Classes and entities

Task 1

Interface `Stack` defines an operation `push` to the stack with a parameter `obj` of type `Element`, an operation pop that extracts the top elements of the stack with return value of type `Element`. Use class diagrams to present the solution.

a) Add operation `reset` to interface `Stack` without parameters, and a static operation `createNew` that creates and returns a new instance of `Stack`.

b) Show on the diagram that interface `Stack` depends on class `Element`.

c) Add class `ListStack`, which implements interface `Stack`. Show operations in the class that implement the interface.

d) Add a private structural property `arr` of type `Element` into class `ListStack` with multiplicity greater than zero, values of which are sorted in some order and may contain duplicates.

e) Add a private integer read-only attribute `increment` and a protected operation `resize` to change stack size with integer parameter `newSize`.

f) Show on a diagram instance `stack` of class `ListStack`, an `arr` property of which contains the first item `first` of type `Element` and `second` as the second item. Set attribute `increment` of the instance stack to `10`.

Task 2

An abstract class `Account` has two derived classes: a consumer account PersonalAccount and a company account CompanyAccount. Use UML2 class diagrams.

a) Add a class `Person` with a public attribute FullName of string type and connect the class with `PersonalAccount` with an association Owns with an end `owner` at `Person` and navigable end account at PersonalAccount

b) Using an anonymous association in a similar way, add an owner of type Company to a CompanyAccount and give association ends appropriate names

c) Add class Address with string attributes street and city and a positive integer attribute building. Using new anonymous associations specify that a Person can have a permanent address registeredAt, actual address actual, while a Company could be linked with a legal address legalAddress and postal address postAddress.

Task 3

Smart country house SmartHouse consists of four walls Wall and a roof Roof. The house reacts to storm notifications stormWarning and hardens the roof with harden, closes windows closeWindows. All the building materials Material have feature price and unit weight unitWeight.

a) Add to the model the following materials: red and white bricks Brick, wood planks Plank made of oak or pine.

b) Specify that bricks are the material for the walls. Using associations specify that the roof frame Frame is made of no more than forty planks and can be of one of these FrameKind: triangle roof, plain roof and French (mansard) roof.

c) A roof frame can be covered with a Tiling material, add this to the model.

d) Suppose we invent a universal building material that substitutes planks, bricks and tiling. Build a country house out of it. How many instances of the material will you need? Explain your answer.

Task 4

A `Teacher teaches` several courses `CourseOffering`. Use ER diagrams to represent the model.

a) Using the appropriate type of relation, show that a course consists of a single `Lecture` and several `Practice`.

b) Specify that a teacher gives lectures as a `lecturer` and conducts practice as an `assistant`.

c) Show that there could be several tasks at the practice, each related to some topic given at the lecture. At least one topic is covered at a lecture, but practice may exercise no tasks.

d) Each task given at practice has a unique `ID`, a `text` and a `correct answer`.

Use cases

Task 5

Actor User interacts with a system OnlineTranslator in an abstract use case Translate. Use cases TranslateText and TranslateWebPage detail Translate. Show this at a use case diagram.

a) A use case TranslateWebPage includes «include» a use case SetURL.

b) A use case SetLanguages extends «extend» another use case Translate in an extension point specifyLaguages. Extension condition “language is not detected ”.

c) Add an ExperiencedUser actor derived from User. An ExperiencedUser can interact with the system in a ProposeTranslation use case, which details the TranslateText use case.

Task 6

A collector Cashier and utility operator Loader take care of a vending machine. Collector is responsible for money collection CollectCash while operator replaces water tanks ChangeWater and gas ChangeGas.

a) Extract a common maintenance use case, which includes authorization in the system and finalization of the maintenance session.

b) Add that the machine could also be loaded with syrup.

c) In which case a collector could load water tanks in the system? Explain your answer.

d) Add that a collector could monitor the vending machine using a remote video camera over the Internet that is activated by a motion sensor of the building. Justify your solution.