

Freelancing Management System
FMS

The Team — Yaser, Umar, Taha and Boda

March 25, 2017

Contents

1	System Description	4
2	System Actors	4
2.1	Root	4
2.2	Admin	4
2.3	Freelancer	4
2.4	Employer	4
3	Diagrams	5
3.1	ERD	5
3.2	Use Case	6
3.3	Class Diagram	7
3.4	System Sequence Diagrams	8
4	Use Case – <i>in detail</i>	9
4.1	Access Root	9
4.2	Specify Penalties	10
4.3	Add Admin	10
4.4	Remove Admin	11
4.5	Block Admin	11
4.6	Unblock Admin	11
4.7	Search Freelancers	11
4.8	Search about Freelancer	12
4.9	Review Employer history	13
4.10	Reject offer submitting	14
4.11	Accept Offer	15
4.12	Accept Submitting Offer	16
4.13	Reply Complaints	17
4.14	View Statistics	18
4.15	Remove Account	18
4.16	Receive Complaint	18
4.17	View Reports	18
4.18	Ban Account	19
4.19	Login	19
4.20	Review Profile	19
4.21	Create Profile	19
4.22	Review History	20
4.23	Withdraw Money	20
4.24	List Tasks	20
4.25	Apply Task	21
4.26	Cancel App	21
4.27	Overtime Reguest	21
4.28	Submit Task	22
4.29	Cancel Submit	22

4.30	Rate the employer	22
5	Design Patterns	23
5.1	Singleton Design Pattern	23
5.2	ReadOnly Pattern	24
5.3	Factory Pattern	25
5.4	Composition Pattern	26
5.5	Bridge Design Pattern	27
6	Call Us	27

1 System Description

This is a *Freelancing Management System*, a system which allows *Customers* and *Freelancers* to find one-another easily. It manages the processes from finding workers, verifying them, engaging, paying up to rating. Let us take you on an interesting tour into the *system actors*, every one of them sees the system differently according to their role.

2 System Actors

2.1 Root

The Root is the owner of the system. He is the one who set it up for the first time with a special root access from him. The Root also specifies the system settings like penalties, bonuses on the rate and ratio of the profit on each Employer-Freelancer deal. Besides, He add *Admins* to manage the system.

2.2 Admin

Admins are the managers of the system. They have *The managers access* over the hole system, they can *delete* or *ban* any consumer account. Also, They can view some Statistics like the number of the Employers and Freelancer. They can *Receive* and *Reply Complaints* of the consumers as well. Admins can view reports about the system i.e *economic, consumers, blocked, ...* reports.

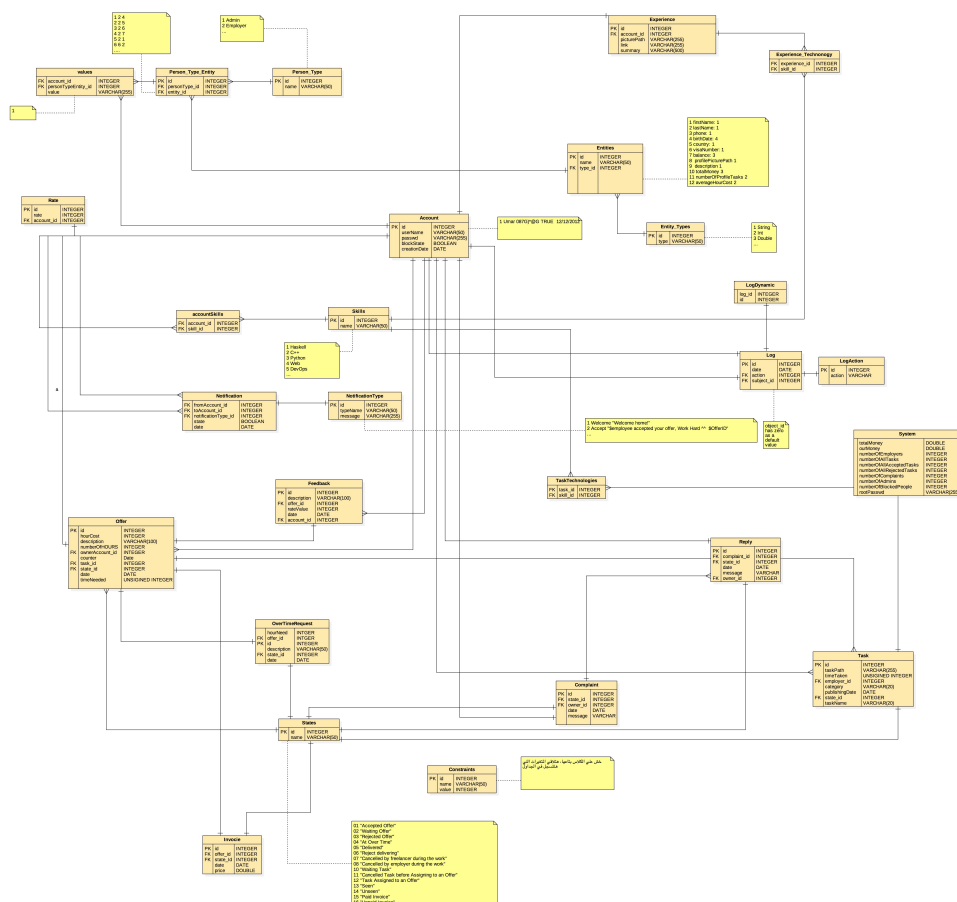
2.3 Freelancer

Freelancers are the workers of the system. Every one them have an *Profile* where he can share his experience, and work examples. His *Profile* contains enough information so that the Employer can decide wherever he is going to invest in that *freelancer* or not. There is a *Rate* in the *Freelancers Profile* is computed with very accurate algorithms based on the *Employers Feedback* and how commitment is the Freelancer! *Freelancer* are paid per hour, and it is in the *profile*. A *Freelancer* can review the *Employers' Profiles* and their *Tasks* which they can apply for it and make special offers for the *Employers*.

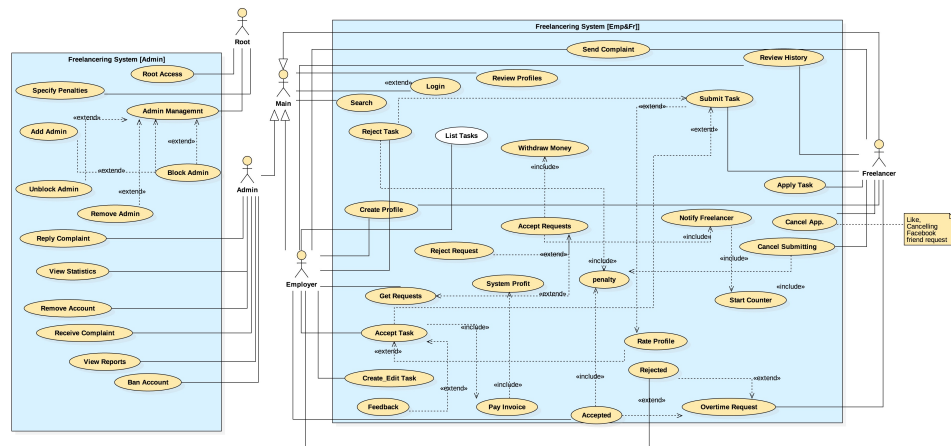
2.4 Employer

Employer are the stockholders. They also have their *profiles*. An *Employer* can create a *Task* or a *list of tasks* seen by interested *Freelancers*. Then the *Employer* can choose the best *Freelancer* among who applied. Once the *Employer* accepts the *offer* the money is *withdrawn* from his account to the pending state in the system. After the *task* is finished, the *Employer* can *rate* and *feedback* the *freelancer*, and sure the freelancer is paid.

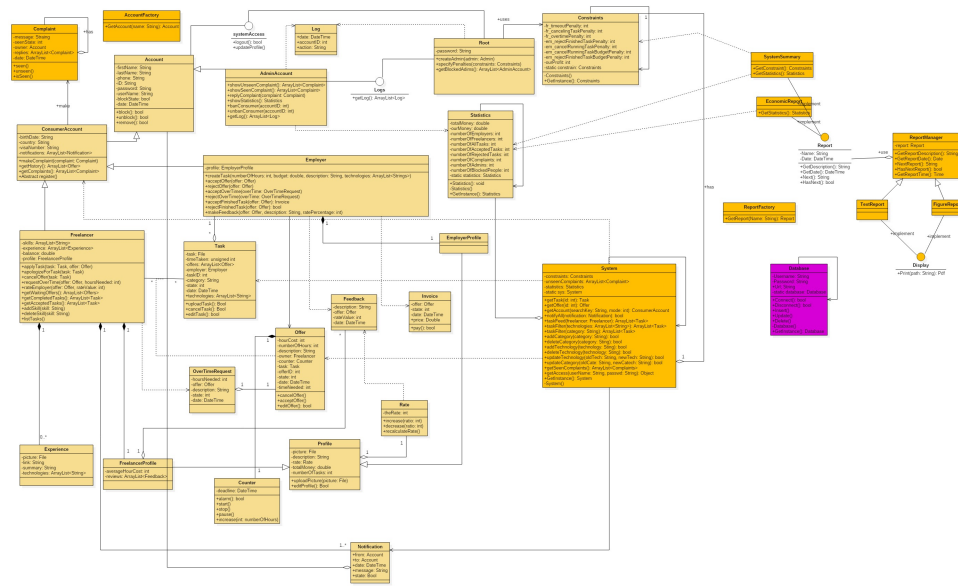
3.1 ERD



3.2 Use Case



3.3 Class Diagram



3.4 System Sequence Diagrams

Figure 1: "System Sequence of the Root"

Figure 2: System Sequence Diagram of the *Root*

Figure 3: System Sequence Diagram of the *Employer*

Figure 4: System Sequence Diagram of the *Freelancer*

4 Use Case – *in detail*

4.1 Access Root

Use Case Name	<i>Root Access</i>
Actors	<i>Root — Exclusively</i>
Pre-condition	<i>1. At configuration time for the very first use. 2. Root is given a default password.</i>
Basic Flow	<i>1. Open the System 2. Enter ‘root’ in the user name text box 3. Enter ‘0000’ in the password text box</i>
Post-condition	<i>have full control of FMS settings</i>

4.2 Specify Penalties

Use Case Name	<i>Specify Penalties</i>
Actors	<i>Root — Exclusively</i>
Pre-condition	<i>Root Access</i>
Basic Flow	<i>1. Open the system with the Root Access 2. Click on Specifies Penalties 3. Fill the Data 4. Click submit</i>
Post-condition	<i>Penalties is configured</i>

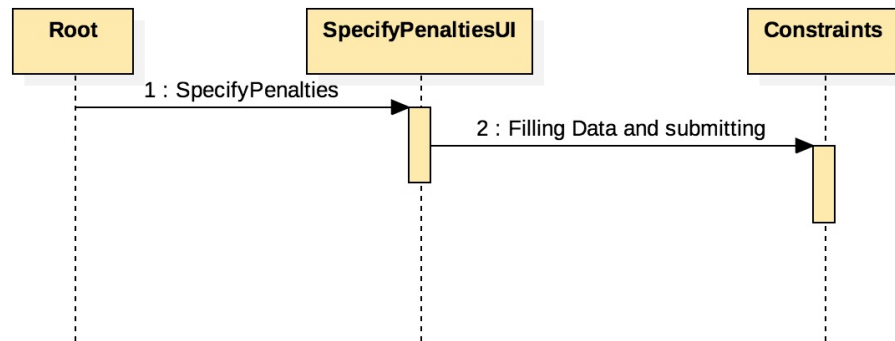


Figure 5: Sequence Diagram of the *Specifying Penalties*

4.3 Add Admin

Use Case Name	<i>Add Admin</i>
Actors	<i>Root — Exclusively</i>
Pre-condition	<i>Root Access</i>
Basic Flow	<i>1. Open the system with the Root Access 2. Click Add Admin 3. Specify its data 4. Click submit</i>
Post-condition	<i>Admin is added</i>

4.4 Remove Admin

Use Case Name	<i>Remove Admin</i>
Actors	<i>Root — Exclusively</i>
Pre-condition	<i>Root Access</i>
Basic Flow	<ol style="list-style-type: none">1. <i>Open the system with the Root Access</i>2. <i>Click Remove Admin</i>3. <i>Choose the Admin [drop down list</i>4. <i>Click submit</i>
Post-condition	<i>Admin is removed</i>

4.5 Block Admin

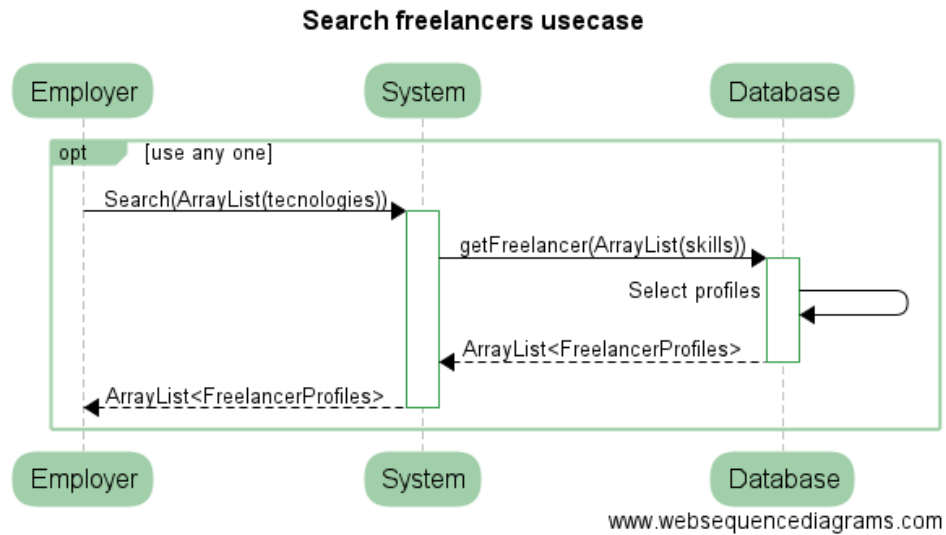
Use Case Name	<i>Block Admin</i>
Actors	<i>Root — Exclusively</i>
Pre-condition	<i>Root Access</i>
Basic Flow	<ol style="list-style-type: none">1. <i>Open the system with the Root Access</i>2. <i>Click Block Admin</i>3. <i>Choose the Admin [drop down list</i>4. <i>Click submit</i>
Post-condition	<i>Addmin is removed and is put in the block list</i>

4.6 Unblock Admin

Use Case Name	<i>Unblock Admin</i>
Actors	<i>Root — Exclusively</i>
Pre-condition	<i>Root Access</i>
Basic Flow	<ol style="list-style-type: none">1. <i>Open the system with the Root Access</i>2. <i>Click Unblock Admin</i>3. <i>Choose the Admin [drop down list</i>4. <i>Click submit</i>
Post-condition	

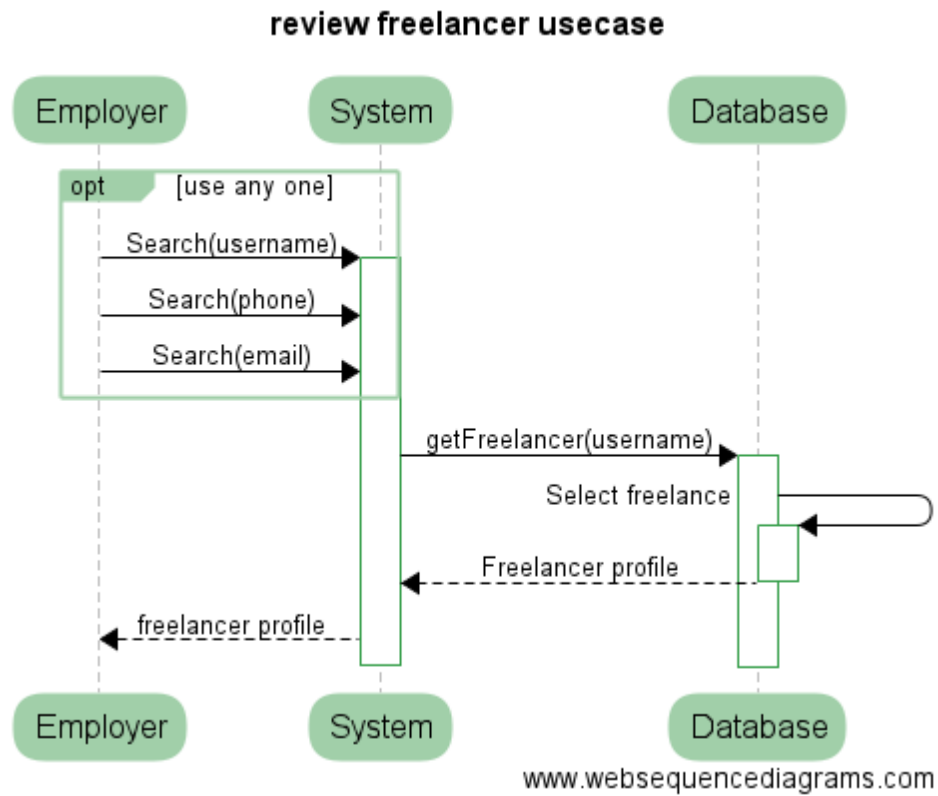
4.7 Search Freelancers

Use Case Name	<i>Search Freelancers</i>
Actors	<i>Employer — Exclusively</i>
Pre-condition	<i>Employer Access</i>
Basic Flow	<ol style="list-style-type: none">1. <i>Open the system with the Employeer Access</i>2. <i>Click Search About Freelancers</i>3. <i>set the technologies that you want</i>4. <i>Click submit</i>
Post-condition	<i>Show All freelancers that there Skills mathch with your Technologies</i>



4.8 Search about Freelancer

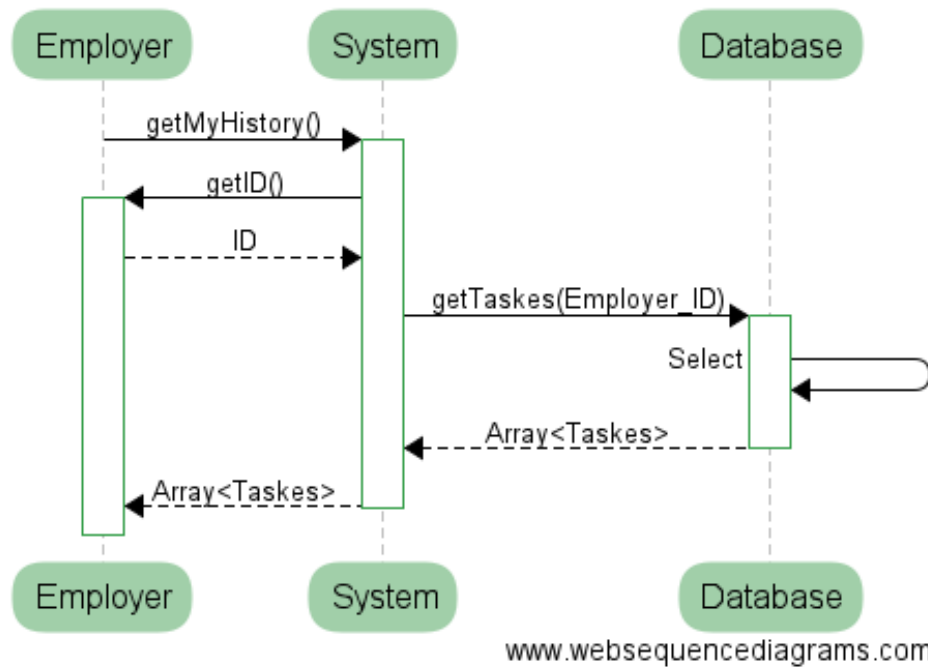
Use Case Name	<i>Search about Freelancer</i>
Actors	<i>Employer — Exclusively</i>
Pre-condition	<i>Employer Access</i>
Basic Flow	<ol style="list-style-type: none"> 1. Open the system with the Employer Access 2. Click Search About specific Freelancer 3. set his username or mail or phone 4. Click submit
Post-condition	<i>Show freelancer profile</i>



4.9 Review Employer history

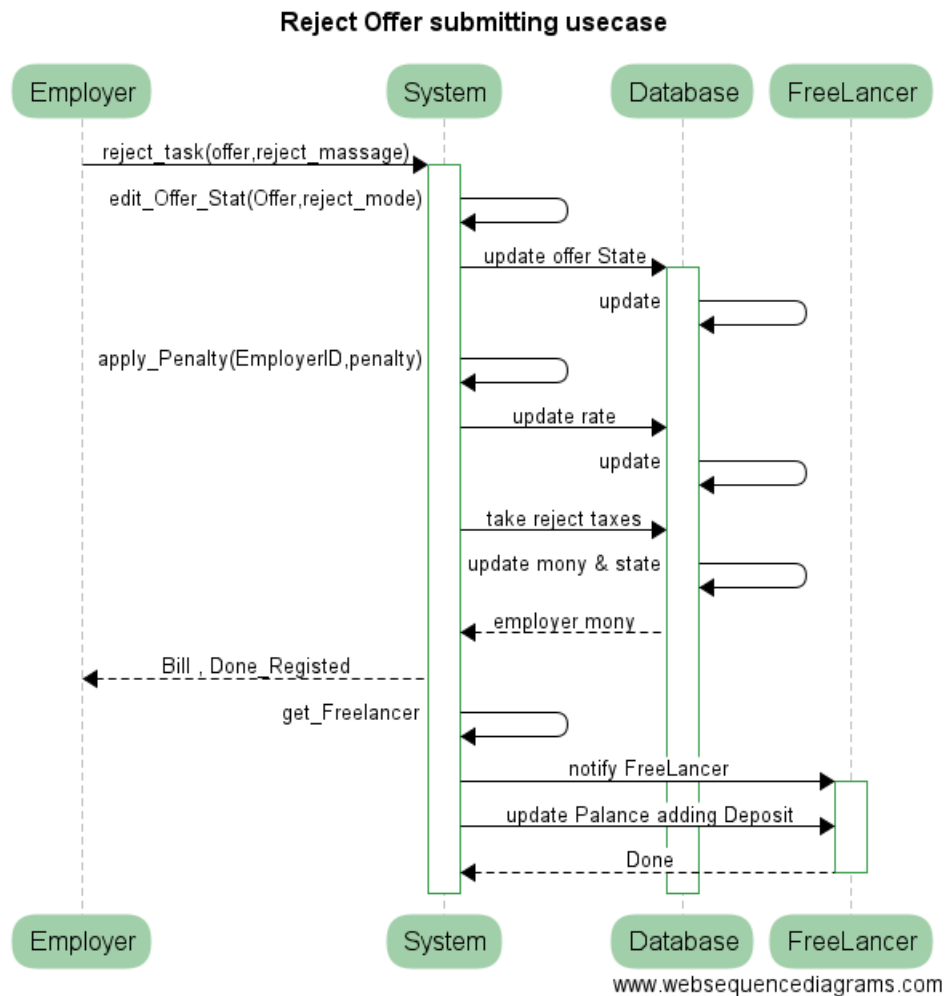
Use Case Name	<i>Review Employer history</i>
Actors	<i>Employer — Exclusively</i>
Pre-condition	<i>Employer Access</i>
Basic Flow	<ol style="list-style-type: none"> 1. <i>Open the system with the Employer Access</i> 2. <i>Click Show My history</i> 3. <i>Click submit</i>
Post-condition	<i>Show all your Tasks and offers and his modes</i>

Review history usecase



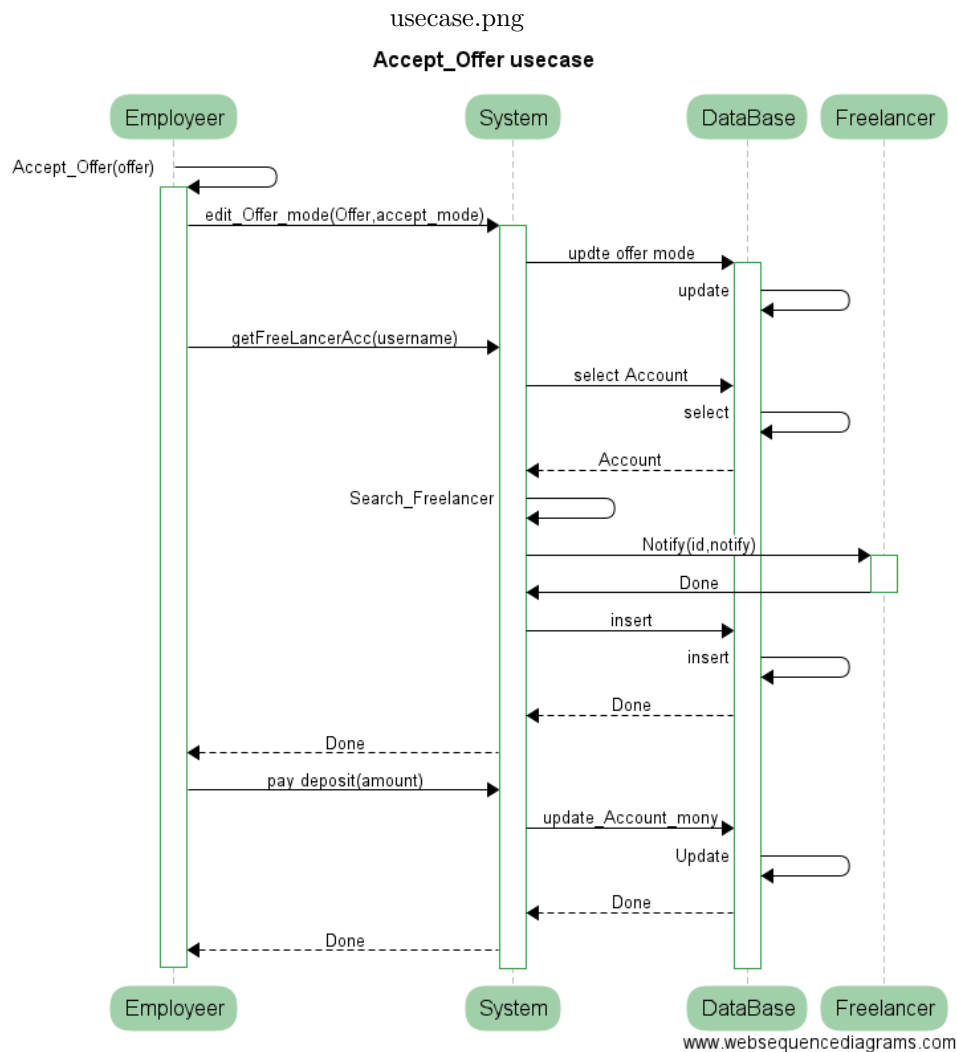
4.10 Reject offer submitting

Use Case Name	<i>Reject offer submitting</i>
Actors	<i>Employer — Exclusively</i>
Pre-condition	<ol style="list-style-type: none"> 1. <i>Employer Access</i>
Basic Flow	<ol style="list-style-type: none"> 1. <i>Open the system with the Employeer Access</i> 2. <i>Check submitting Offer</i> 3. <i>Click reject this Offer</i>
Post-condition	<ol style="list-style-type: none"> 1. <i>Take rejected</i> 2. <i>Notify Freelancer</i> 3. <i>Apply penalty</i>



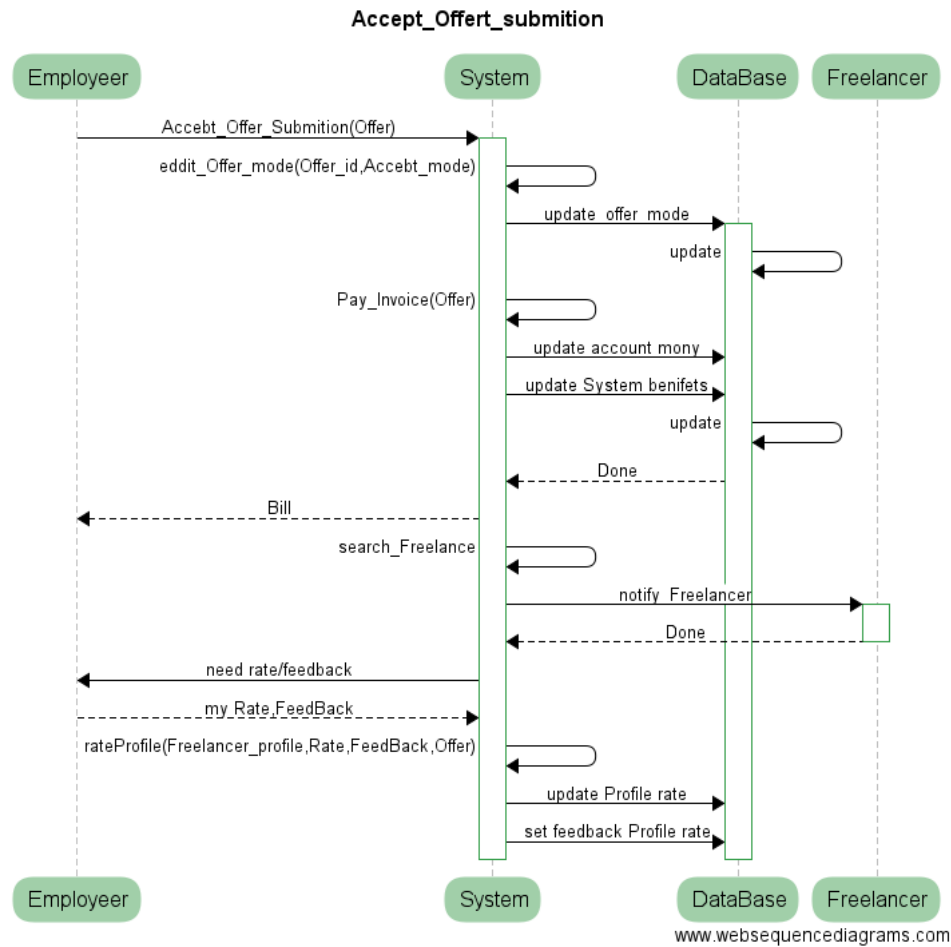
4.11 Accept Offer

Use Case Name	<i>Accept Offer</i>
Actors	<i>Employer — Exclusively</i>
Pre-condition	<ol style="list-style-type: none"> 1. <i>Employer Access</i> 2. <i>Task that have offers to do it</i>
Basic Flow	<ol style="list-style-type: none"> 1. <i>Open the system with the Employer Access</i> 2. <i>Check Take Offer</i> 3. <i>select and Click show offer</i> 4. <i>Click Accept Offer</i>
Post-condition	<ol style="list-style-type: none"> 1. <i>change Tase mode</i> 2. <i>Notify Freelancer to start</i> 3. <i>pay deposit</i>



4.12 Accept Submitting Offer

Use Case Name	Accept submitting Offer
Actors	Employer — Exclusively
Pre-condition	1. Employer Access
Basic Flow	2. Task is finish and freelancer upload his work and task wait to submit from Employer 1. Open the system with the Employer Access 2. select Offer that wait submit 3. Click submit offer and finish task 4. Pay invoice 5. Rate and FeedBack Freelancer profile
Post-condition	1. change account balance 2. re-Count to Freelancer profile 3. receive work



4.13 Reply Complaints

Use Case Name	<i>Reply Complaints</i>
Actors	<i>Admin</i>
Pre-condition	<ol style="list-style-type: none"> 1. Admin Access 2. Freelancer/Employer has sent a Complaint
Basic Flow	<ol style="list-style-type: none"> 1. Open the system with the Admin Access 2. Select Show All Complaints from the sidebar 3. Select the complaint and click to reply 4. Write your reply and submit
Post-condition	<i>A notification has sent to the sender of the commitment</i>

4.14 View Statistics

Use Case Name	<i>View Statistics</i>
Actors	<i>Admin</i>
Pre-condition	<i>1. Admin Access 2. Freelancer/Employer has sent a Complaint</i>
Basic Flow	<i>1. Open the system with the Admin Access 2. Select View Statistics from the sidebar 3. Now, You can select one of the statistics perspective</i>
Post-condition	<i>A graph is appeared</i>

4.15 Remove Account

Use Case Name	<i>Remove Account</i>
Actors	<i>Admin</i>
Pre-condition	<i>1. Admin Access 2. Freelancer/Employer has sent a Complaint</i>
Basic Flow	<i>1. Open the system with the Admin Access 2. Select Show All Freelancer/Employer from the sidebar 3. Select the one you want to removed 4. Select Remove Account 5. Confirm!</i>
Post-condition	<i>An Account it removed</i>

4.16 Receive Complaint

Use Case Name	<i>Receive Complaint</i>
Actors	<i>Admin</i>
Pre-condition	<i>1. Admin Access 2. Freelancer/Employer has sent a Complaint</i>
Basic Flow	<i>1. Open the system with the Admin Access 2. You will find a new notification</i>
Post-condition	<i>You've been notified</i>

4.17 View Reports

Use Case Name	<i>View Reports</i>
Actors	<i>Admin</i>
Pre-condition	<i>1. Admin Access 2. Freelancer/Employer has sent a Complaint</i>
Basic Flow	<i>1. Open the system with the Admin Access 2. Select View Reports 3. Select the Subject of the Report</i>
Post-condition	<i>You have an up-to-date report, you can get the PDF version</i>

4.18 Ban Account

Use Case Name	<i>Ban Account</i>
Actors	<i>Admin</i>
Pre-condition	<ol style="list-style-type: none"> 1. Admin Access 2. Freelancer/Employer has sent a Complaint
Basic Flow	<ol style="list-style-type: none"> 1. Open the system with the Admin Access 2. Select Show All Freelancer/Employer from the sidebar 3. Select the one you want to Ban 4. Select Ban Account 4. Confirm!
Post-condition	<i>An Account is Banned</i>

4.19 Login

Use Case Name	<i>Login</i>
Actors	<i>Freelancer — Exclusively</i>
Pre-condition	1. Enter Username , Password.
Basic Flow	<ol style="list-style-type: none"> 1. Open the System 2. Enter Username and Password 3. Click on Login Button
Alternative Flow	<i>If The Username or Password is incorrect: should re-enter Information again</i> <i>If The User Don't Have Account : The User is prompted to Register</i>
Post-condition	<i>Users can access</i>

4.20 Review Profile

Use Case Name	<i>Review Profile</i>
Actors	<i>Admin , Freelancer , Emplier — Exclusively</i>
Pre-condition	<i>User login successfully</i>
Basic Flow	<ol style="list-style-type: none"> 1. Open Profile Page 2. Edit and Upload Profile 3. Click on Save Button
Alternative Flow	<i>If The User Hasn't login yet: The User is prompted to Login</i>
Post-condition	<i>If The case completed Successfully, Users can see his profile and edit it</i>

4.21 Create Profile

Use Case Name	<i>Create Profile</i>
Actors	<i>Freelancer , Emplier — Exclusively</i>
Pre-condition	<i>Open Register page</i>
Basic Flow	<ol style="list-style-type: none"> 1- Fill all basic information about him First name , Last name , Job Title...etc 2. Click on Submit Button
Alternative Flow	<i>If missed any information: The User is prompted to enter it</i>
Post-condition	<i>If The case completed Successfully, Users allow to enter Home Page</i>

4.22 Review History

Use Case Name	<i>Review History</i>
Actors	<i>Freelancer — Exclusively</i>
Pre-condition	<i>User is logged Successfully ,has a Profile</i>
Basic Flow	<i>1-Open Profile Page 2.Click on View History Button</i>
Alternative Flow	<i>should be login If The User Hasn't login yet : should be login</i>
Post-condition	<i>If The case completed Successfully, User can see his History</i>

4.23 WithDraw Money

Use Case Name	<i>WithDraw Money</i>
Actors	<i>Freelancer — Exclusively</i>
Pre-condition	<i>User is logged Successfully ,has a Profile</i>
Basic Flow	<i>1-Open Profile Page 2-Click on WithDraw Money Button 3- System Show it his Balance 4- System Connect to Visa Card And add his Balance</i>
Alternative Flow	<i>should be login If The User Hasn't login yet : should be login</i>
Post-condition	<i>User can see his Balance And Deliver It To His Visa Card</i>

4.24 List Tasks

Use Case Name	<i>List Tasks</i>
Actors	<i>Freelancer — Exclusively</i>
Pre-condition	<i>User is logged Successfully</i>
Basic Flow	<i>1-Open Profile Page 2-Click on List Tasks Button</i>
Alternative Flow	<i>should be login If Hasn't login yet : should be login</i>
Post-condition	<i>See All Tasks</i>

4.25 Apply Task

Use Case Name	<i>Apply Task</i>
Actors	<i>Freelancer — Exclusively</i>
Pre-condition	<i>User is logged Successfully</i>
Basic Flow	<i>1-Open Profile Page</i> <i>2-Open Tasks Page</i> <i>3- Choose Task , Specify hour cost and the number of hours and all info</i> <i>4- Click on Apoly Task Button</i>
Alternative Flow	<i>should be Login If Hasn't login yet : shoul be Login</i> <i>If There aren't Any Tasks : Refresh Page</i> <i>If user enter number of hours greater than specified in task :</i> <i>should be enter number of hours less than specified hours.</i>
Post-condition	<i>If The case completed Successfully, User send request to do this task</i>

4.26 Cancel App

Use Case Name	<i>Cancel App</i>
Actors	<i>Freelancer — Exclusively</i>
Pre-condition	<i>User is logged Successfully</i>
Basic Flow	<i>1-Open Profile Page</i> <i>2-Open Tasks Page</i> <i>3- Choose Task</i> <i>4- Click on Cancel Task Button</i>
Alternative Flow	<i>should be Login If Hasn't login yet : should be Login</i> <i>If Emplyer Accept request:</i> <i>to finish the task or Cancel then System will penalty cut of 4%</i>
Post-condition	<i>If The case completed Successfully, User Cancel request to do this task</i>

4.27 Overtime Reguest

Use Case Name	<i>Overtime Reguest</i>
Actors	<i>Freelancer — Exclusively</i>
Pre-condition	<i>User is logged Successfully , Task's Reguest is accepted</i>
Basic Flow	<i>1-Open Offer Page</i> <i>2-Enter overtime hours</i> <i>3- Click on overtime request Button</i>
Alternative Flow	<i>The penalty cut of 7% If The Task's time finished : The penalty cut of 7%</i> <i>The penalty cut of 7% If Emplyer reject request: The penalty cut of 7%</i>
Post-condition	<i>If The case completed Successfully,</i> <i>Overtime add to counter , will not take money for them -</i> <i>and 2% will be cut from his rate</i>

4.28 Submit Task

Use Case Name	<i>Submit Task</i>
Actors	<i>Freelancer — Exclusively</i>
Pre-condition	<i>User is logged Successfully , Task is finished</i>
Basic Flow	<i>1-Open Offer Page 2-Upload Task's files 3- Click on submit Button</i>
Alternative Flow	<i>If Task's Time finished : penalty cut of 7% ,User Can't submit task and. If user upload wrong files : The User is prompted to upload correct files</i>
Post-condition	<i>If The case completed Successfully, Task is submitted , User will get nofication and Earn Money .</i>

4.29 Cancel Submit

Use Case Name	<i>Cancel Submit</i>
Actors	<i>Freelancer — Exclusively</i>
Pre-condition	<i>User is logged Successfully , Task is submitted</i>
Basic Flow	<i>1-Open Offer Page 2- Click on Cancel Submit Button</i>
Alternative Flow	<i>If Task's Time finished : System will make panlty</i>
Post-condition	<i>cancel Task'submission , System will make panlty</i>

4.30 Rate the employer

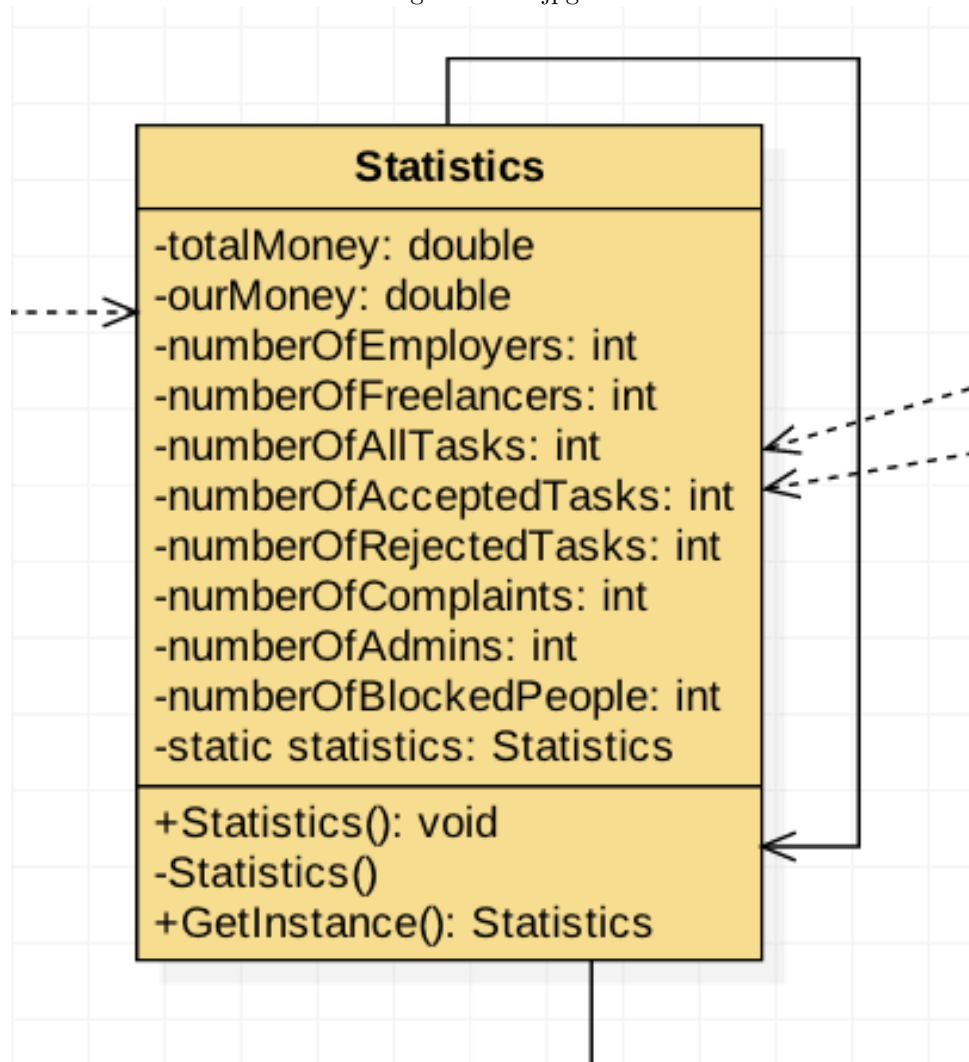
Use Case Name	<i>Rate the employer</i>
Actors	<i>Freelancer — Exclusively</i>
Pre-condition	<i>User is logged Successfully</i>
Basic Flow	<i>1-Open emplyer profile 2- make rate 3- Click on rate Button</i>
Alternative Flow	<i>If The User Hasn't login yet : The User is prompted to Login</i>
Post-condition	<i>User make rate</i>

5 Design Patterns

5.1 Singleton Design Pattern

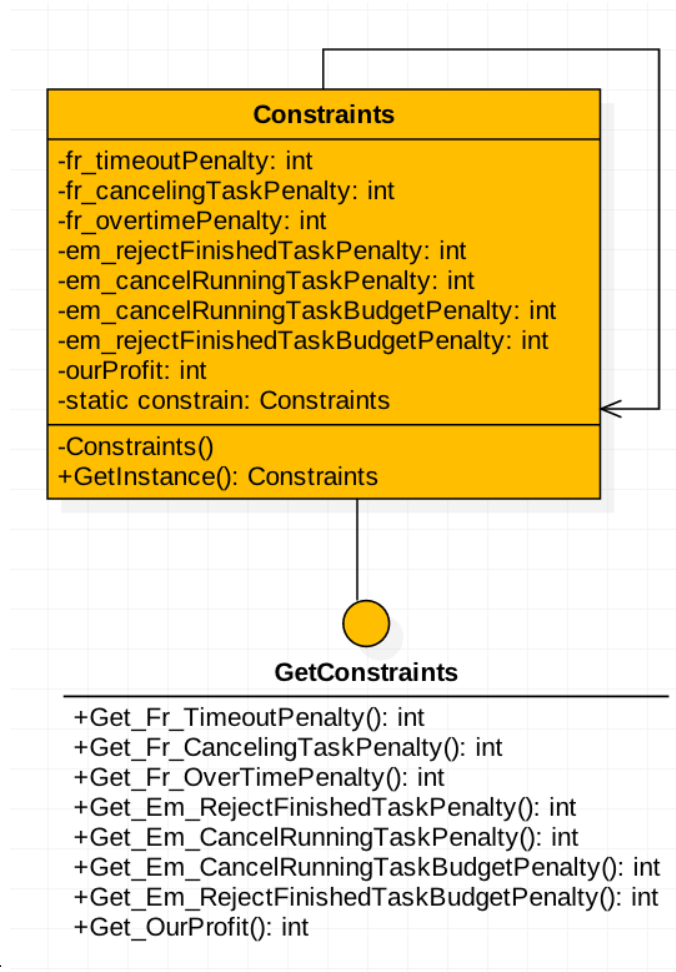
Context	We need to get only one instance (Statistics/Database Class...)
Problem	How can we insure that it is not possible to get more than one instance of a Class
Solution	1. <i>Make a the constructor private.</i> 2. <i>Make public static method, It creates single object.</i>

Design Pattern.jpg



5.2 ReadOnly Pattern

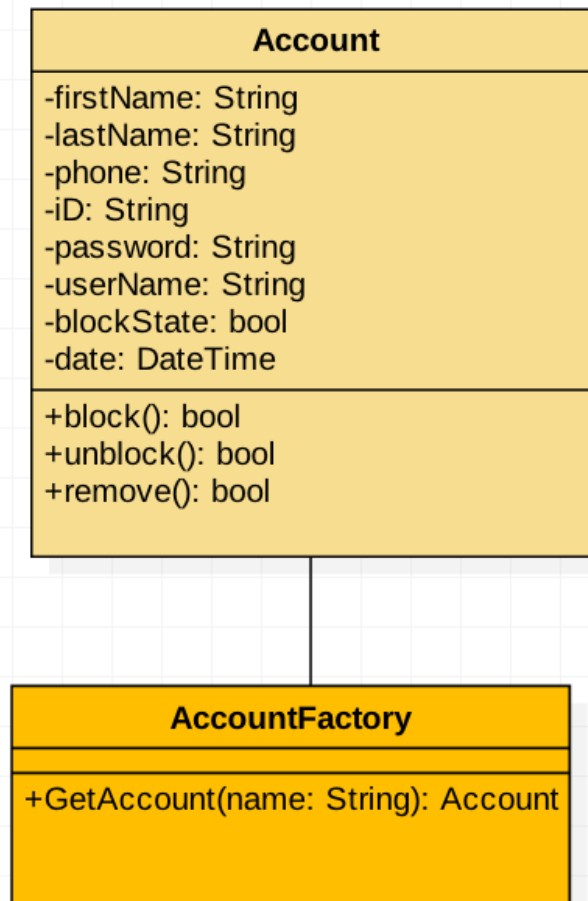
Context	We need an object to be read-only (We cannot edit it) [Statistics].
Problem	How do you create a class whose instances are immutable.
Solution	1. Make an Interface 'Read-Only' with getters methods.



Design Pattern.png

5.3 Factory Pattern

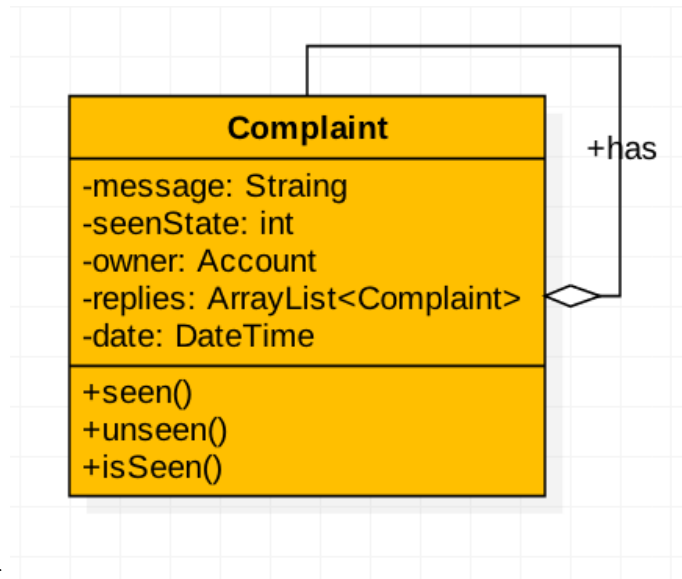
Context	We need to generate objects as a template (automatically).
Problem	How to generate objects using a generating method (Factory).
Solution	<i>Make a class (Factory), that contains a method which returns, an object of the wanted class</i>



Design Pattern.png

5.4 Composition Pattern

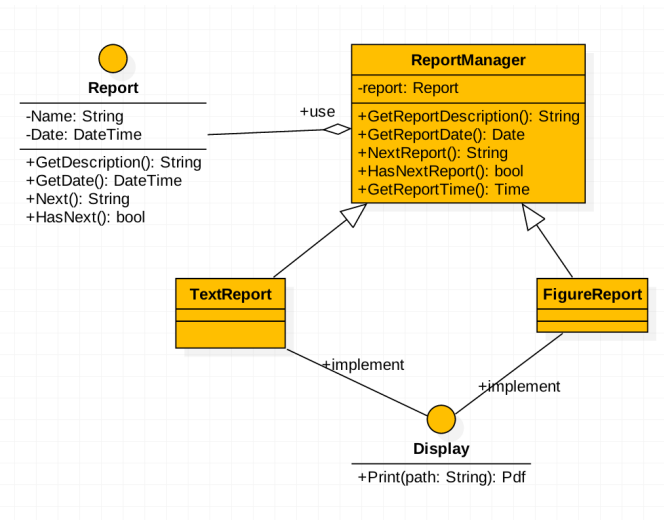
Context	Recursive composition, Something that can contain it self.
Problem	How can we Enable the complaint to contain a reply (<i>Which is a complain</i>).
Solution	<i>It can be done by self-composition</i>



Design Pattern.png

5.5 Bridge Design Pattern

Context	We need abstraction, and it can be implemented in different ways.
Problem	We have <i>Report class</i> , it has an abstraction, but there are different subjects of a Report.
Solution	<i>Make a Report Class, that implements an interface (abstraction)</i> <i>All different types of the report are children of the Report</i> <i>Each of them implement the abstraction in its own way</i>



Design Pattern.png

6 Call Us

THE END