Webapps File Catalogue

# Web Pages

## Index.xhtml

Main page of the application, contains links for registering user accounts, go to the user page, go to the admin page and log out the current user which is also displayed

## Login.xhtml

Page used so both the admins and users can log in

## Registration.xhtml

Page used by new users to enter all personal details such as email, password, first name and last name. On creation £1000 is put in their accounts

## Admin.xhtml

Page only accessible by user ins the Admin User Group. Allows admins to add more admins in the same fashion as the registration page, view all users and view all transactions

## User.xhtml

Main deposit page allowing users to send and request deposits aswell as accept or reject any deposits sent to or requested of them. Users are also able to view any transactions relating to them as well as their balance

## Error.xhtml

Page used when wrong username or password is entered in login

# EJBs

## DepositService

EJB that enacts the functionality of making or requesting deposits as well as accepting and rejecting them. This EJB also calls on the queries that gets deposits and balances to be viewed by the appropriate pages as well as checking balances whilst the deposits are made.

## IntSingleton

EJB that is called when the Payara server first starts that creates the admin1 user

## UserService

EJB that handles the functionality of registering users and admins as well as the functionality for retrieving all users so they can be displayed

# Entities

## SystemDeposit

Entity of system deposit is used to create deposit entities with the attributes of date, amount, currency, fromName, toName and status. Also contains the getter and setters for each of these attributes and the constructor for making a new deposit which is used by the deposit service. To Name and From Name are meant to be User entity types but this was notice too close to the deadline to risk changing and refactoring this

## SystemUser

Entity of system user is used to create new users with the attributes of id, username, user password, name, surname and balance. Username is meant to be called email but having it called username still serves the same functionality. The class also has the getters and setters for each of these attributes as well as constructor for making a new user which is used on registration and by the admin page.

## SystemUserGroup

This entity enacts the SystemUserGroup class with an ID, username and group name attributes which correlates with a particular user to give them a suer group. This is what is checked when a user or admin tries to login to sections of the application to see whether they are authorised. This also contains the getters and setters for these attributes as well as the constructor for a user group.

# JSFs

## DepositBean

Handles the data received from the main deposit page by passing it to the Deposit Service EJB. This is done when the submit or request button is pressed as well as the accept and reject buttons.

## LoginBean

Bean used for login services by retrieving data from the login webpages and passing it to the servlet for authentication

## RegistrationBean

Bean used for getting all of the data from the registration .xhtml page and passing to the User Service EJB.

# RSE

RSE functionality was limited due to an issue where the server refused to start the service.

After taking the late period to fix it I was unable to find a resolution in time, therefore it was commented out and a hard code solution was implemented in its place.

Check RSE files and bottom of deposit bean for info

## Currency

Class used by the RSECurrency file to store a currency

## RSEApp

Registers RSECurrencies so it can be used by the RSE

## RSECurrency

File called when deposits are requested or accepted to handle conversion rates for the deposits