoriseperson: Staff, person: Student) : bool
+ SelectStudent(id: string): Student
+ GetSubmission(name: string): Submission
+ DisplayStaffList(): void
+ DisplayStudentList(): void
+ ToString(): string

**Assignment**

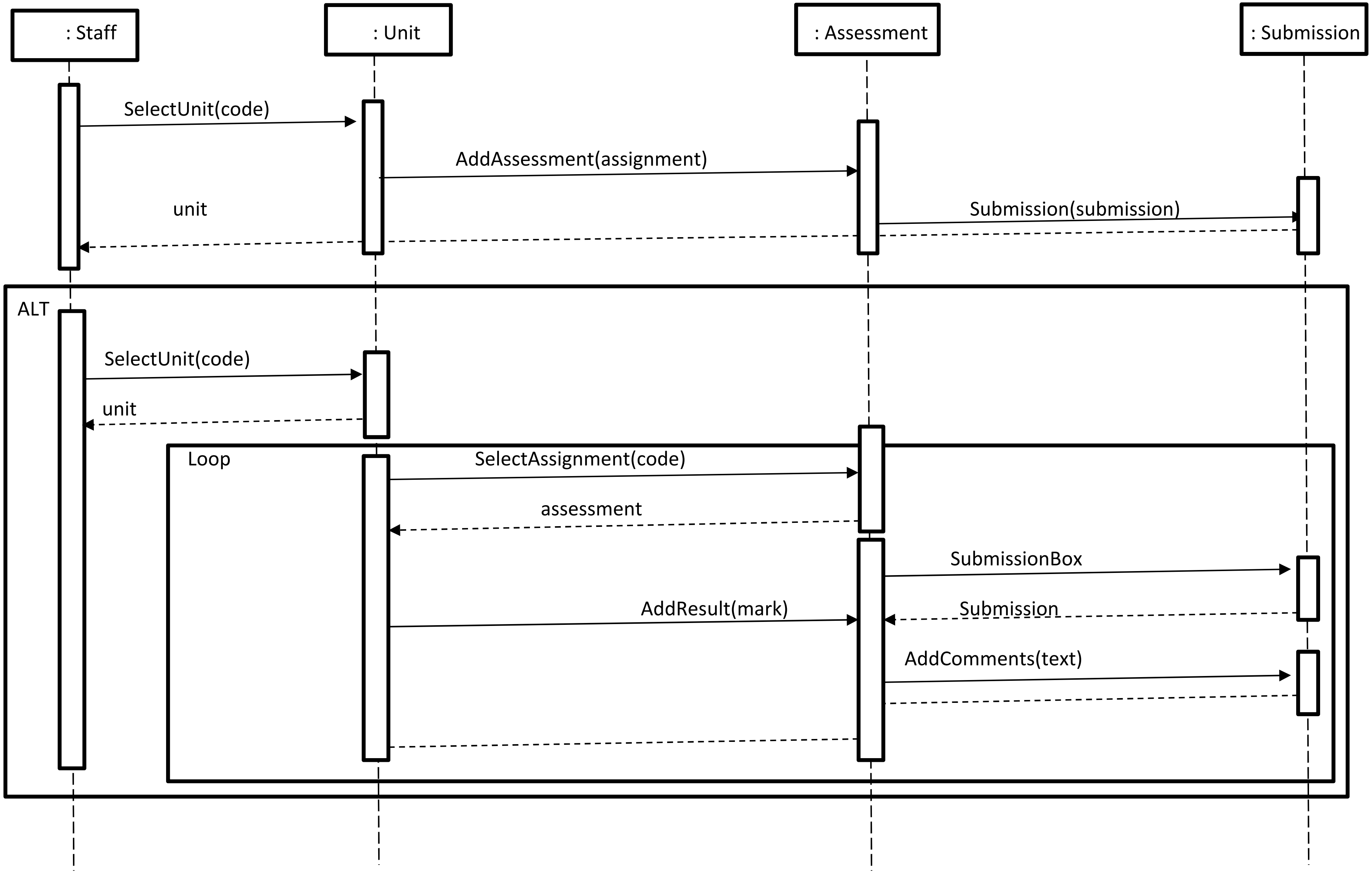
-_Name: string
-_DueDate: DateTime / string
-_Result: double
-_Total: double
-_Weight: double
-_Attachement: List<File>
-_SubmissionBox: Submission
+<<property>> Name: string {readOnly}
+<<property>> DueDate: DateTime {readOnly}
+<<property>> Result: double {readOnly}
+<<property>> Total: double {readOnly}
+<<property>> Weight: double {readOnly}
+<<property>> Grade: string {readOnly}
+<<property>> Attachement: List<File>
+<<property>> SubmissionBox: Submission

+ Assignment(name: string, due: DateTime, total: double, weight: double, submission: Submission)
+ Assignment(assessment: Assignment)
+ ChangeDueDate(due: DateTime)
+ AddResult(mark: double)
+ AddAttachment(file: File)
+ UpdateAttachement(file: File) : bool
+ ToString(): string

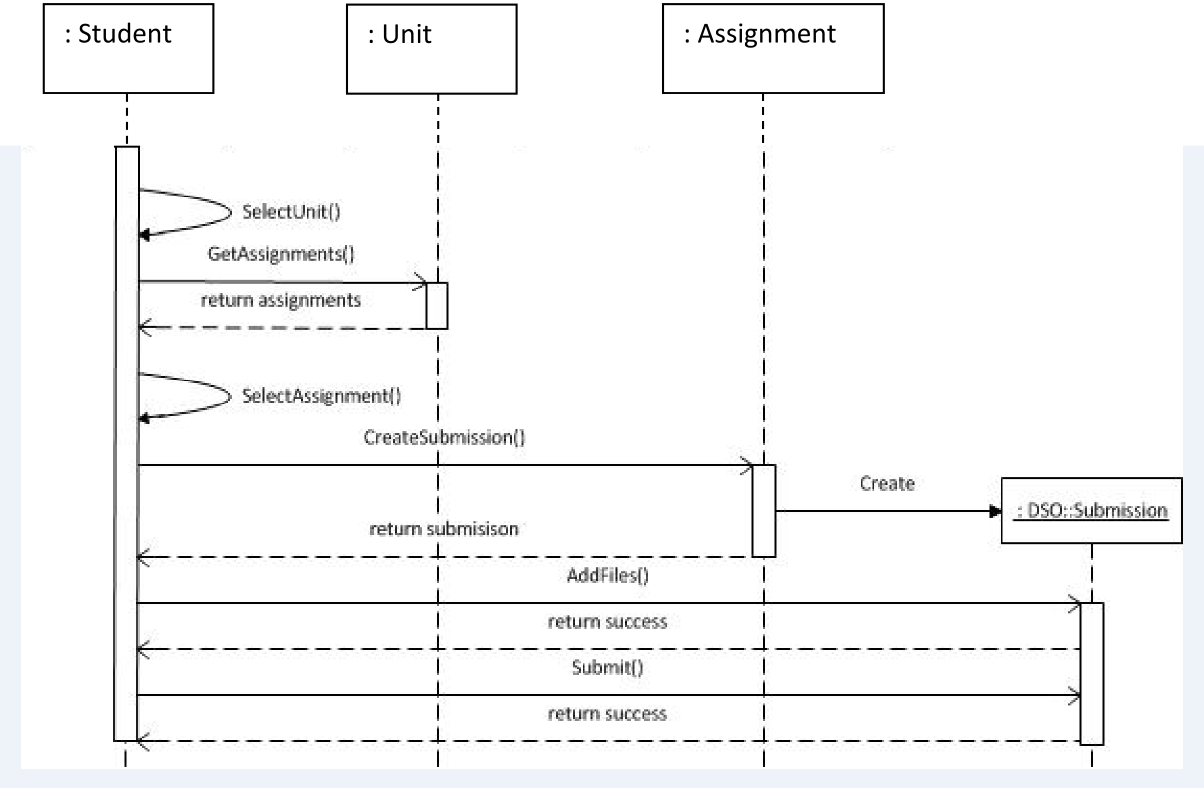
**Submission**

-_Name: string
-_Comments: string
-_SubmitDate: DateTime
-_Attachement: List<File>
+<<property>> Name: string {readOnly}
+<<property>> Comments: string
+<<property>> SubmitDate: DateTime {readOnly}
+<<property>> Attachement: List<File>

+ Submission(name: string)
+ Submission (assessment: Submission)
+ AddComments(text: string)
+ AddAttachment(file: File)
+ ListAttachment()
+ ToString(): string


: Staff
: Unit
: Assessment
: Submission
SelectUnit(code)
AddAssessment(assignment)
unit
Submission(submission)
ALT
SelectUnit(code)
unit
Loop
SelectAssignment(code)
assessment
SubmissionBox
AddResult(mark)
Submission
AddComments(text)

http://www.sparxsystems.com.au/resources/uml2_tutorial/uml2_sequencediagram.html (sd fragment)



## *Task 7.4:*

Association class: An association class is a diagram that shows an association relationship in a generalised overview.

Multiplicity: Multiplicity is the range or number of the objects.

UML activity diagram: This is similar to flow charts. It shows the flow of a process. It also allows to model parallel operations.

UML class diagram: This diagram represents the structure of an object.

UML communication diagram: This diagram is a type of interaction diagram except that it emphasises the structure and connection of objects.

UML interaction diagram: This shows how objects exchange messages to create a behaviour.

UML object diagram: This is used to clarify complex class diagrams by showing an example of that the class and sub classes would look like at run-time.

UML sequence diagram: This diagram is a type of interaction diagram except that it emphasises the order of the events.

UML state machine diagram: A state machine diagram shows different states of a system and all the cases that would cause transition from one state to another.

UML use case diagram: These diagrams show what the system does and provides.

Use case: The Use case is a group of scenarios that show interaction with the system.