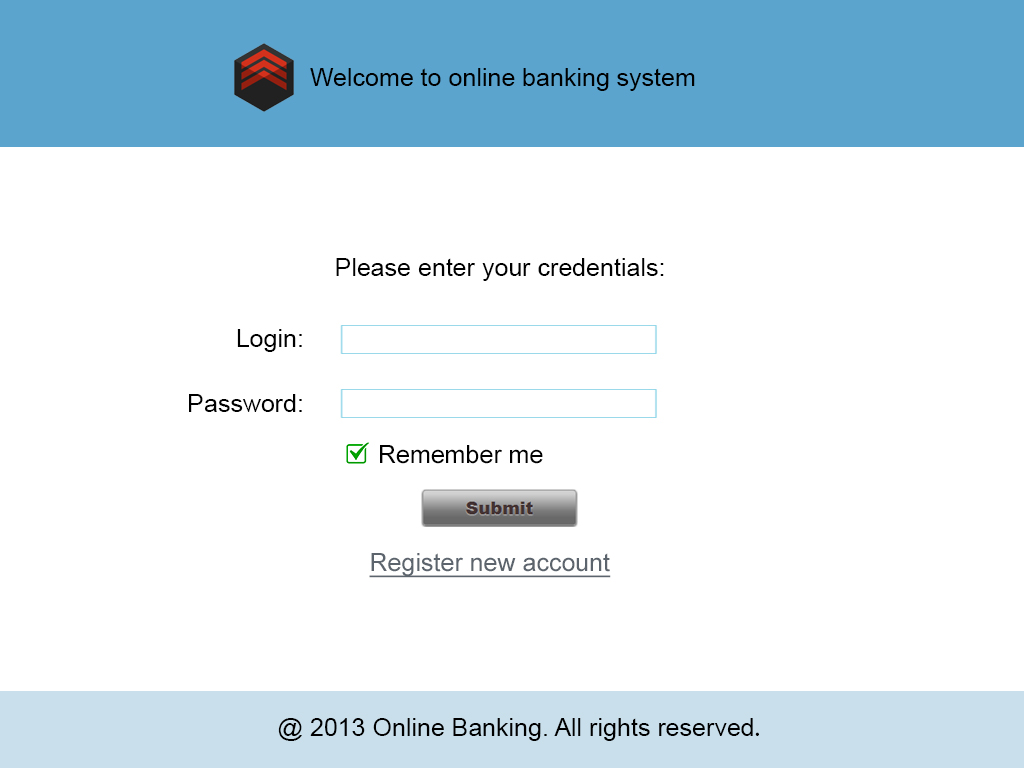
Application “Online Banking for Managers”

Goal of application: Show clients information for managers in nice view.

Overall project requirements:

1. Pages should fit in 1024x768 resolution
2. Page must be stretch for whole browser window
3. Font: Arial for all texts
4. All code must be covered with unit tests at least 80%(NUnit Framework, MOQ)
5. Application should not crash with errors at any situation even if something doesn't work
6. Most of Microsoft rules must be done like: Naming conventions, readability, comments conventions, styling conventions, etc. Nice tool which can help: Resharper.
7. All Autofac services must be checked with CodeContracts.
8. Applications type: MVC 4 – Razor engine for views – VS2012
9. Latest updates for all required libraries like jQuerry, etc…(using Nuget package manger)
10. All css styles must be done using "less" stylesheet language
11. Browser compatibility: IE8, 9,10, latest versions of Chrome, Firefox.

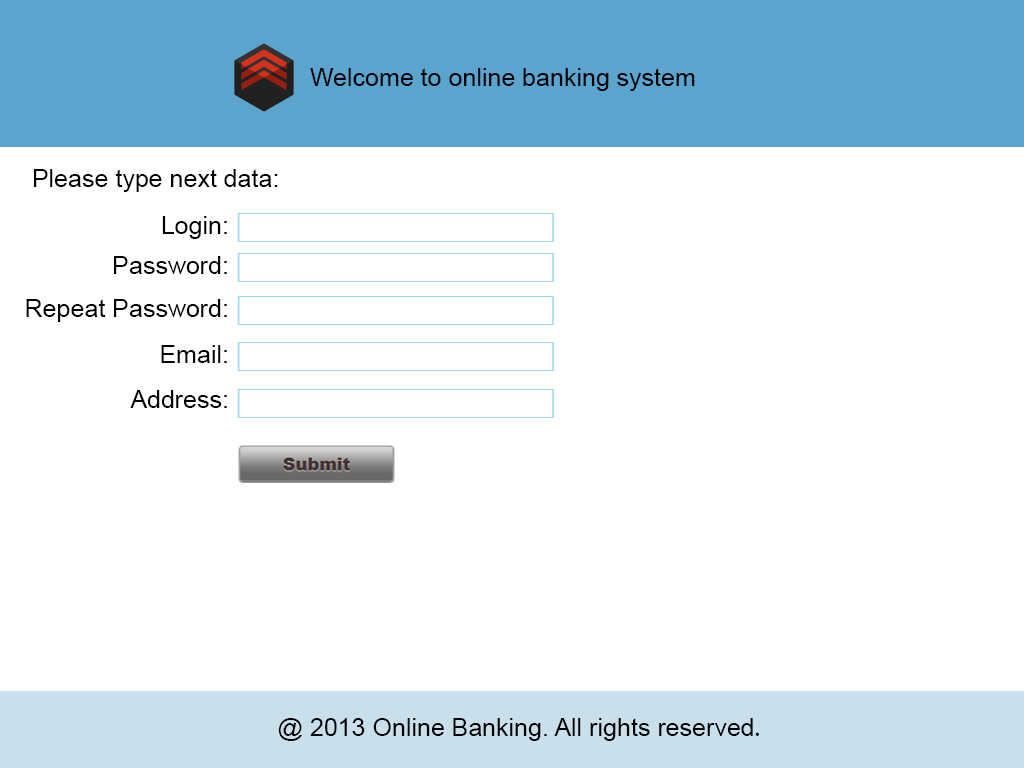
1. Login page approximate mockup:



Requirements:

* 1. Year in footer must be current year
  2. Selected text box should be stressed up with different style
  3. Validation must be done on client side as well as on server side
  4. Appropriate login errors must be shown for users in nice view like in red color, etc... Try to handle all possible errors like "Technical error" - in case of errors, "Service unavailable. Please try again later" - in case of service error, "Login or Password is incorrect", etc.
  5. Login functionality should be done at "login" service and application shouldn't know how it works at all. You just need to call appropriate service and pass login/password to it (using Autofac).
  6. In case of errors from service need to log it using log4net
  7. If for existing account typed more than 5 types wrong password then application should block this account(and show "blocked" error for further login attempts) and send email to account mail address with blocking information and link to page with ability to unblock it(simple page is fine).
  8. Blocking account should be logged by log4net
  9. Emails service should be done separately using autofac (passed parameter like recipient, subject, body, etc) – it should works using any free SMTP server.
  10. Login service itself should work with database for example using linq to sql(usual MSSQL is fine) or membership provider
  11. Clicking on logo/logo text should redirect you to home page of site
  12. Remember me should work ok

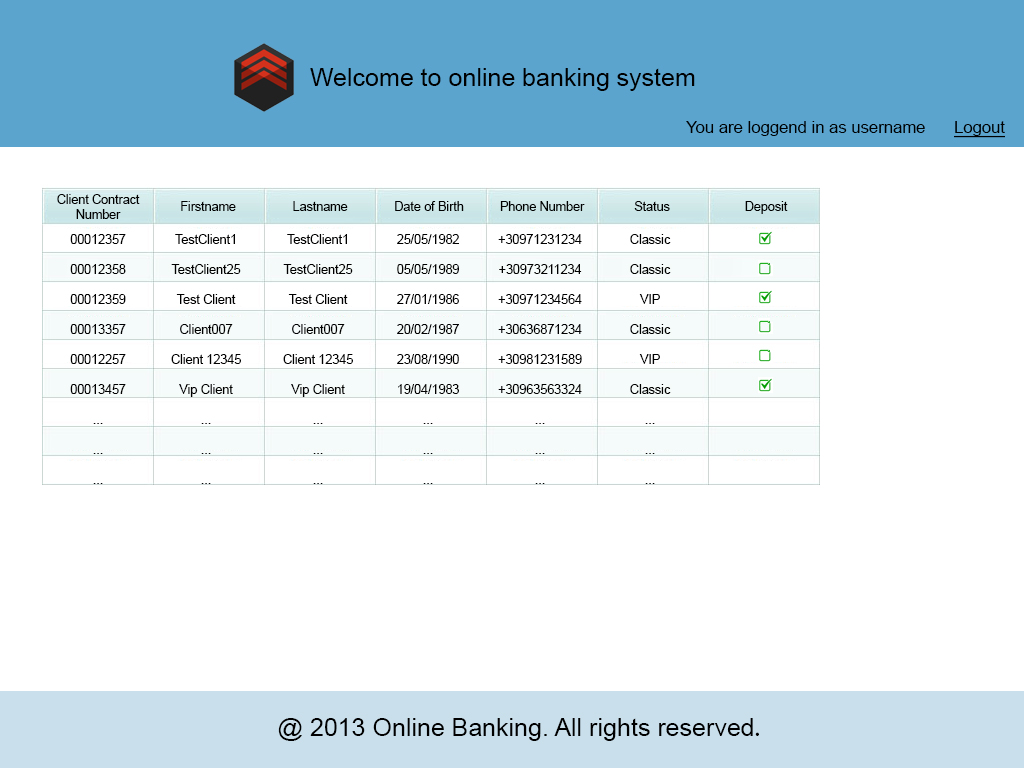
2. Register page approximate mockup:



Requirements:

1. Clicking on logo/logo text should redirect you to home page of site
2. Registering should be done using “Register” service (using Autofac)
3. Selected text box should be stressed up with different style
4. Validation must be done on client side as well as on server side
5. In case of errors from register service need to log it using log4net
6. Address field is optional, all other fields - required.
7. Email field should valid
8. Login/Password fields must be at least 8-16 symbols in length
9. Password/Repeat password should match
10. Appropriate register errors must be shown for users in nice view like in red color, etc... Try to handle all possible errors.
11. Tooltips must be done for all fields with descriptions of field and errors text(qtip plugin)
12. For correctly typed value near field must be places green check. Otherwise – red check.
13. Successful registering should send confirmation email to user email address. (simple confirmation is ok)
14. Additional validation like “Captcha” would be nice

Homepage for logged in users:



Requirements:

1. This page can see only authorized users
2. Username in upper right should be replaced with real login name
3. Logout should work ok
4. Table should be sortable
5. Clicking on any client row should open edit client page with all data(simple page with textboxes, dropdowns, checkboxes are fine – all same here regarding Autofac service and validations)
6. Create new client could be same like edit page
7. Paging must be implemented with ability to easy change configuration of it
8. Try to implement filters for table (for example show only VIP clients)
9. Add print button which will print only current result
10. Add to table “Delete” button (after pressing need to show confirmation)
11. Try to add summary information at bottom: Total count of clients, Count of VIP clients, Count of Classic clients, Clients with Deposit.