Foreclosure

DB Connection strings & URLs

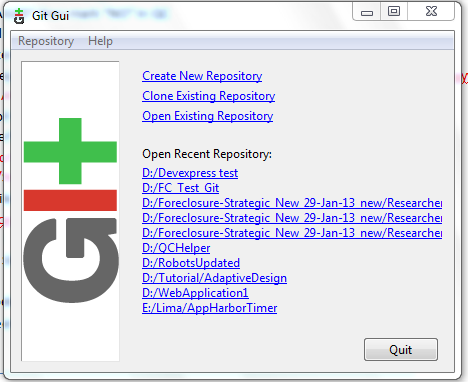
* Live :
  + Server=0c8e3fb0-75c5-4bd6-9bcf-a444017da4de.dedicatedsqlserver.sequelizer.com
  + Database=db0c8e3fb075c54bd69bcfa444017da4de
  + User ID=idhlmdhbpntdvtgu
  + Password=8zETUMBEHoGfBFnnrY2JGcFgYUy8gA8DVLcVjJMPNU3oGmHdsEXb4Np63x48oCyg
* Test:
  + Server=2c497a89-11c6-45e7-86ec-a19a00b1adc7.sqlserver.sequelizer.com
  + Database=db2c497a8911c645e786eca19a00b1adc7
  + User ID=pumditnpfesvsrvf
  + Password=5TTbKDsGs3pnB4DN3uKQNYDYaysDV5bJ6uwhAFkfE5WzDtirNd6wfjEep3ammVTv
* Amazon S3 credentials
  + URL: https://strategicla.signin.aws.amazon.com/console
  + Account : strategicla
  + Username :developers@jonesen.com
  + Password :r#uj{XqUKg&&
* Github credentials
  + Live:
    - UserName: robinexalt
    - Password: exaltrobin100
  + Test
    - UserName: limaexalt
    - Password: exaltlima100

QC Creation

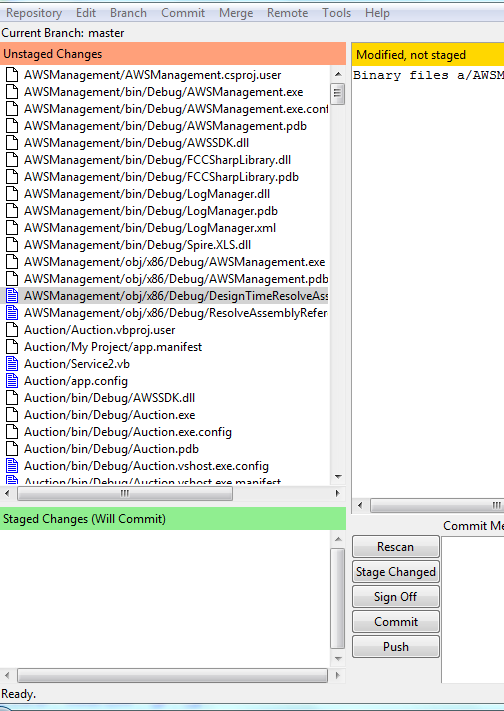
* QC Helper URL: <http://qchelper.apphb.com/>
* Visual Indicator URL: <http://foreclosure.apphb.com/Admin/VisualIndicator.aspx>]
* SP for QC: QCvaluesforthedate'<date in mm/dd/yy> format’ Eg: QCvaluesforthedate'06/25/2015'
* Sp for Getting the temp table details: spgettemptablecount'<date in mm/dd/yy> format’ Eg: spgettemptablecount'06/25/2015'
* SP for getting log data : QCLog'<date in mm/dd/yy> format ' Eg: QCLog'06/25/2015'
* QC Creation Procedure
  + Morning Rundown
    - Execute the SP QCvaluesforthedate , note down the count values and enter to the space provided in the QC file
      * + All Robots Functioning
    - Check whether the robot code is working, for that check logs are created by each robot in the current time
  + Visual Indicator Functioning
    - Check the visual indicator page, for each robot check the “connection” if the value is showing “FAILED” then mark “NO” in QC
  + CSV Files Created
    - Check Amazon S3 repository & for each robot check a csv file is created for current day
    - AmazonS3 repository: All Buckets /Strategic/CSV/<robotName>/<date in the mmm-dd-yyyy format> eg: All Buckets /Strategic/CSV/Auction/Jun-25-2015
    - In the case of migration
      * Check the files are created in the repository All Buckets /Strategic/CSV/MigratingDataToRetRequred/<date in mmm-dd-yyyy format> Eg: All Buckets /Strategic/CSV/MigratingDataToRetRequred/Jun-25-2015
      * Check a file for the robot is created for the current date <RobotName>\_Iteration \_<iterationNumber>\_<fileNumber>.CSV Eg: Auction\_Iteration \_0\_3.CSV
  + Log Files Created
    - Execute the SP QCLog and check for “YES/NO” values against each robots
  + Backup Check
    - Enter last code upload date
* No need to fill other worksheets

Code upload to github

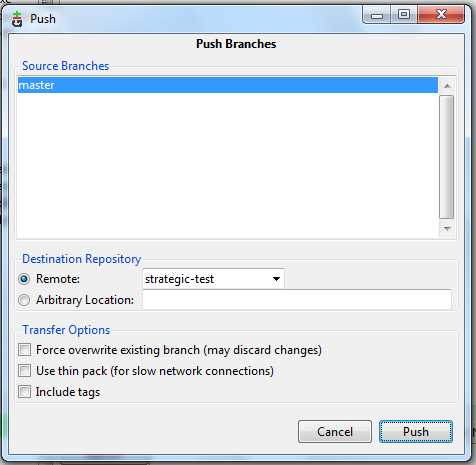
* Git GUI



* Using this UI we can create,clone,or open a git repository
* After creating a repository the recent repositories will be shown below “open recent repository”
* Click the corresponding path the following page will be seen



* In the left the changed file will be listed, if the file is clicked, on the right pane the changed portion will be shown
* Click on the icon on left side of the files we need to upload, the selected files will be moved to bottom left side under ” staged changes”
* A commit message should be provided before uploading the changes in the space provided (bottom right)
* Rescan: scans the repository again
* Staged changes: Selects all changed files
* Commit: commit the changes to the git repository
* Push: Upload the changed files to Github & Appharbor
* When push button is clicked the following window will be displayed



* Remote: shows the git repository
* Arbitary location: paste the repository url from appharbor
* Force overwrite: if we done commits multiple times & has not performed push operation this option is checked inorder to take the latest changes
* When push button is pressed the windows for entering the username & password will be shown, just enter those data the, files will be uploaded to the corresponding repository

Project Details

Contains two parts

* Web
* Robots

WEB

* Live URL: <http://foreclosure.apphb.com/Admin/Index.aspx>
* Devexpress controls Demo: <http://demos.devexpress.com/ASPxGridViewDemos/>
* Devexpress controls Documentation: <https://documentation.devexpress.com/#HomePage/CustomDocument9453>
* Project Folder: Foreclosure
* User types: Admin, Users, Super Admin
* Login Page: ForeClosure\Admin\Index.aspx
* Home page for admin : ForeClosure\Admin\Home.aspx
* Home page for users : ForeClosure\User\Home.aspx
* Grid Control currently using for admin: \ForeClosure\Admin\CommCont\RecapRecords.ascx
* Grid Control currently using for users: User/OptimizedPages/OptimizedGridUser.ascx

In addition to these pages old pages & alternative pages are there which can be accessed by directly typing URL

Main Functionalities for super admin:

* Home Page:
  + Displays the data based on the tabs displayed in the form of devexpress grid the data source used is devexpress XPO datasource. The grid functionalities are
    - Edit
    - View masterslip
    - View researcher documents
    - View comps
    - Sort by columns (by default the grid is sorted by Auction time & Sale location)
    - Group by columns (by default the grid is grouped by Auction time & Sale location which is not changed during further grouping)
    - The layout shown for the grid the saved in the DB whenever there is a change in the layout apart from default settings. The layout change includes sorting, grouping, column order change, adding/ removing a column, resizing a column
    - The grid allows editing via doubleclick in a cell, on single click anywhere other than the cell to which edit is done will save changes.
    - The researcher column allows view/upload the external documents for the corresponding record.
    - The context menu inside the grid is available when we select atleast one record. The functions for the context menu is
      * Allows to assign the selected record(s) to an user
      * Allows to categorize the selected record(s) to another category
      * Print selected record(s)
      * Export selected record(s) to excel
      * Reassign the selected record(s) : Changes auction time & auction location
  + Other functions
    - Get data Button:
      * Selects the data for the selected date & for the active tab
    - Collapse & Expand Buttons:
      * Collapse the Groping/ Expanding the grouping
    - Go button:
      * Based on the selection from the dropdowns on left side of the button corresponding action is done
      * If the selection for first dropdown is Manual pull the selection of record in grid is not needed for all other case the selection of records in grid is required.
    - Set column visibility:
      * Shows the columns list for the grid
      * Can drag & drop columns to the grid & from the grid
    - Show Filter:
      * Shows the filter bar inside the grid
* User Management Menu:
  + Create a new user
  + Edit or view user
  + Change password
* Master menu:
  + Col Add/ remove:
    - This is mainly for the users with user type user
    - The selected columns will be displayed in the user grid
    - For admin selection the visibility won’t be affected since the layout saving option is available.
  + Def. value
    - Which is saved as def value for robots
  + Manual Entry
    - Manual adding of property details
  + Restore data
    - Restore the deleted data
    - Now there is deletion policy which clears all data before a week so this functionality is only for those data that is deleted within a week
  + Foreclosure Radar
    - Commonly called as FRADAR upload
    - Client will upload .csv files for LA/ Ventura
    - The multiple upload function is available , which processes one file in the middle of another upload process.
    - If any error occurs the system will try to re-upload the data which is done for 3 times
    - The error data is uploaded to amazon s3
    - Bulk print/upload: will print after the upload
  + Visual Indicator
    - Shows the status of the URLs used in the robots
    - Records downloaded details
    - If ” Connection” shows failed then the robot’s site might have issues
  + FASAP manual upload:
    - Manual upload of FASAP data in the form of excel file
  + Audit trail Actions
    - Shows the manual changes made in the property listing by an user
    - Data is fetched from Audit\_Trail\_Actions
  + Robot time scheduler
    - Uses scheduler control from devexpress
    - Manages the working of robot
      * Add time slot
      * Deactivate
      * Delete
      * Set the priority
      * Days for which the robot is active
      * Add a new robot
      * Deactivate/ Activate all robots
  + Error alert
    - Add the rules for error alerting
    - Error alert robot works according to these rules
  + Metrics
    - Shows the graphical representation of updates made by the robots or the number of updates for specific fields.
* Legal
  + Glacier backup for a specific APN
* Printing
  + Used to monitor the remote printers
  + Views the current job queue
  + Status of the printers available
* Logout: logs out an user

Main Functionalities for admin:

All functionalities like that of Super admin is available for admin except the following

* User create
* View password for users in the user list

Main Functionalities for User:

* User has the option to view records based on the categories assigned for the user, column visibility set for the user & the records that is assigned for user
* The users can update only the column Purchase price

ROBOTS

Robot working cycle

* Fetch the data from the site
* Apply formatting logic to the data
* Create csv string for the data
* Upload the data to Amazon S3
* Insert the data to the temp table
* When migration robot works, the data from temp table will be moved to retrequred table.

Robots for gathering data

* FASAP: The data is from the XML file which is available in the URL
  + Locations: LA & Ventura
  + http://www.lpsasap.com/saleswirexml.aspx?state=ca&county=<los angeles or ventura>&saledate=<saledate>
  + Robot Types: Manual & Automatic
  + Number of robots: 4
* Auction:
  + Locations: LA & Ventura
  + The data is fetched from an excel sheet
  + [http://www.auction.com/xhr?format=xls&q=get\_inventory\_list&page=event&auction\_number=<auction](http://www.auction.com/xhr?format=xls&q=get_inventory_list&page=event&auction_number=%3cauction) number>&report\_type=inventory\_trustee
  + For getting auction number need to parse the html from the URL
  + <http://www.auction.com/calendar/all/CA>
  + Get the calendar list from that check for the auction date & sale location and find the auction number
  + Robot Types: Manual & Automatic
  + Number of robots: 4
* For other robots the data is parsed from the respective HTML page
  + FPassport:
    - URL: <https://www.fidelitypassport.com/logon.asp>
    - Purpose: Getting comparable properties
    - User Name : Strategic1
    - Password: Strategic
    - State: CA
    - County: VENTURA
    - Robot type: Automatic
    - Number of robots: 1
  + Google Address:
    - URL: https://maps.googleapis.com/maps/api/geocode/xml?sensor=false&address=<address>
    - Fetches google address from the geocode api
    - The input for the geocode is the address for the property from the robots
    - If google address is fetched correctly then that address won’t be fetched again
    - Robot type: Automatic
    - Number of robots: 1
  + AztecTrustee
    - URL: <http://aztectrustee.com/?page_id=82>
    - Locations: LA & Ventura
    - Robot type: Automatic & Manual
    - Number of robots: 4
  + Priority posting
    - URL: <http://www.priorityposting.com/PPPAdvancedWebSearch/Pages/disclaimer.aspx>
    - Locations: LA & Ventura
    - Robot type: Automatic & Manual
    - Number of robots: 4
  + RPP
    - URL: <http://www.rppsales.com/TrusteesSales/PropertiesNew.aspx>
    - Locations: LA & Ventura
    - Robot type: Automatic & Manual
    - Number of robots: 4
  + Nationwideposting
    - URL: <http://search.nationwideposting.com/SearchTerms.aspx>
    - Locations: LA & Ventura
    - Robot type: Automatic
    - Number of robots: 2
  + Realquest
    - URL: <http://realquest.com/jsp/rq.jsp?action=switch&page=main>
    - Purpose: Getting comparable properties
    - User Name : Warroom3
    - Password: Lakers24
    - State: CA
    - County: LOS ANGELES
    - Robot type: Automatic
    - Number of robots: 1
  + Nationwideposting
    - URL: <http://salestrack.tdsf.com/>
    - Locations: LOS ANGELES
    - Robot type: Automatic
    - Number of robots: 1
  + Zillow
    - Purpose: Gathering zestimates
    - The data is collected for the records having opening bid , having 3 months old zestimate & doesn’t have an zestimate
    - URL: http://www.zillow.com/webservice/GetSearchResults.htm?zws-id=X1-ZWz1dg0kk1y68b\_76gmj&rentzestimate=true&address=<address> &citystatezip=<city+state+zip>
    - Locations: LOS ANGELES
    - Robot type: Automatic
    - Number of robots: 1
  + Researcher
    - Get the records with a research file uploaded
    - Get the retflid of the rcords with research files
    - Insert the data to temp table
    - Upload the data to AWS S3 in the format of a CSV

Other Robots

* AWSManagement
  + Used for clearing the CSVs uploaded to Amazon S3
  + Find the files which is 5 days old & deletes those files
* ErrorAlert
  + Alert mail is send when there is an error in PrintUtility
  + Alert mail is send when there is no log written by a robot in 15 minutes
  + Send recap mail
  + Check for error alert condition’s violation
* FCBidAlert
  + Not using currently
* FCCSharpLibrary
  + Library functions in c# for Amazon services, google cloud print, grid data formatting etc
* Getconfigdetails
  + Gets the configuration data for each robots
  + The configuration details are
    - Time slot
    - Days to run
    - Whether the robot is active or not
  + Mailing function to send a mail if there is an error occurred during the creation of csv using data from site
* Gmail
  + Reads mail from the gmail account of Foreclosure & downloads the pdf attachment file & update the for the corresponding file for the APN which is read from mail’s subject
  + The downloaded files are stored in Amazon s3
* Library
  + Contains common dlls for the project, the dlls location is bin/debug
* LitHoldBackupRobot
  + Uploads the backup data of 6 days, this robot works for every Saturday.
  + The backup details are
    - Retrequred data for last 6 days
    - Log data corresponding to the records from retrequred
    - All files corresponding to the selected records from aws s3
  + The database details are made to a .csv file
  + The files from aws is downloaded
  + All these files are zipped together and uploaded to AWS Glacier
  + After uploading the zip an archiveid is received from Glacier which is inserted into DB
* LogManager
  + Inserts log details into the db
* Migration robot
  + This robot inserts the data from temp table to the retrequred for each robots
  + A priorityrank table is maintained inorder to check which temp table is processing
  + The migration details can be viewed using the SP “spgettemptablecount'<date in mm/dd/yyyy format>'” eg: spgettemptablecount'06/25/2015' the column name is “IsProcessing”
* RDSManagement
  + Not using currently
* Recap
  + Creates a pdf file from the qc values for certain time intervals, this pdf is then mailed to the given mailids
  + Creates an one time pdf for the qc values termed as “early morning check” , this pdf is mailed every day at 3pm IST
* RecapTab
  + Select the updation data from Audit table for the columns 'OpeningBid', 'MinBid', 'SalesStatus' by robots
  + The details "Type", "TableName", "PK","FieldName", "OldValue", "NewValue", "UpdateDate", "UserName", "Insert\_UpdatedBy" are entered into “Audit\_For\_Recap” table
  + This data is displayed in the recap tab in FC UI

Foreclosure Researcher

This application is a part of Foreclosure but deployed as different application there is only one repository ie Live

DB Connection strings & URLs

* Database
  + Server=c4c2a995-8f97-4e78-9ead-a18e00ce3f54.sqlserver.sequelizer.com
  + Database=dbc4c2a9958f974e789eada18e00ce3f54
  + User ID=kmqetuepnqidqcjp
  + Password=zsuDQQzBmQ84Ttxp3qNon8xNUnbbDMvExBP4chDSdk7eWpZnqdnwygwz4NSBNx7G
* URL:
  + Login Page: <http://researcher.apphb.com/index.aspx>
  + Super Admin Home: <http://researcher.apphb.com/SuperAdmin/Home.aspx>
  + Researcher Home : <http://researcher.apphb.com/Researcher/Home.aspx>

Researcher Web

* Researcher:
  + In the home page the records from foreclosure application is listed.
  + The records will have the next aution day
  + These records are inserted to researcher DB via Migration robot’s ResearchDataFetch.vb class
  + Here the timezone needs to be converted to PST from UTC
  + The functionality of researcher user type is to upload documents for a particular property
  + The files uploaded will be stored to Amazon S3 ,the file path is : /Strategic/Researcher
  + The file details is stored in the table FileUploadData of Foreclosure DB
* SuperAdmin
  + The function of superadmin is to create new users ,edit the exitsting user details or delete an user from the system

Researcher Robots

Researcher has two robots, which is deployed as part of Foreclosure

1. Robot which inserts data to researcher
   1. The code is written as part of migration robot. The researcDataFetch.vb class works after the processing of other robots ie moving data from temp table to retrequred table
2. Robot which reads the file details from researcher to temptable
   1. The researcher robot fetches the details of files ie the records having a file, to the temp table
3. Robot which reads the file details from researcher temptable to retrequred
   1. ResearchDataUpdate.vb is the class for this purpose which written as part of Migration robot
   2. The researcher temp table contains the retflid(PK of retrequred) which is used for marking the hasresearchfiles column’s value as 1. HasResearchFiles=1 indicates there is a valid file uploaded for that RetFlid.
   3. In the foreclosure UI, the file details will be gathered from the FileuploadData table for the selected RetFlId
   4. From the Foreclosure Web Files can be uploaded.