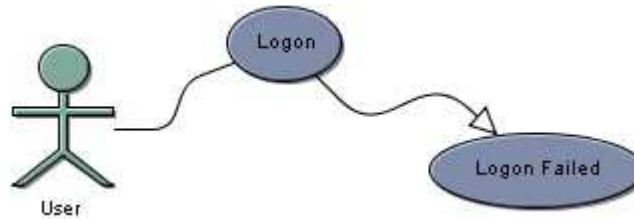


Диаграми на взаимодействие - Sequence and Collaboration

След като вече са определени случаите на употреба и са моделирани обектите в системата чрез диаграми на класове, може да се разработи динамичното поведение на системата. Диаграмите на взаимодействие описват начина, по който групи от обекти постигат съвместно някакво поведение.

Случаите на употреба обхващат взаимодействието между потребителите и системата. Диаграмата на взаимодействието описва поведението на един единствен случай на употреба, показвайки сътрудничеството на обектите в системата. Тази диаграма показва обектите в системата и съобщенията, които са разменят между тях.

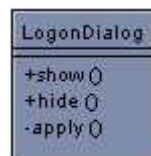


Пример: Потребител се логва в системата. Случаят на употреба се описва със следните стъпки:

1. Показва се диалога за логване
2. Потребителя въвежда потребителско име и парола
3. Потребителя натиска бутона ОК или клавиша enter
4. Потребителското име и парола се проверяват и потвърждават
5. Потребителя се допуска до системата

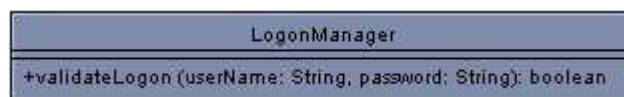
Алтернатива: Logon Failed – ако на стъпка 4 не са потвърдени потребителското име и парола, допуска се потребителя да опита отново.

Класове в системата



Класът *LogonDialog* има public методи за показване и скриване на прозорец и private метод, който се извиква, когато потребителя натисне бутона KO или клавиша enter.

Клас *LogonManager* включва метод, който връща true, ако логването е успешно и false, ако не е.



Клас *DatabaseAccess* ще позволи да се изпълни заявка към базата данни.



След като са моделирани класовете, може да се направят диаграмите на взаимодействието.

Инстанции и съобщения

Диаграмите на взаимодействието са композирани главно от инстанции (обекти) на класовете и съобщения между тях. **Инстанция** е реализацията на клас. Ако ние имаме клас *Доктор*, инстанции са *Д-р Джоунс*, *Д-р Смит*, и т.н.

В UML, инстанциите се представят с правоъгълник, в който се записва:

instanceName: datatype

Може да се зададе име на инстанцията или да не се зададе, но винаги трябва да се зададе datatype.

Под името може също така да се изброят атрибутите и техните стойности, специфични за този случай на употреба. Това се прави когато стойностите са много важни.

Под името, вие можете също така да се изброят атрибутите и техните стойности. В Визуалния Случай, вие можете да изобразите атрибути от вашия клас и влизате в новите стойности специфични за този пример. потребност на Атрибути само бъде показана когато те са важни и вие нямате да определите точно и показвате всички от атрибутите на класик

Съобщенията представляват извиквания на операциите. Ако инстанция извиква операция, се изпраща съобщение. Също така при завършването на операцията се връща обратно съобщение към инстанцията, която е инициирала повикването.

Sequence

Диаграма на последователностите се използва, за да се разгледа поведението на няколко обекта в рамките на единствен случай на употреба. Те показват добре съвместната работа между обектите. Диаграмата на последователностите показва реда, в който се случват нещата, като последователност от съобщения.

- Потокът на време е показан отгоре до долу, това означава, че съобщенията по-високо на диаграмата се случват преди тези, които са по-ниско
- Показва кое съобщение се изпраща от един обект към друг обект и кога се изпраща.
- Правоъгълниците са **инстанции** на представените класове и вертикалните линии са линиите на живот
- Хоризонталните линии показват класовете или техните роли.
- Стрелките (връзки) са **съобщения** - повиквания на операции и връщане от операции
- Съобщенията са означени с операциите, които се изпълняват. Може да се задават параметрите.

– **Class Identification** – правоъгълник с подчертано име на инстанция и клас *InstanceName : ClassName*.

– **Class Lifeline** – прекъснатата линия, показваща съществуването на обект.

– **Termination** - с X в края се означава разрушаването на обект.

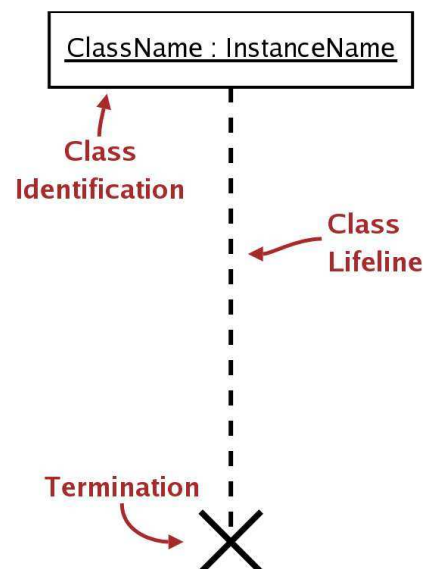
– **Activation** – Правоъгълник върху линията на обекта означава активирането на обекта.

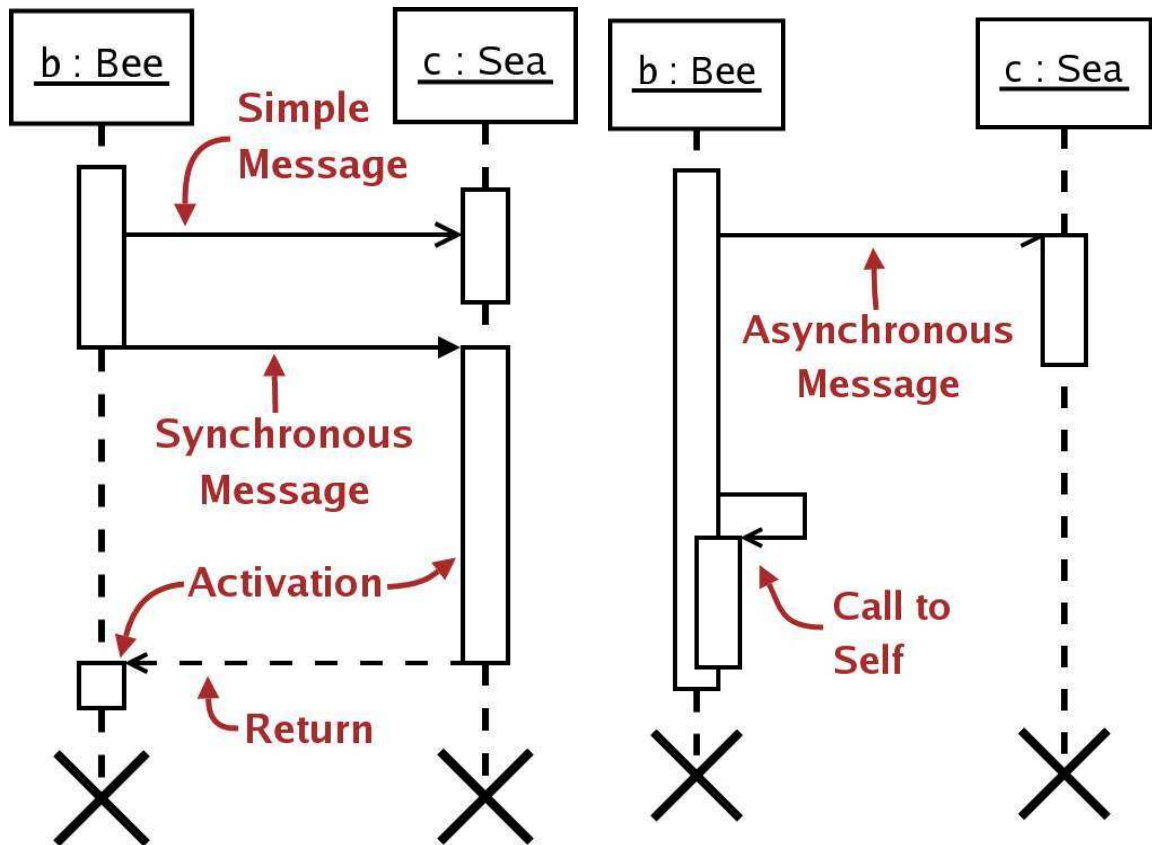
– **Simple message** – линия със стрелка, показваща съобщенията мейду обектите.

– **Synchronous message** - линия с пълтна стрелка, показваща синхронните съобщения.

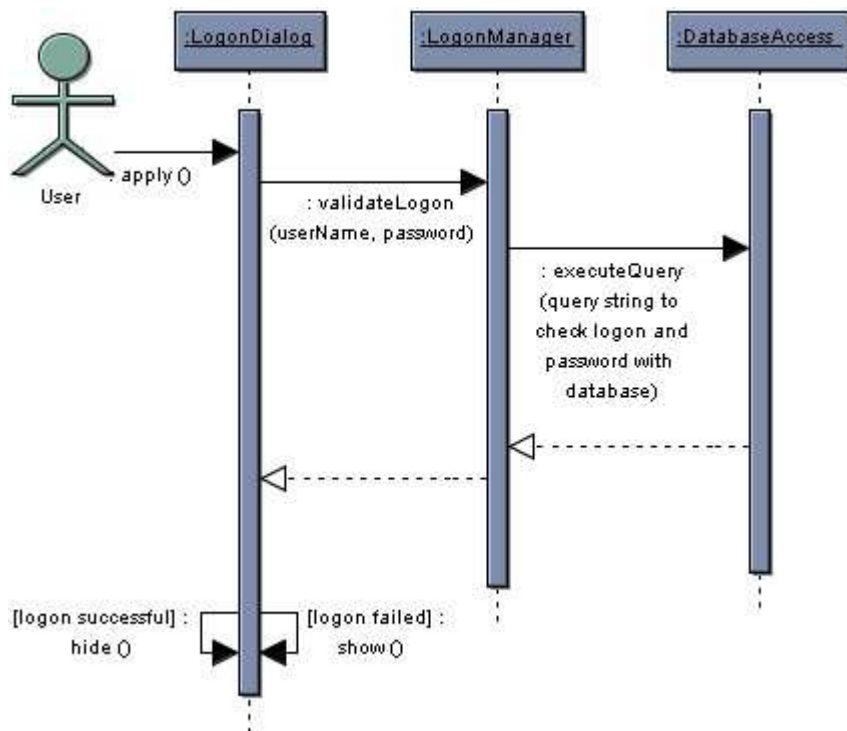
– **Asynchronous message** - линия с половин стрелка, показваща асинхронните съобщения.

– **Call to self** – обект извиква себе си.

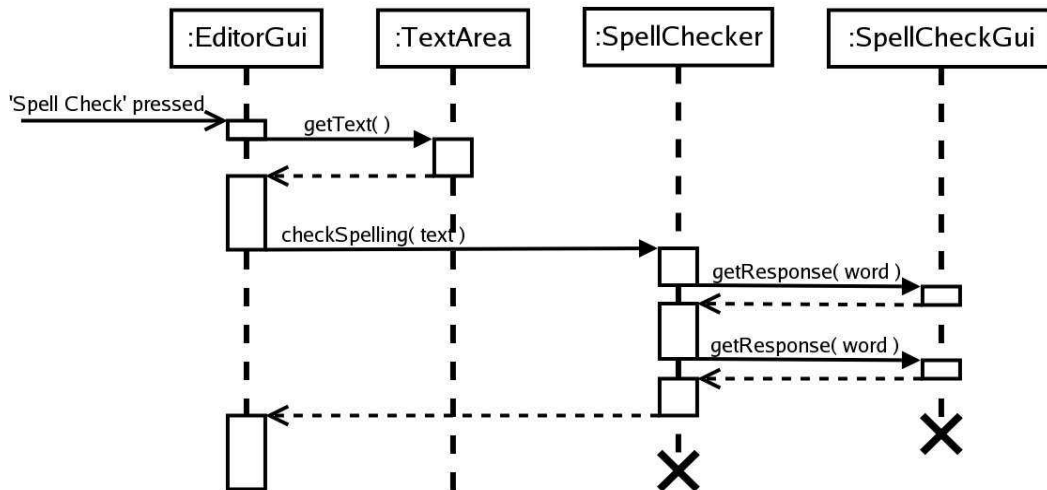




Пример за диаграма на последователност при логване в системата:



Пример за Sequence diagram на текстов редактор – проверка на правописа:



Collaboration Diagrams

Диаграмата на сътрудничеството е по-комплексна. Тя показва взаимодействията, организирани около обектите и връзките между тях.

Диаграмата на сътрудничество моделира потока на съобщенията между обектите. Тук времето отсъства.

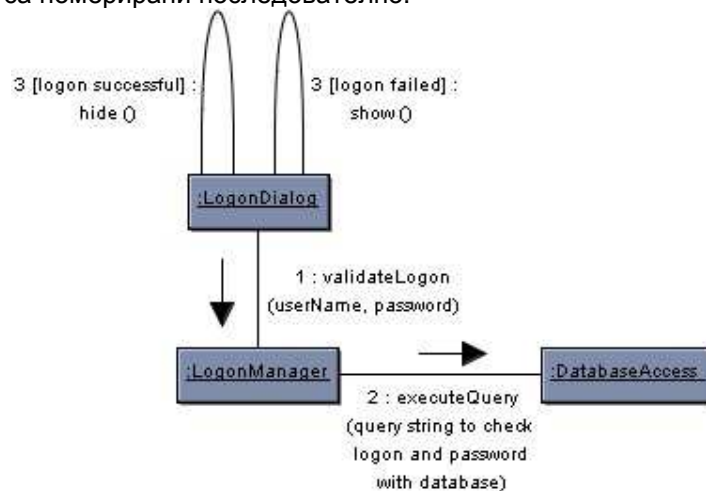
Класовете се представят в правоъгълници с имена. Форматът е: *instance/role name : class name*. Инстанциите се подчертават.

Съобщенията са както в диаграмата на последователностите.

Съобщенията имат пореден номер.

Времето се представя като номер на последователност на изпълнение.

Пример за взаимодействията при логване в системата като диаграма на сътрудничеството. Всички съобщения са номерирани последователно.



Пример за диаграма на сътрудничеството на текстов редактор – проверка на правописа.

