SOFTWARE REQUIREMENTS SPECIFICATION

for

Internet banking system

Version 1.0 approved

Prepared by Wangzhihui Mei

CCNU-UOW JI

January 17, 2021

Contents

1 Product Constraints

1.1 The purpose of the product

The purpose of the Internet Banking System is to ease bank account management and to create a convenient and easy-to-use application for users, trying to perform money transferring. The system is an online system for bank business providing feasible and comfortable user experience for user.

1.2 The client, customer and other stakeholder

The customers are the existing offline bank customer who have bank accounts, either saving account or credit account. They can access their bank accounts via a browser or a mobile App using this system. Other stakeholders include bank service provider, financial institution and public security department. The bank set up servers supporting business functions. Financial institution can conduct financial supervision for bank money transfer. Public security department can combat money laundering and fraud crime.

1.3 Users of the product

Users are the existing offline bank customer who have bank accounts, either saving account or credit account, who can login to bank account, check account balances, transfer money and logout the account.

1.4 Requirements constraints

The concurrency of the backend server is limited. When the server support more than 10000 users concurrently, it may be down and stop service. There also be some delay as server need some time to process business computation. The internet down or the browser or the mobile App crashes may interrupt the function of the system.

1.5 Relevant facts

The internet bank service has been deployed worldwide. As the mobile internet era comes, more and more users tend to use their mobile App to access their bank account as it is more convenient and easier to manage. The online banking system are typically connected to core banking system. The trend of the service online is irreversible.

1.6 Assumptions

The assumption include several perspective. From the client perspective, user can access their account by mobile terminal and conventional PC terminal, they are free to login to their account and perform money transfer. From the server perspective, the backend system should be available for 24 hours per day and 7 days per week.

2 Functional requirements

2.1 The scope of the product

The purpose of the Internet Banking System is to ease bank account management and to create a convenient and easy-to-use application for users, trying to make money transferring. The system is based on client-server architecture. There will be some backend server supporting all business functions and user can access their bank accounts via a browser or a mobile App. The goal of the system is to make online banking available. follow the online t Above all, the product can provide a comfortable user experience for users.

2.2 Functional and data requirements

2.2.1 Login to the online bank account

Customer can login their account by inputting the account number and password and press "login" button for authorization. If the authorization is valid, the user can access detail of his account or making money transferring. Otherwise, if the authorization is invalid, the user should re-input the account number and password. The account will be locked for half an hour if the authentication of account number or passwords failed 3 times.

2.2.2 Check account balances

Customers can check their the balances of their accounts. If the account is saving account, the balance should be no less than 0 dollar. Otherwise, if the account is credit account, the saving should be in a range less than overdraft limit.

2.2.3 Money transfer

The customer can transfer money from his account to another. If the target account is his/her own, there will be no limit except for the balances requirements. Else, if the target account is another person's, the customer should receive a security code from system and input the security code via the client terminal within 1 minute. If the security code is consistent with the received one, the money will be transferred. There will be both daily limit and balances requirements for the transferring.

2.2.4 Logout the online bank account

The customer can logout his/her online bank account by clicking "logout" button, then the browser or the mobile app will return to the home page. The browser or mobile app should terminate the login session and clear any data related to the login credentials.

3 Non-functional requirements

3.1 Look and feel requirements

As there are browser client and mobile App client, the user interface of different clients should be designed separately. But all clients should share the same design theme from a usability standpoint. There are generally following points should be concerned:

- Apparently simple to use

- Approachable, so that people do not hesitate to use it

- Authoritative, so that users feel they can rely on it and trust it

- Conforming to the client's other products

- Attractive to children or some other specific group

- Unobtrusive, so that people are not aware of it

- Innovative and appearing to be state of the art

- Professional looking

3.2 Usability requirements

With the aid of instructions for use, each requirement must be met by all adequately trained users on the first attempt to reach established usability goals. All observed errors shall be responsive, intuitive and user-friendly. For appropriate users, they must:

1) Operate with quick and accurate operation feedback.

   When user click the UI button, there should be some progress indicator tool such as progress bar or loading icon to relieve users' waiting anxiety. If they succeed in using some features, they should get quick response and be redirect to the new state page, else, the failed message or warning page should be displayed.

2) Important information must be clearly displayed.

   The login session should be intuitive. Other important information including the balance of the account, the credit repayment date and the history of money transferring should be clearly available on the home page.

3) The redirection of UI page should be quick.

   The UI response time should be less than 200 milliseconds with some animation.

3.3 Performance requirements

The internet banking system service should be available for 24×7 (24 hours per day, 7 days each week). The maximum number of concurrent users should be over 10000 should be supported by the system with less than 1000 milliseconds delay. The backup system should be activated and take over the internet banking system within 10 minutes when the original system is down.

3.4 Operational requirements

If client disconnection happen due to internet down or browser/mobile App crash, all uncompleted money transferring should be cancelled. The operation should be atomic and consistent. The operation history should be informed to users and the crash log should be kept for analysis of developers.

3.5 Maintainability and portability requirements

The product shall follow modular design pattern to be able to be modified to cope with anew class of use. Besides, the client should be portable to all kinds of browsers(Chrome, Firefox, Safari and so on) and all mobile platforms (iOS and Android).

3.6 Security requirements

All data transferring should be encrypted. The login session should be terminated after certain idle time or expired. The data must satisfy confidentiality, integrity and availability.

3.7 Cultural and political requirements

The product must support multi-language switching and support instant translation service and perform product localization. The data should be kept on the demand of which country the service is deployed.

3.8 Legal requirements

The system should be committed to the laws by examining the legislation or asking a local lawyer. The service should be authorized by the local government.

4 Project issues

4.1 Open issues

There are generally some open issues:

1) The data synchronization between original system and backup system.

2) The high concurrency issues

4.2 Off-the-shelf solutions

The master-slave deployment can solve the data synchronization problem between original system and backup system. We can deploy message queue service and elastic compute service to solve the high concurrency issues.

4.3 Tasks

There are generally following tasks:

- Requirement review
- UI design
- Client terminal design
- Server terminal design
- Service deployment
- Service maintenance

4.4 Risk

The lack of traceability and inadequate validation may be the risks. Some features cannot be mapped to business requirements. Requirements may not be validated by the appropriate subject matter experts.

4.5 Costs

The time needed for development and deployment is 6 months. The total