Working with BLOB

To get the KYC documents of the customers we worked with the BLOBs i.e. to save an image into the database we made use of these blobs. The customer are required to complete their KYC (Know Your Customer) in the case when they want to open a new savings account in our portal. This KYC requires some documents to be uploaded from the customer. But if the customer is already having an account in other banks we can ask for that ekycno and check with that bank for the status of KYC as well using the web services.

In case customer don’t have any account and is applying for a new savings account in our portal then the customer is asked to complete his/her KYC. The KYC portal will take the details of the customer like pan number, pan photo, aadhar number, aadhar photo, passport number, passport photo and display photo. Once the user has uploaded the required documents the approval status if set to pending as the approver person or admin has to verify and approve the KYC documents. Once the approver opens their portal and verify the details they have the option to approve or reject the KYC of the customer.

To store and retrieve the images we created a form to allow the user to browse to the files path so they can upload the documents. Then to handle multiple files coming from a single HTML widget we need to make use of file upload class. This class can handle multiple images coming to the page and can store them as well. Then to acquire the contents of the html form we created a list of FileItem and we parse the request to do so. It created a list of fileitem which has all the data stored in it. The content was fetched individually using a get method and each element of the form is taken separately to establish a connection to database and store it over there. The get method works on index, which is nothing but the reference to the html form we created to take the content from users. By using those indexes we can fetch all the content individually and store them into the database. This whole functioning is covered in the AddPhoto Servlet.

Once the user has uploaded the documents required for KYC verification they have to wait for approval status from the admin. The user can also view their documents whatever they have uploaded to the database. To retrieve the images from the database we used two different servlets. ListPhoto servlet is used to display the content to the user. Here we established a connection to the database and we used the img tags to display the images. The href attribute in img tags will take us to DisplayPhoto Servlet which will fetch that particular column image at a time and return to ListPhoto Servlet to show the image to user.

Web Services

We are working to integrate the other banks into our web portal. The other banks will have their resources on their servers. Whenever the user wants to link different bank account we will be needing the web services to see whether the customer has verified kyc in that bank and then we can fetch the balance from that account and display it on our portal. So, they don’t have to remember all the passwords, only one portal can do all this thing for them. So here there are client and resources concept. The other bank have their resources on their server and our portal will make a request to get particular resources from their server. This mechanism will be deployed by using REST API. REST stands for Representational State Transfer i.e. When RESTFUL API is called, the server transfers a representational state of required resource to the client.

We created a dynamic web project for HDCC bank and INFI bank and run it on the server so that whenever a call comes from the client it can handle and process that request easily. The resources that other banks can access is that whether that customer exists with an ekycno or not and if that customer exist the other banks just ask for the balance from their accounts.

By adding this web services we make it more secure as the other banks will not provide their whole data. They can agree to provide only small part of information to the client. The client needs to specify the url for the resource to fetch and some parameter as well to retrieve only particular information. The other banks returns the json object once the call is successfully handled and then the client can read that and use the values inside that object accordingly.

REST Services