**Appendices R**

**System Testing**

As the system I developed has a broad functionality range, I decided to split my testing into a number of areas focusing on the priority functionality from the system, this meant I looked at testing the below areas:

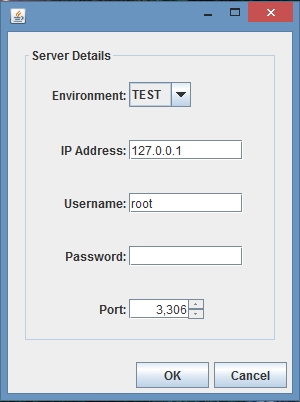
* Login functionality
* Password reset functionality
* User permissions functionality
* Home Screen – Tenancy/Lease/Rent Account live feed functionality
* Document Management functionality
* Advertise Services through website
* Service Request through website

Furthermore, as I developed a GUI for the user of the system to interact with the system I am going to document my results through print screens of the GUI demonstrating that system functionality is working.

1. **Login Functionality**

As explained in the report, the system functionality requirements include implementing a login system which requires the user of the system to input a username and password combination, along with system server settings to enable login to the system, and if the information entered did not match then entry to the system would be refused.

As I have adopted a client server architecture, I firstly have to run the server, which the client will connect to, and to do this I need to supply the server parameters which include the environment (LIVE, TEST, TRAIN), the MySQL IP address, username and password.

Fig. 1 – Server Login Dialog

As you can see from Fig.1 (left), Once I compile and run the MSc Properties Server package I am presented with a server login dialog, in which I have to enter the server details explained above. For this session I have selected the TEST environment, and as the MySQL database is running on localhost I have entered localhost IP address, along with the Port 3306 and the database username and password.

Once I select OK the server will start running, unless there are errors with the information supplied, if I click Cancel the dialog will close and the application ends.

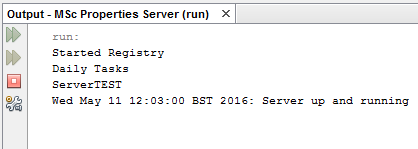


Fig. 2 – Server Output

As you can see from Fig. 2 (left), once the Server has been run a message will be output to the terminal stating the Server run information.

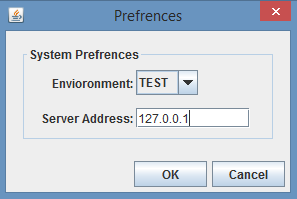
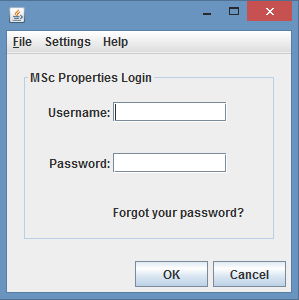
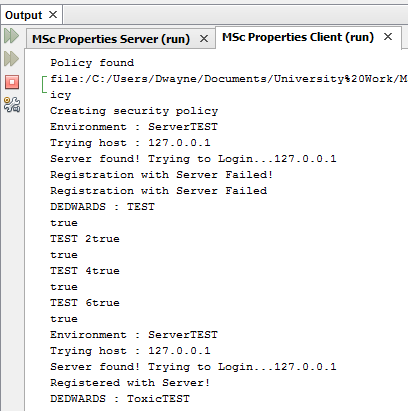
Now the server is running, I can run the MSc Properties Client package to connect to the server.

Fig. 4 – Client Preferences Dialog

Fig. 3 - – Client Login Dialog

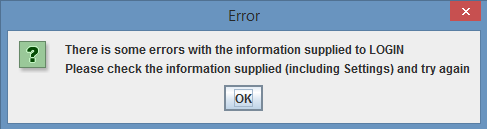
As you can see from Fig. 3, when the MSc Properties Client package is run I am presented with a Client Login dialog, where I can enter the username and password combination to log in, however the system environment and server IP address also needs to be provided to enable the login process to be successful.

The system developed has system preferences for the server details and on first log in the system preferences is set to default values, but once I change the settings, these will be stored locally (on the same computer) even once the client application ends, which means I do not need to keep providing the system details, unless the system details change, or I want to log in to a different environment. Once the information is correctly entered, and I click OK, the application will register the client with the server, and one of two screens will be presented depending on the user account state, if I click cancel then the client login dialog will close and the application ends.

Fig. 5 – Successful Login Terminal Output

As you can see from Fig. 5, the terminal output, when the correct login information is supplied the Server is found and Registration with the Server is Successful, the login dialog will then disappear and either the home screen or the Employee Security Update screen will appear depending on the state of the user.

Fig. 6 – Failed Login Dialog



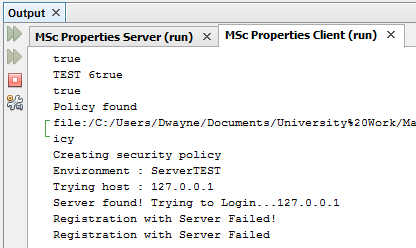


Fig. 7 – Failed Login Terminal Output

As you can see from Fig. 6, if the Client parameters are incorrect then an Error dialog will appear telling the client that there are errors with the information supplied, and that the client needs to check information supplied and try again, also Fig. 7, the terminal output which shows that the Server was found and login process is being attempted, but then registration with server failed.

This testing shows that the login functionality is working as expected, as the login functionality prevents a user from logging in to the system unless the username or password combination matches and existing user, and the server details are correct. Additionally, once I set the server settings and logged out of the application the, settings were retained and when I logged back in these did not have to be amended again.

1. **Password Reset Functionality**

As explained in the report, the system functionality requirements include implementing a password reset functionality, which has two aspects to the functionality, being that a user is able to update their own password by providing their employee details and an email is then sent with the new password to the user, or by an employee that has sufficient privileges to update another employee’s password, amending the employee password.

* 1. **Reset own Password**

For a user to reset their own password it requires the user to provide information which includes username, email address, memorable location (set up by the user), their employee ref, and as well as the same information required when logging in normally, which is the server IP address and environment.

As you can see from Fig. 3 the Client Login Dialog, there is a “forgot your password?” link which takes me to a Forgot your Password dialog, where I need to input the above listed information, which is shown in Fig. 8 below.

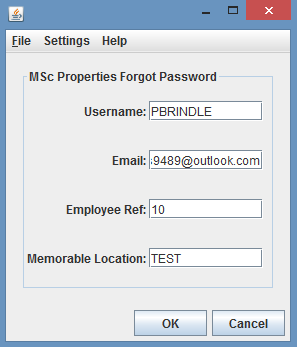


Fig. 8 – Forgot Password Dialog

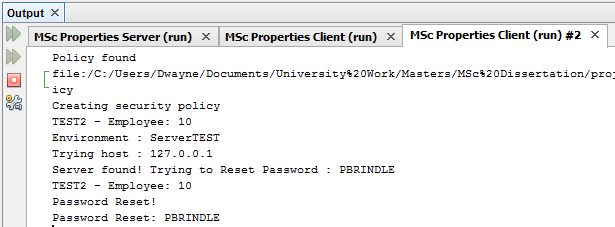


Fig. 9 – Password Reset Successful Terminal Output

If I click OK and the information supplied is correct then the server will then reset my account password and send an email to the registered email account, containing the updated password. The Forgot Password dialog then disappears and the Login dialog appears for the me to then login with the new password provided through the email. The email is shown in Fig. 10, which shows the amended password.

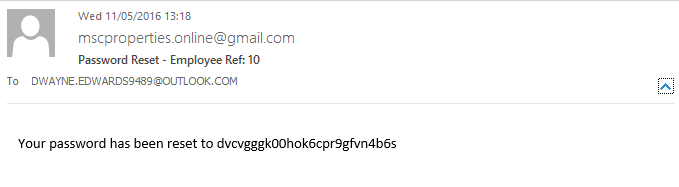


Fig. 10 – Password Reset Email

As you can see from Fig. 10, an email has arrived from [mscproperties.online@gmail.com](mailto:mscproperties.online@gmail.com) to my registered email address of [Dwayne.edwards9489@outlook.com](mailto:Dwayne.edwards9489@outlook.com) stating that the password has been reset and providing the new password, this can then be used to log in to the system.

If I click OK and the information supplied is incorrect then an Invalid User Exception is thrown, and I can amend the information and try and submit the information again.

If the client clicks Cancel, then the Forgot Password dialog disappears and the Login dialog reappears.

This test shows that the reset your own password functionality is working correctly as upon submitting the correct forgot your password details, my password was reset and an email was sent to my email address stating the amended password, for me to attempt to log in, which was successful with the password in the email.

* 1. **Update another Employee Password**

For a user to reset another employee’s password it requires the user to go to the Employee Details of the employee in question, and select the Actions menu, and the Update Employee Security menu item, an update employee password dialog appears where the user can enter the new password, and once the user selects OK the password is updated. The user can then forward the amended password on to the user to allow them to log in.

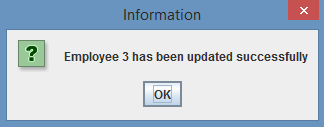
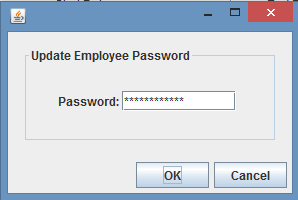


Fig. 12 – Confirm Dialog

Fig. 11 – Update Employee Password Dialog

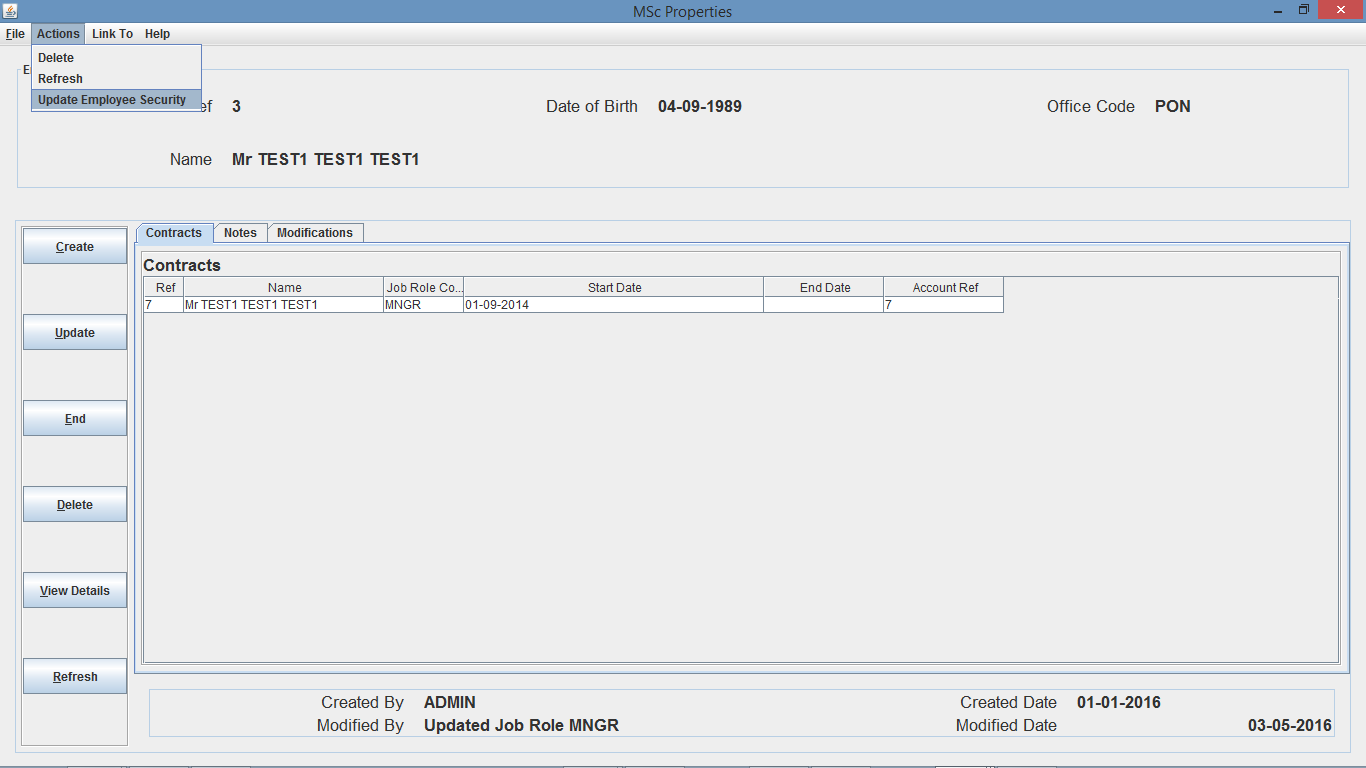
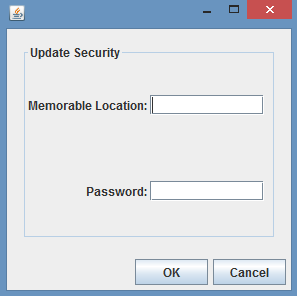


Fig. 13 – Employee Details Frame (with Actions menu open)

I can then login in with the username and amended password combination successfully. This test shows that the reset another employee password functionality is working correctly as upon me amending the user’s password, and clicking OK, I was then able to log in with the user’s username and amended password.

Additionally, to improve the security of the system, if a user’s password is amended in any of the two ways tested above, when the user tries to log in for the first time after the password amendment, the user will be presented with the update employee security dialog, shown in Fig. 14, which prompts the user to amend their password before moving on to the home screen which was successfully tested during this process.

Fig. 14 – Update Employee Security

1. **User Permissions Functionality**

As explained in the report, the system functionality requirements include implementing user restrictions, which restrict users from performing functions that they do not have permission to perform. The user permissions functionality works on the creating, reading, updating and deleting non-employee information or employee information, which is assigned to the user through their Job Role. So an employee can have 8 different permission rights.

* 1. **Create non-employee information**

All users with the permission to create non-employee information can create data within the system relating to anything other than employees, meaning that if I am logged in as a user with create privileges for non-employee information I will be able to create a system element for the title of a person.

To do this I navigated to the System Configuration area by clicking the System Config button from the home screen, and then navigate to the Titles area within the left pane of the System Configuration area, which will open up a Titles branch to the System Configuration area, where I can select Create Title. Whether the user I am logged in with has the permission to create a title or not, the create title dialog will open, and the user is able to enter the information to create a title.

If the user I am logged in with enters correct information and clicks OK then a dialog will appear asking to confirm the creation even if the user does not have permission, and if I select OK then this is where the system will throw an Exception if I do not have permission to invoke this method, stating “User does not have privileges to invoke this method – username” and does not invoke the method on the server, which is shown in the terminal output as shown in Fig. 15. If I do have permission, then the system will return a dialog saying the title was created successfully as shown in Fig. 16.

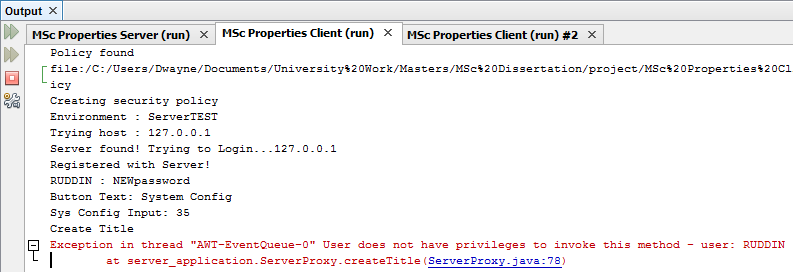


Fig. 15 – Create Title Unsuccessful Terminal Output

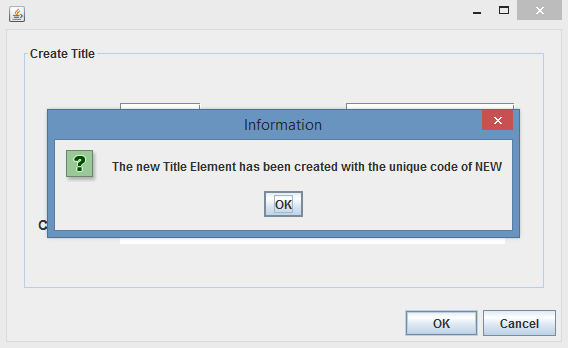


Fig. 16 – Create Title Successful Dialog

* 1. **Read non-employee information**

All users with the permission to read non-employee information can read data within the system relating to anything other than employees, meaning that if I am logged in as a user with read privileges for non-employee information I can view system element details for a title of a person.

To do this I navigated to the System Configuration area by clicking the System Config button from the home screen, and then navigate to the Titles area within the left pane of the System Configuration area, which will open up a Titles branch to the System Configuration area, where I can select View Titles.

If the user I am logged in as does not have permission to invoke this method, then an Exception is thrown, outlining the same exception as previously outlined, and the method is not invoked on the server, which is shown in the terminal output as shown in Fig. 17. If the user I am logged in as does have permission then this will open up a table of titles, where the I can perform actions on a title and is shown in Fig. 18.

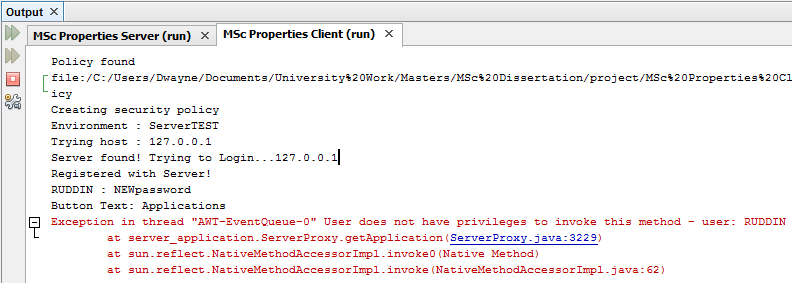


Fig. 17 – Update Title Unsuccessful Terminal Output

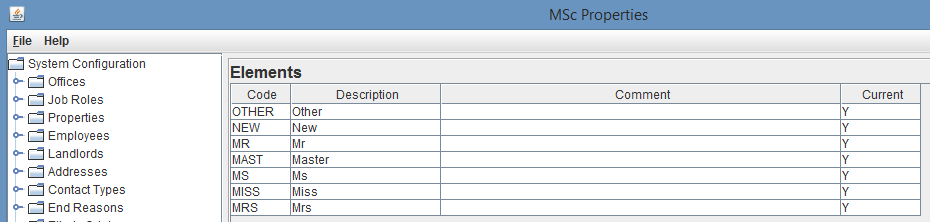


Fig. 18 – Successful Read Titles

* 1. **Update non-employee information**

All users with the permission to update non-employee information can update data within the system relating to anything other than employees, meaning that if I am logged in as a user with update privileges for non-employee information I will be able to update a system element for the title of a person.

To do this I navigated to the System Configuration area by clicking the System Config button from the home screen, and then navigate to the Titles area within the left pane of the System Configuration area, which will open up a Titles branch to the System Configuration area, where I can select View Titles. This will then open up a table of titles, where the user can right click on a title and select view Element, and then select update from the Actions menu. This will open up an update title dialog, where the user can amend information of the title.

If I enter correct information and click OK then a dialog will appear asking to confirm the creation even if the user I am logged in as does not have update permission, and if I select OK then this is where the system will throw an Exception if I do not have permission to invoke this method, stating “User does not have privileges to invoke this method – username” and does not invoke the method on the server, which is shown in the terminal output as shown in Fig. 19. If I do have permission to invoke the method then the system will return a dialog saying the title was updated successfully as shown in Fig. 20.

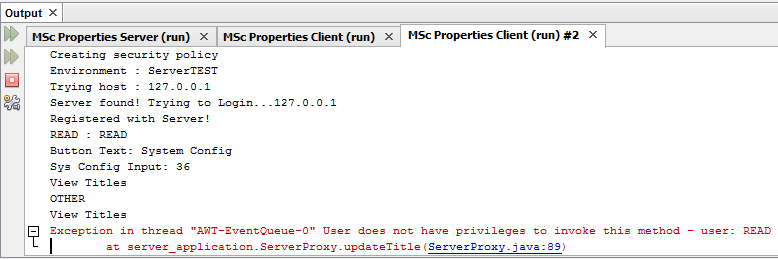


Fig. 19 – Unsuccessful Terminal Output

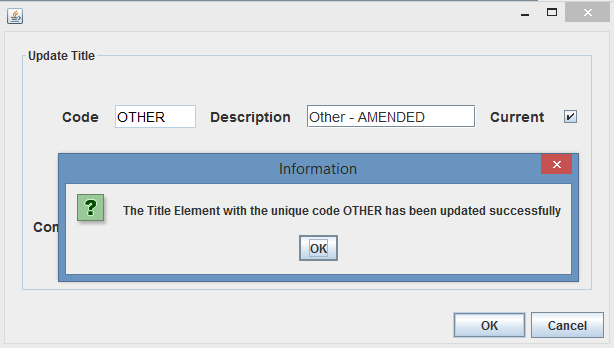


Fig. 20 – Successful Update Title

* 1. **Delete non-employee information**

All users with the permission to delete non-employee information can update data within the system relating to anything other than employees, meaning that if I am logged in as a user with delete privileges for non-employee information I will be able to delete a system element for the title of a person.

To do this I navigated to the System Configuration area by clicking the System Config button from the home screen, and then navigate to the Titles area within the left pane of the System Configuration area, which will open up a Titles branch to the System Configuration area, where I can select View Titles. This will then open up a table of titles, where I can right click on a title and select view Element, and then select delete from the Actions menu. This will open up a Yes/No dialog, asking me to confirm deletion of the title.

If I do not have permission and select yes then the system will throw an Exception, stating “User does not have privileges to invoke this method – username” and does not invoke the method on the server, which is shown in the terminal output as shown in Fig. 21. If I do have permission, then a dialog box will appear confirming deletion of the title as shown in Fig. 22.

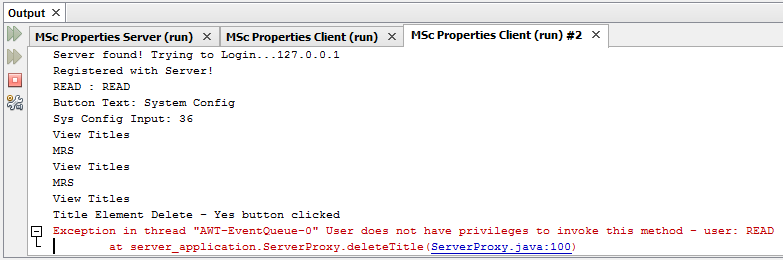


Fig. 21 - Unsuccessful Terminal Output

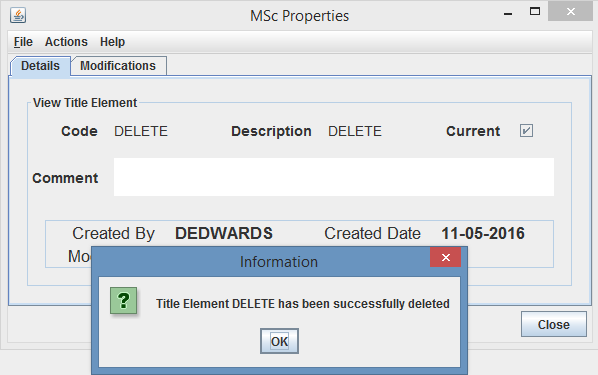


Fig. 22 – Successful Delete Title

* 1. **Create employee information**

All users with the permission to create employee information can create data within the system relating to employees, meaning that if I am logged in as a user with employee create privileges then I will be able to create a contract for an employee.

I can do this from a few different areas, such as the contract search screen or employee details, I am going to show my testing through the contract search screen route. So I navigate to the contract search screen by clicking the Contracts button from the home screen, and then select the create button. A create contract dialog will open and the user will have to provide the create contract parameters, employee ref, length, start date, office code and job role code.

If I enter correct information and do not have permission to create employee information then upon clicking on the OK button a yes/no create dialog will open for me to confirm creation of contract, and if I select yes, an Exception will be thrown stating “User does not have privileges to invoke this method – username” and does not invoke the method on the server, which is shown in the terminal output in Fig. 23. If I do have permission, then a dialog will appear confirming the creation of the contract as shown in Fig. 24.

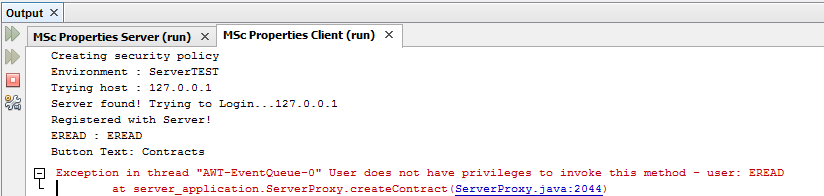


Fig. 23 – Unsuccessful Terminal Output

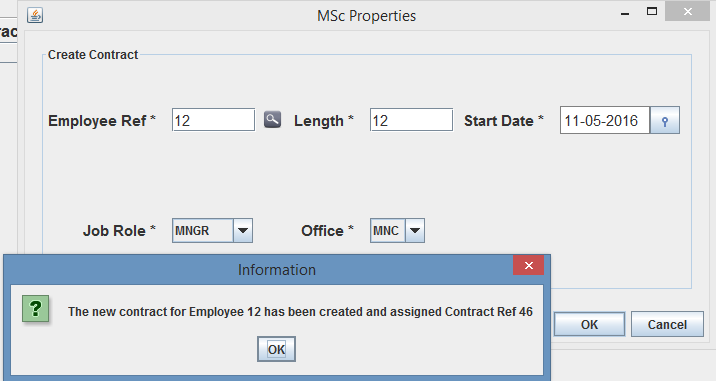


Fig. 24 – Successful Contract creation

* 1. **Read employee information**

All users with the permission to read employee information can read data within the system relating to employees, meaning that if I am logged in as a user with read employee privileges, I will be able to view contract details for an employee.

To do this I have to navigate to the Contracts search screen by clicking the Contracts button from the home screen, and either use the advanced or standard search to search for a contract, if there is a contract that meets the search criteria, then I can right click on a contract returned and select view details. This will then bring up the contract details screen.

If the user I am logged in as does not have permission to read employee information then upon clicking on the search button either through the advanced or standard search, an Exception will be thrown stating “User does not have privileges to invoke this method – username” and does not invoke the method on the server, which is shown in the terminal output as shown in Fig. 25. If the user does have permission, then contracts will return in the search results screen and the user can then go to the contracts details screen as shown in Fig. 26.

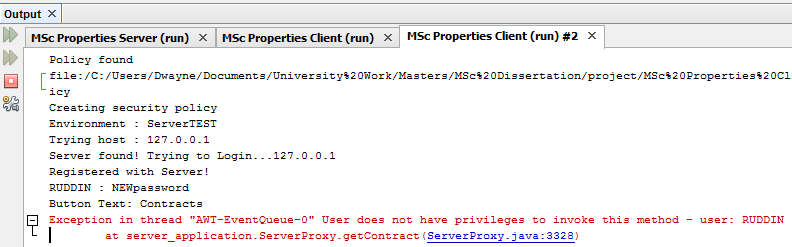


Fig. 25 – Unsuccessful Terminal Output

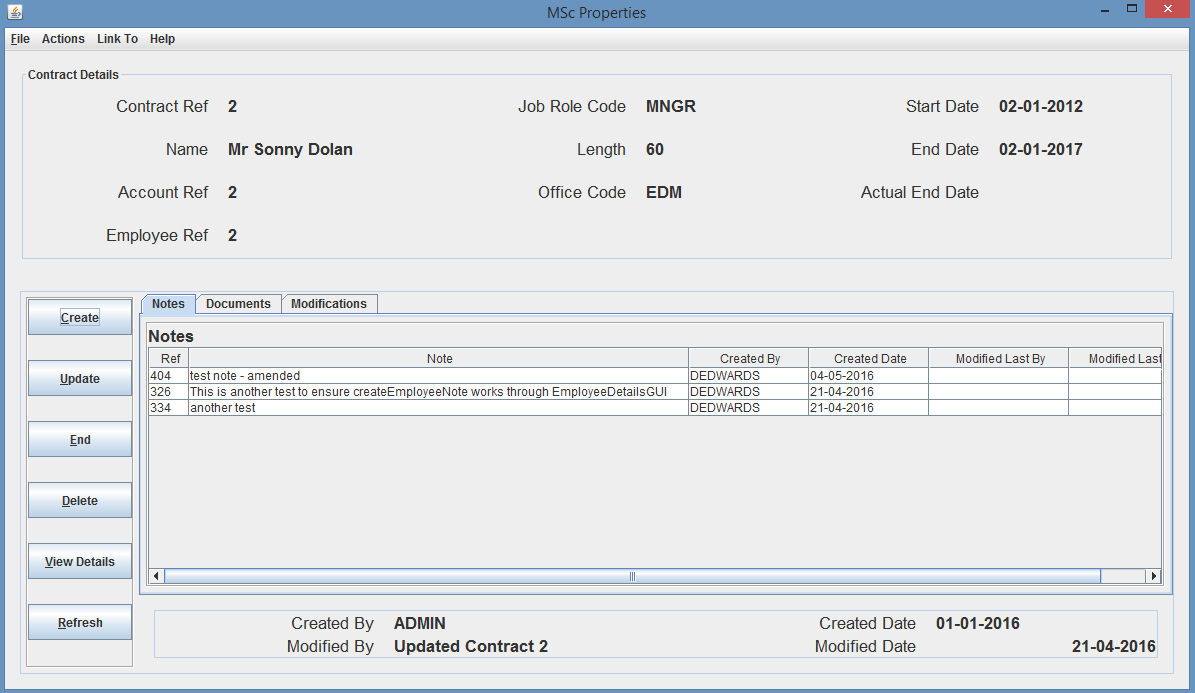


Fig. 26 – Successful viewing of contract details

* 1. **Update employee information**

All users with the permission to update employee information can update data within the system relating to employees, meaning that if I am logged in as a user with employee update privileges I will be able to update contract details for a contract.

To do this I have to navigate to the Contracts Details screen as shown before and then I can select Update from the Actions menu, where an update contract dialog will open allowing me to amend the contract information.

If the user I am logged in as does not have permission to invoke this method, but amends the information as required and clicks the OK button, then a yes/no confirm dialog appears asking to confirm the update, if I select yes, then an Exception will be thrown stating “User does not have privileges to invoke this method – username” and does not invoke the method on the server, which is shown in the terminal output as shown in Fig. 27. If the user does have permission, then a confirmation dialog appears confirming the update of the Contract as shown in Fig. 28.

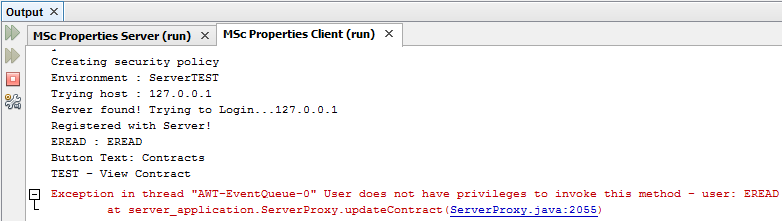


Fig. 27 – Unsuccessful Terminal Output

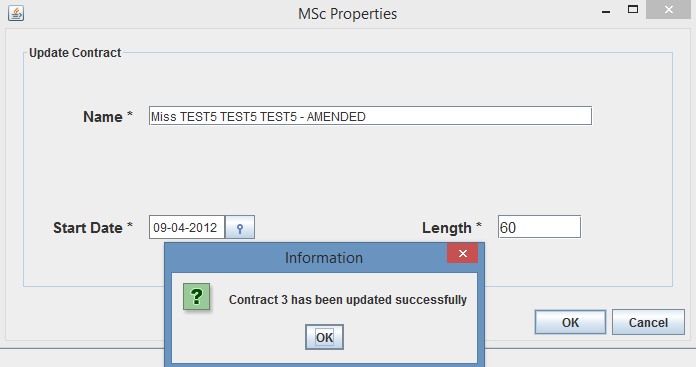


Fig. 28 – Successful Contract Update

* 1. **Delete employee information**

All users with the permission to delete employee information can delete data within the system relating to employees, meaning that if I am logged in as a user with employee delete privileges I will be able to delete a contract providing there is no dependent records relating to the contract.

To do this I have to navigate to the Contracts Details screen as shown before and I can then select Delete from the Actions menu, where a yes/no dialog will appear asking me to confirm deletion of the contract.

If the user I am logged in as does not have permission to invoke this method, but selects yes, then an Exception will be thrown stating “User does not have privileges to invoke this method – username” and does not invoke the method on the server, which is shown in the terminal output as shown in Fig. 28. If the user does have permission to invoke the method, then a confirmation dialog appears confirming the deletion of the Contract as shown in Fig. 29.

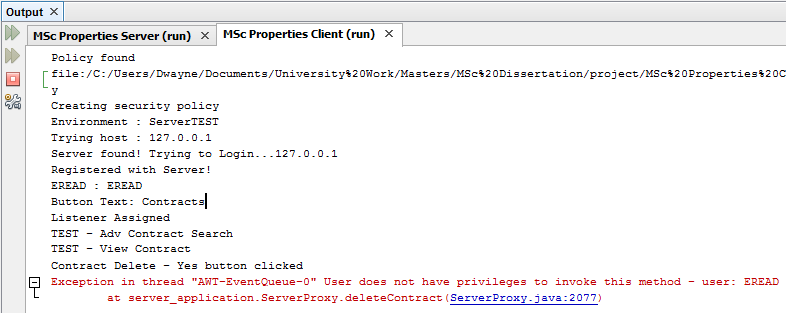


Fig. 28 – Unsuccessful Terminal Output

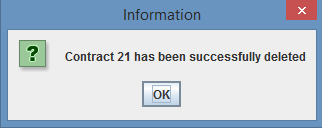


Fig. 29 – Successful Contract Deletion

From the above testing around user permissions, I believe this shows that the functionality has been successfully implemented and is working correctly, because all employees who will have been assigned a user account to login, will be restricted to the user privileges assigned by the employer, ensuring the users are unable to carry out functions that they do not have the correct privileges to perform.

1. **Home Screen – Tenancy/Lease/Rent Account live feed**

As explained in the report, the system functionality requirements include implementing a home screen which provides a live feed of tenancies, leases and rent accounts for the office the logged in user works for. The live feed will provide a list of 10 tenancies and 10 leases in descending order of expected end date, and a list of 10 rent accounts in descending order of balance.

The home screen will update when any tenancy, lease or rent account is created, amended or deleted from the user’s office, providing up-to-date lists at all times.

To carry out testing on the home screen it required me to test with users from different offices, ensuring that updates are not propagated to users from an incorrect office, and ensure that changes to the home screen are working correctly.

Firstly, I had to start the server, as shown in previous testing, and then run 3 different clients, logging in as different users for each client session, of which 2 users have to be of different offices and have to have read privileges, the third client needs to have write, update and delete privileges to ensure the user can create, update and delete a tenancy, lease or rent account, to initiate a user home screen update.

Before I started testing I took some print screens of two users home screens from different offices, as you can see from Fig. 30 and Fig. 31 I have logged in to two clients from office “MNC” and “EDM” respectively.

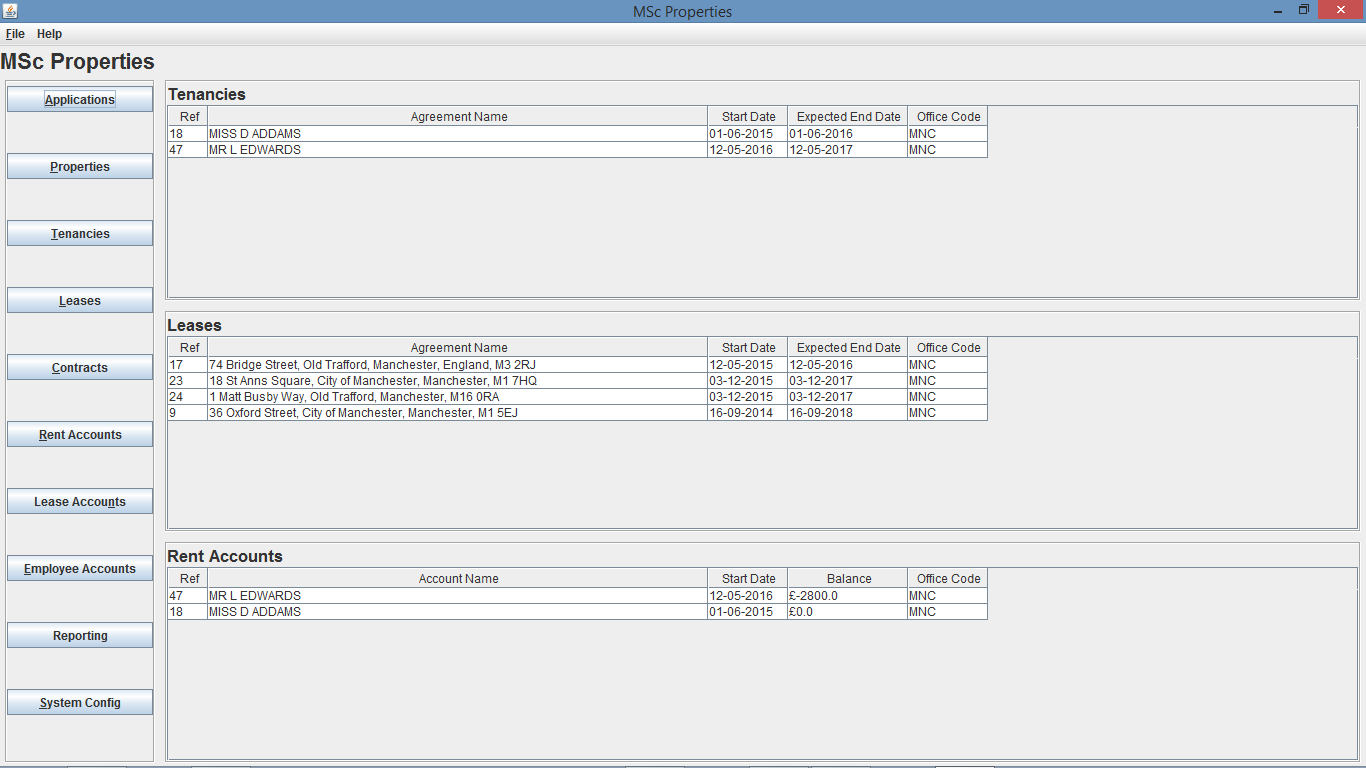


Fig. 30 – Home Screen of user from “MNC” office

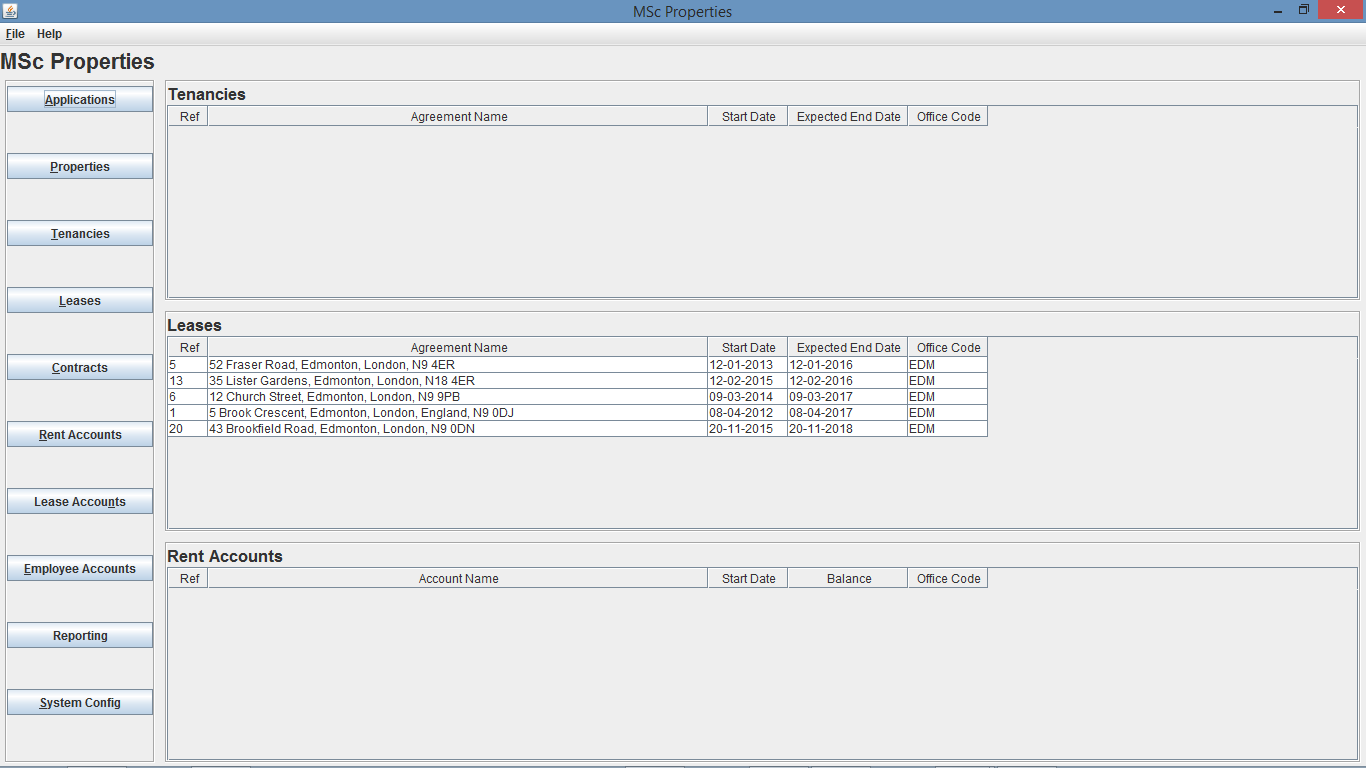


Fig. 31 – Home Screen of user from “EDM” office prior to tenancy creation

* 1. **Create, Update, Delete – Tenancy**

My first test is going to be to create a new tenancy using the 3rd user account that has read, write, update and delete privileges, and due to rent accounts being created off the back off tenancy creations, this test will also result in a rent account update, and therefore will update the rent accounts table as well as the tenancies table.

To create a tenancy, I navigated to the Tenancies search page by clicking on the Tenancies button from the home screen, I then clicked create from the tenancy search screen, where a tenancy create dialog box opens, I can then use the app search button to search for the required application ref if not known.

If I enter all of the correct information ensuring the office I assign the tenancy to, is one of the offices I expect the update to occur on, for this test I decided to use the “MNC” office. Once I click the OK button, a yes/no create dialog will open for me to confirm creation of the tenancy, in which if I select yes a confirm dialog will appear confirming creation of the Tenancy, as you can see in Fig. 32.

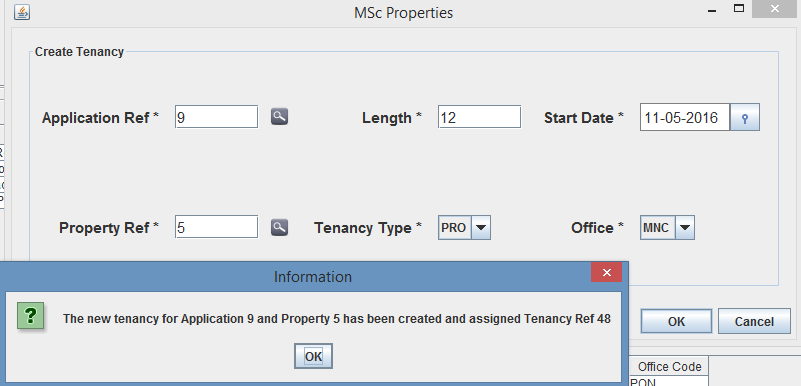


Fig. 32 – Tenancy creation Dialog

As you can see from Fig. 32, I selected “MNC” as the office and the newly created tenancy was assigned Tenancy Ref 48. So I am now expecting the user from office “MNC” to now show the newly created Tenancy along with the Rent Account on the home screen, however I will not see the creation on the home screen of the user from office “EDM” and this is shown in Fig. 33 and Fig. 34.

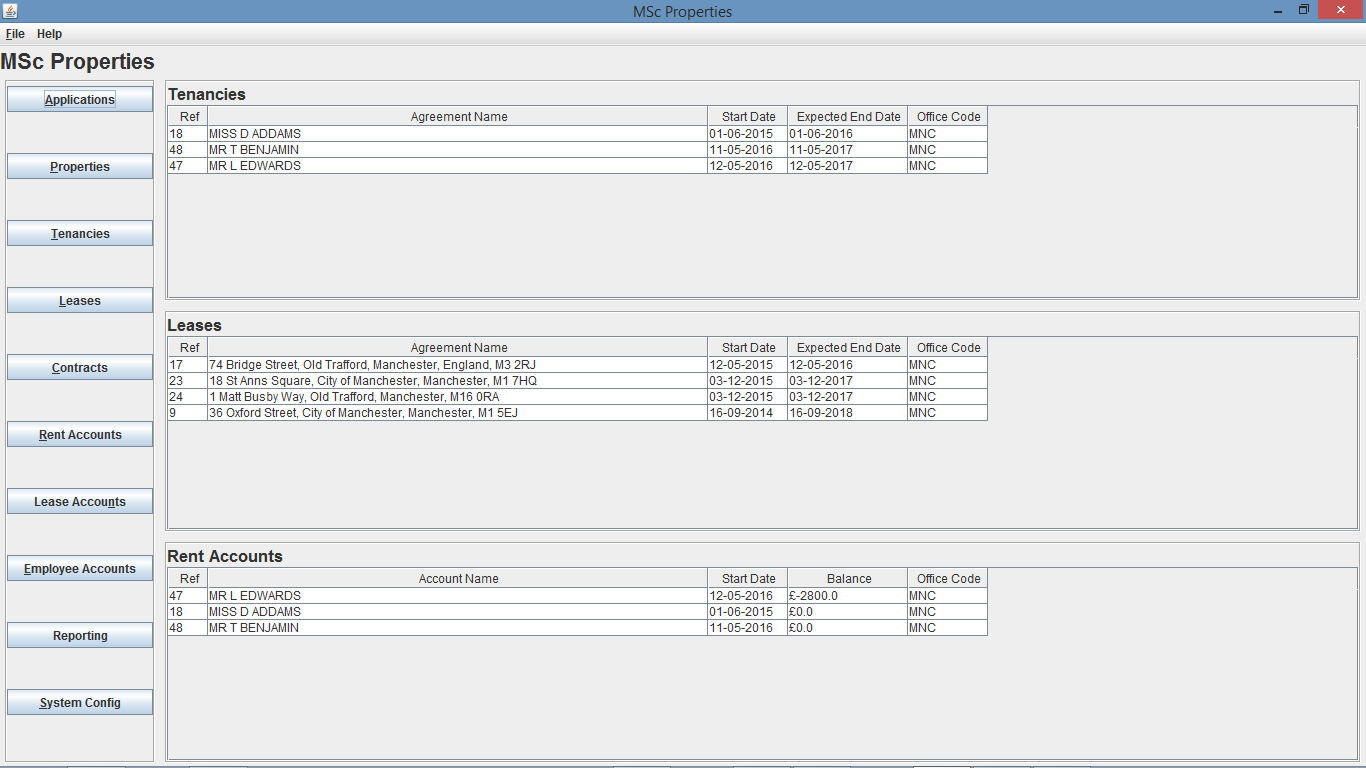


Fig. 33 - Home Screen of user from “MNC” office after creation of Tenancy Ref 48

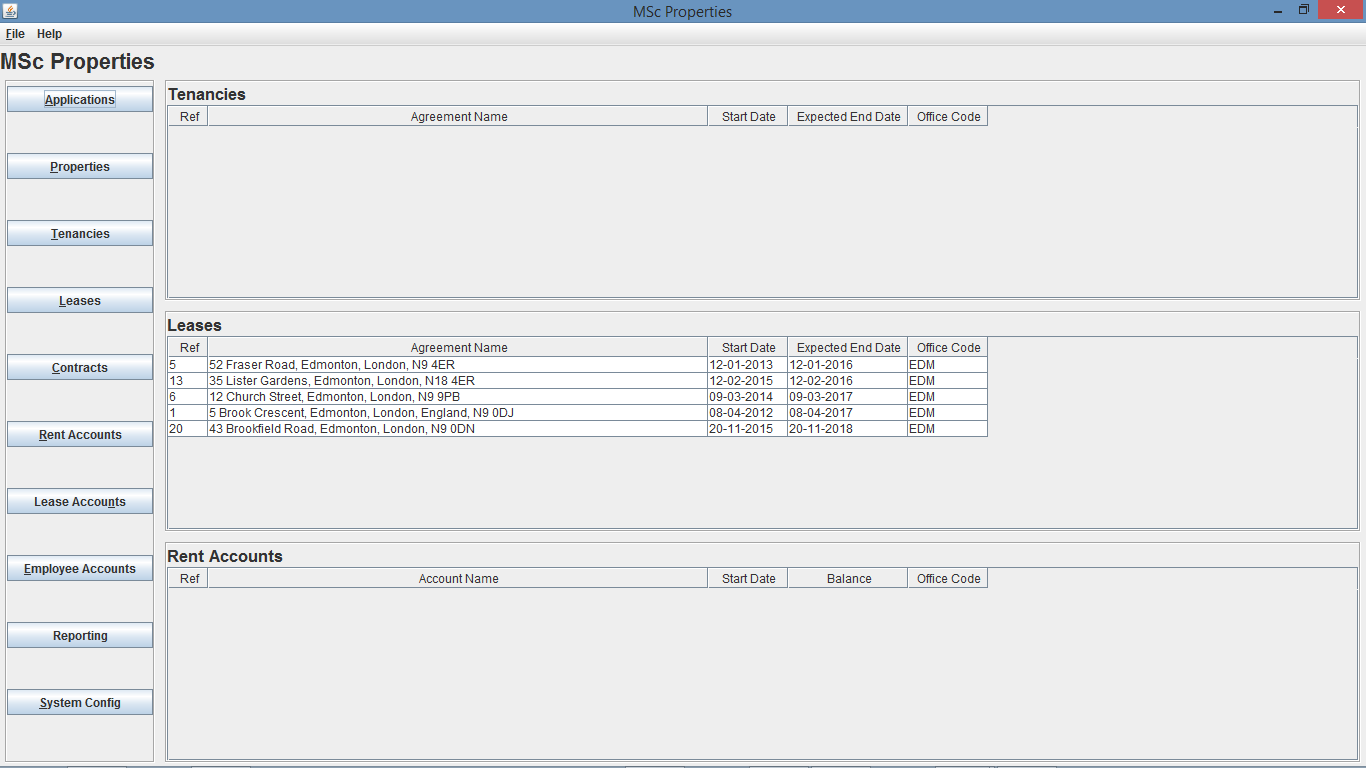


Fig. 34 - Home Screen of user from “EDM” office after creation of Tenancy Ref 48

My second test is going to be to update a tenancy, again using the 3rd user account, and due to rent accounts receiving certain updates like name, these updates will be reflected in both the tenancy and rent account tables.

I can update a tenancy through two ways, either by going to the tenancy details screen, or from the tenancy search screen. For this testing I am going to update a tenancy through the tenancy search screen, again by navigating to it from the home screen by clicking the Tenancies button. Once I am there I have to search for the tenancy, and again as I need to select a tenancy from a particular office I decide to use the standard search and search for tenancy 18, as this is from office “MNC”, but any search method could have been used.

I then need to right click on the tenancy I want to update, and select Update Tenancy, and as explained for testing I decided to update Tenancy Ref 18 from both Fig. 30 and Fig. 33. As you can see from Fig. 34, I updated the agreement name to “MISS D ADAMS – AMENDED” and the length of the tenancy to 36 months, meaning the expected end date should change to 01-06-2018 resulting in the tenancy dropping to the bottom of the list due to the tenancy being the last to potentially end out of the 3 tenancies for office “MNC”.

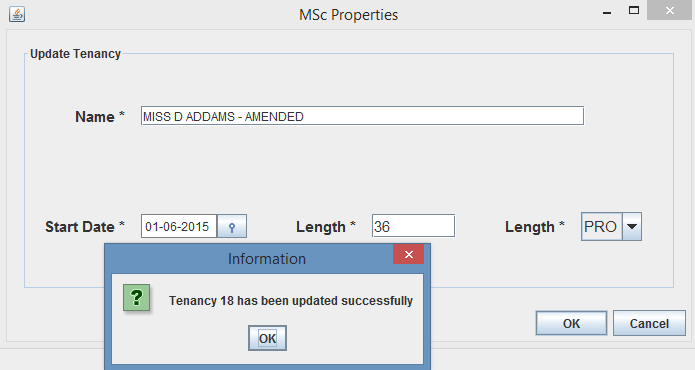


Fig. 35 – Tenancy Amendment Dialog

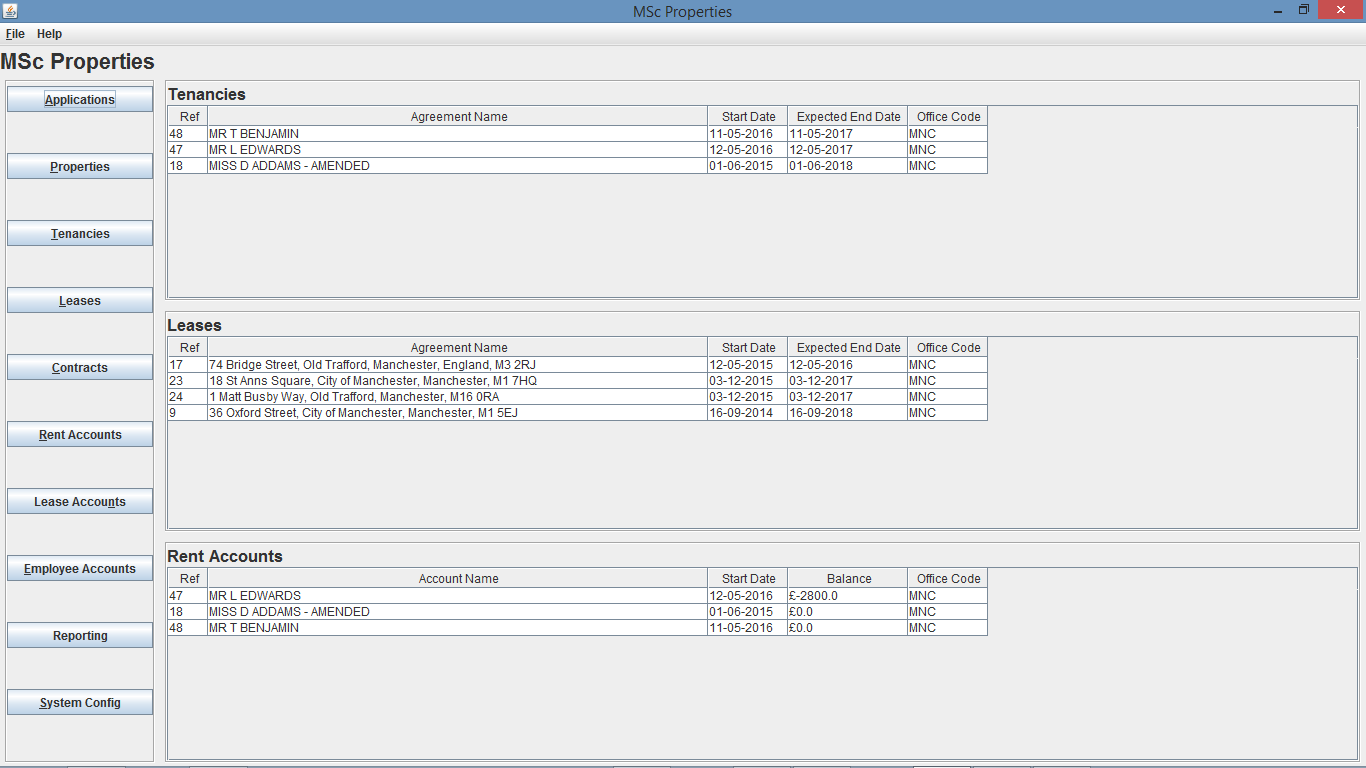


Fig. 36 - Home Screen of user from “MNC” office after amendment of Tenancy Ref 18

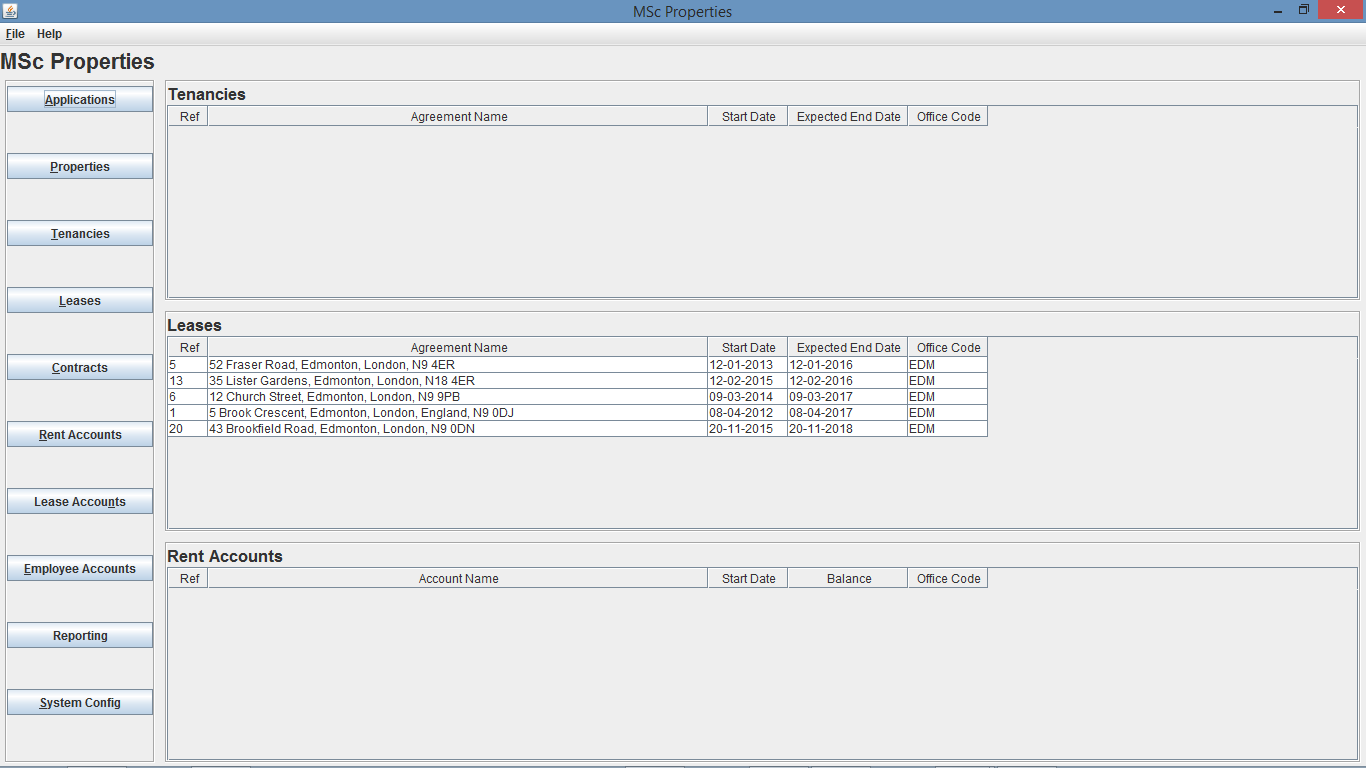


Fig. 37 - Home Screen of user from “EDM” office after amendment of Tenancy Ref 18

As explained I was expecting the user from office “MNC” to now show the amended Tenancy along with the amended Rent Account on the home screen, as well as the updated list position in the Tenancies table (Rent Accounts list positions has not changed as that is down to the balance amount, as a pose to expected end date for Tenancies list). Also as expected I will not see the amendment on the home screen of the user from office “EDM” and this is shown in Fig. 36 and Fig. 37.

My third test is going to be to delete a tenancy, again using the 3rd user account, and due to rent accounts being dependent on the tenancy, meaning the rent account cannot exist without the tenancy existing, the deletion of a tenancy will be reflected in both the tenancy and rent account tables.

I can delete a tenancy through two ways, either by going to the tenancy details screen, or from the tenancy search screen. For this testing I am going to delete a tenancy through the tenancy search screen, again by navigating to it from the home screen by clicking the Tenancies button. Once I am there I have to search for the tenancy, and again as I need to select a tenancy from a particular office, but that does not have any dependent records, I decide to use the standard search and search for tenancy 48 which was created earlier, as this is from office “MNC”, but any search method could have been used.

I then need to right click on the tenancy I want to delete, and select Delete Tenancy, a yes/no dialog will then appear asking me to confirm the deletion of tenancy 48, and if I select yes this will then delete Tenancy 48 and a confirmation dialog will appear as shown in Fig. 38.

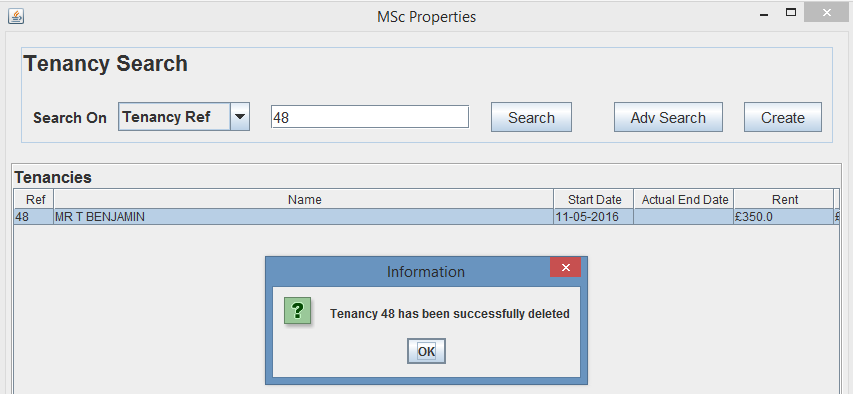


Fig. 38 – Tenancy Deletion Dialog

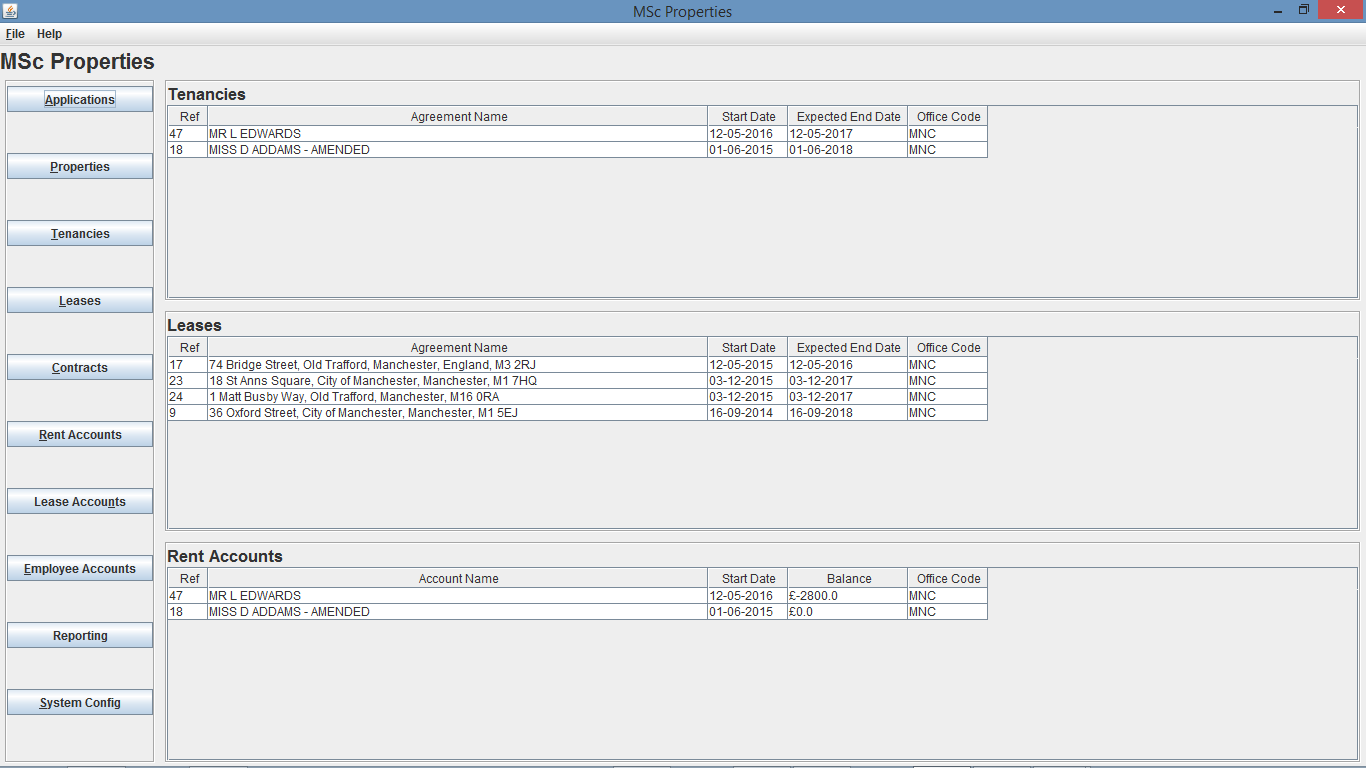


Fig. Fig 39 - Home Screen of user from “MNC” office after deletion of Tenancy Ref 48

As you can see from Fig. 39 the deletion of Tenancy Ref 48 and Rent Account 48 occurred and are now not shown on the home screen of user from office “MNC”. Also as you can see from Fig.40 the home screen of the user from office “EDM” was not changed.

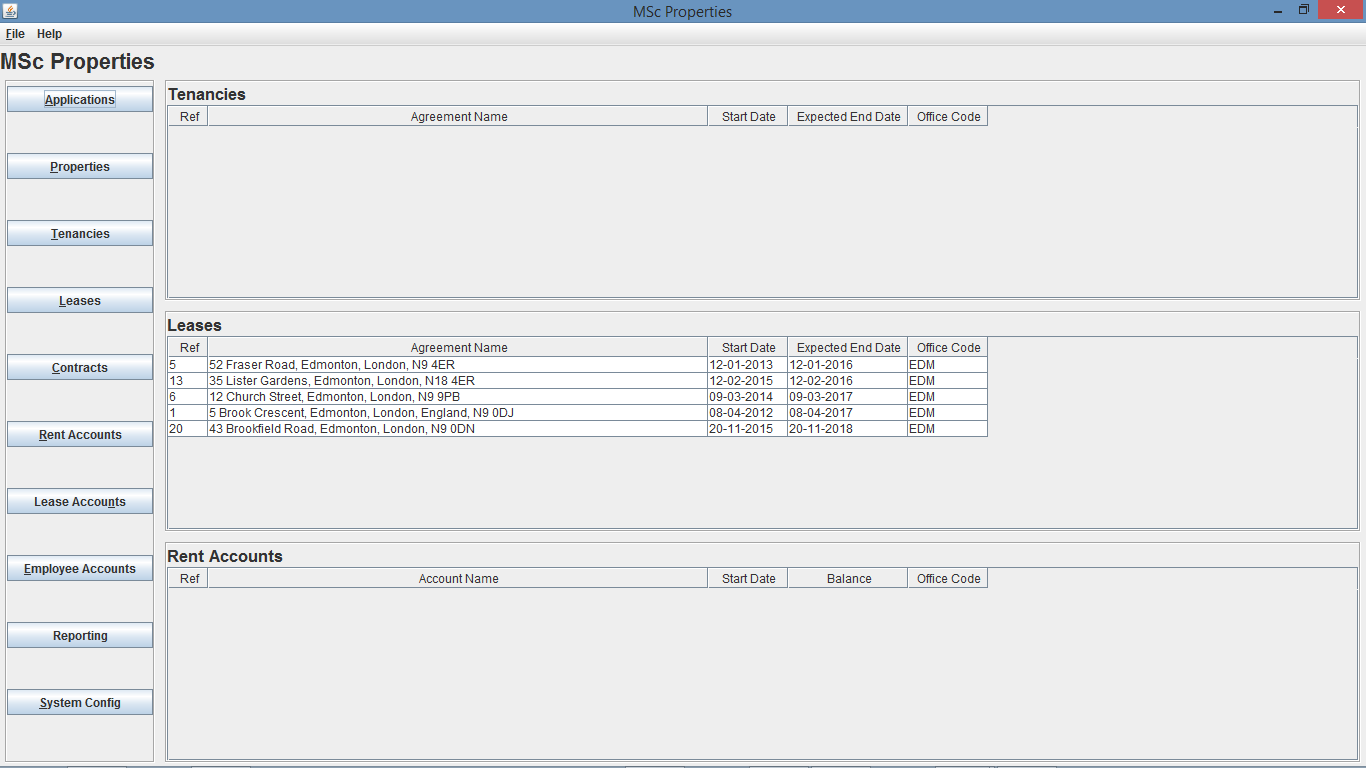


Fig. 40 - Home Screen of user from “EDM” office after deletion of Tenancy Ref 48

My final test is going to be to end a tenancy, again using the 3rd user account, and due to rent accounts being dependent on the tenancy, meaning the rent account cannot be current without the tenancy being current, so the ending of a tenancy will be reflected in both the tenancy and rent account tables.

I can end a tenancy through two ways, either by going to the tenancy details screen, or from the tenancy search screen. For this testing I am going to delete a tenancy through the tenancy search screen, again by navigating to it from the home screen by clicking the Tenancies button. Once I am there I have to search for the tenancy, and again as I need to select a tenancy from a particular office, I decide to use the standard search and search for tenancy 18, as this is from office “MNC”, but any search method could have been used.

I then need to right click on the tenancy I want to end, and select End Tenancy, a date entry dialog will then appear asking me for the end date, if I click OK, a yes/no dialog will then appear asking me to confirm the ending of tenancy 18, and if I select yes this will then end Tenancy 18 and confirmation dialog will appear as shown in Fig. 41.

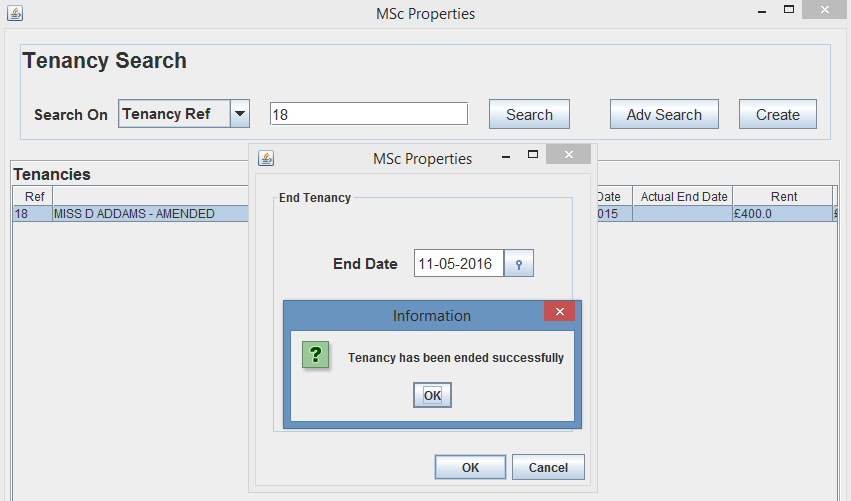


Fig. 41 – End Tenancy Dialog

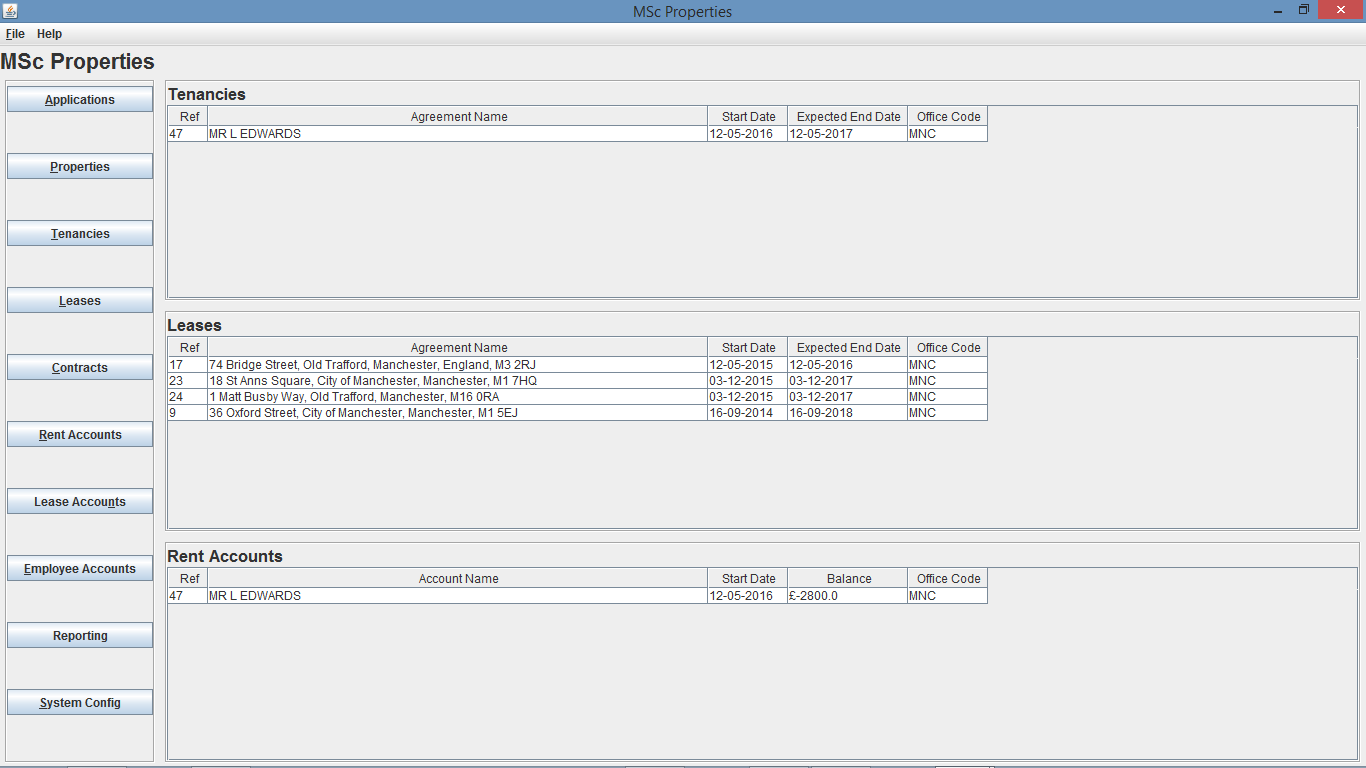


Fig. 42 - Home Screen of user from “MNC” office after ending Tenancy Ref 18

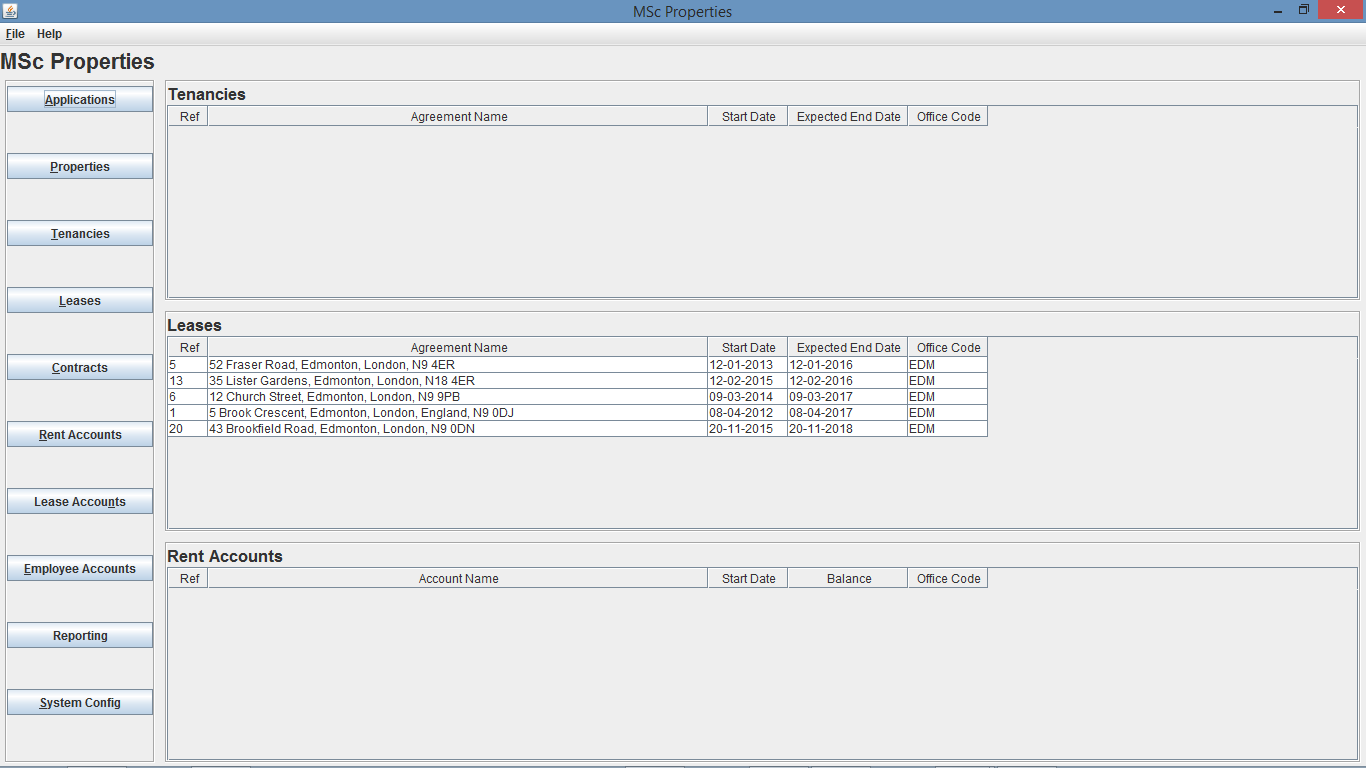


Fig. 43 - Home Screen of user from “EDM” office after ending Tenancy Ref 18

* 1. **Create, Update Delete – Lease**

The tests will continue on from the previous screens state in Fig. 42 and Fig. 43 and my first test is going to be to create a new lease using the 3rd user account that has read, write, update and delete privileges, which will update the leases table.

To create a lease, I navigated to the Leases search page by clicking on the Leases button from the home screen, I then clicked create from the lease search screen, where a lease create dialog box opens, I can then use the prop search button to search for the required property ref if not known.

If I enter all of the correct information ensuring the office I assign the lease to, is one of the offices I expect the update to occur on, for this test I decided to use the “MNC” office. Once I click the OK button, a yes/no create dialog will open for me to confirm creation of the lease, in which if I select yes a confirm dialog will appear confirming creation of the Tenancy, as you can see in Fig. 44.

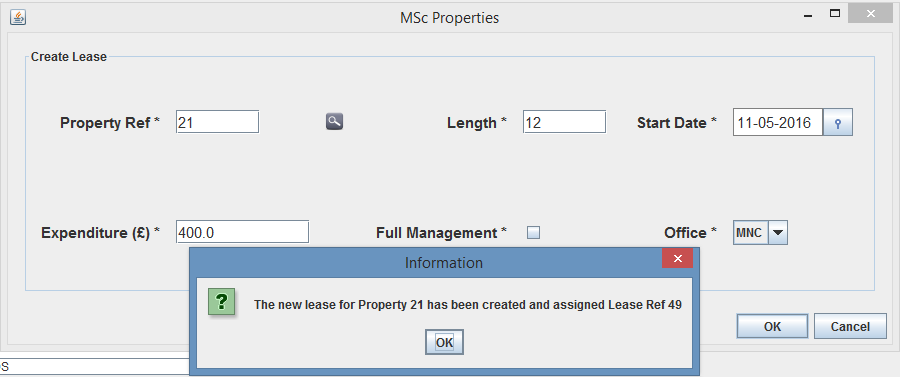


Fig. 44 – Lease creation Dialog

As you can see from Fig. 44, I selected “MNC” as the office and the newly created lease was assigned Lease Ref 49. So I am now expecting the user from office “MNC” to now show the newly created Lease on the home screen and in its correct position within the list depending on expected end date, however I will not see the creation on the home screen of the user from office “EDM” and this is shown in Fig. 45 and Fig. 46.

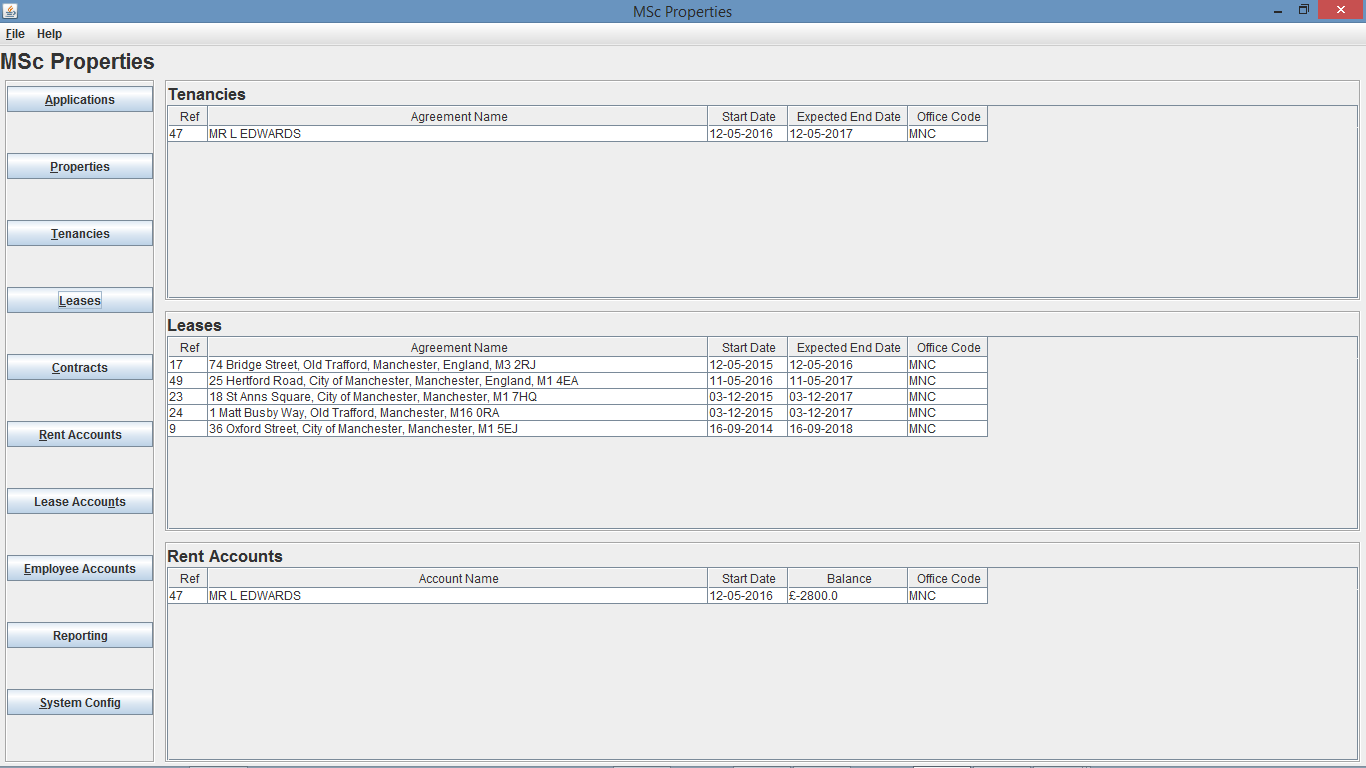


Fig. 45 - Home Screen of user from “MNC” office after creation of Lease Ref 49

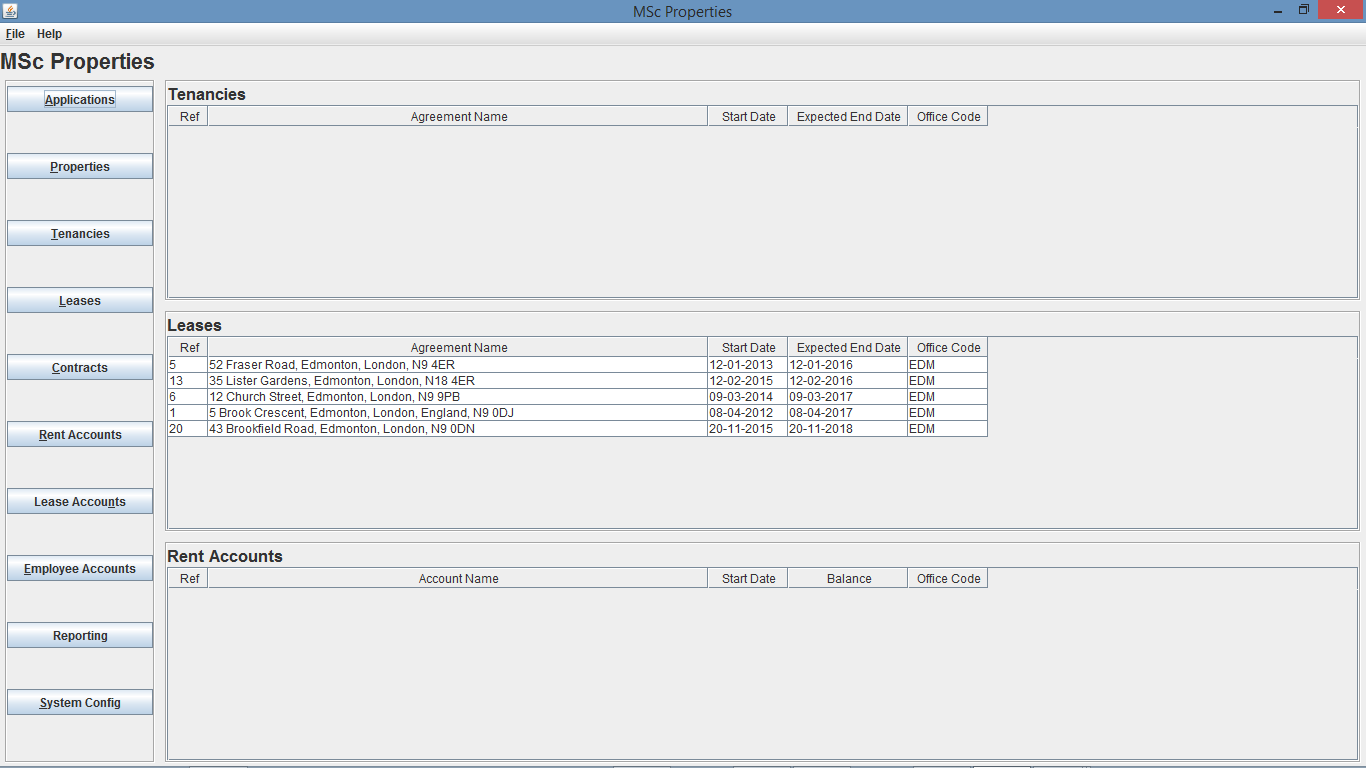


Fig. 46 - Home Screen of user from “EDM” office after creation of Lease Ref 49

My second test is going to be to update a lease, again using the 3rd user account which will be reflected in the leases table.

I can update a lease through two ways, either by going to the lease details screen, or from the lease search screen. For this testing I am going to update a lease through the lease search screen, again by navigating to it from the home screen by clicking the Leases button. Once I am there I have to search for the lease, and again as I need to select a lease from a particular office I decide to use the standard search and search for lease 17, as this is from office “MNC”, but any search method could have been used.

I then need to right click on the lease I want to update, and select Update Lease, and as explained for testing I decided to update Lease Ref 17 from both Fig. 45 and Fig. 46. As you can see from Fig. 47, I updated the length of the lease to 60 months, meaning the expected end date should change to 12-05-2020 resulting in the lease moving to the bottom of the list due to the lease being the last to potentially end out of the 5 leases for office “MNC”.

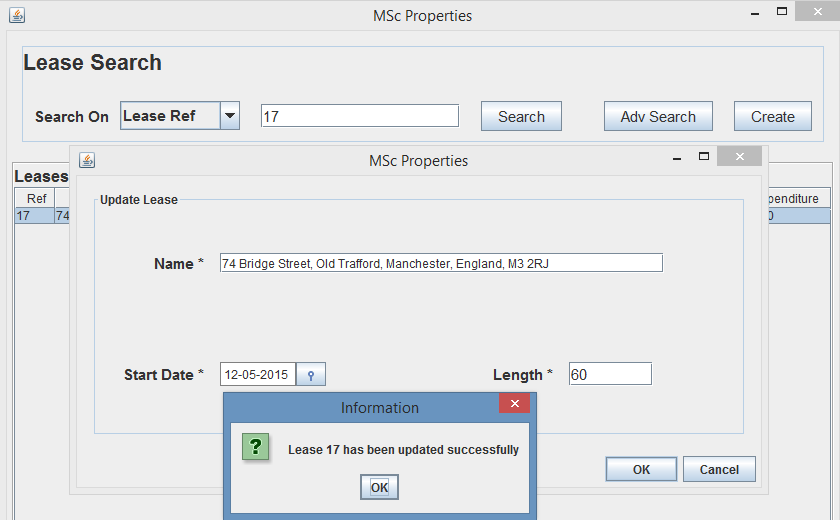


Fig. 47 – Lease Amendment Dialog

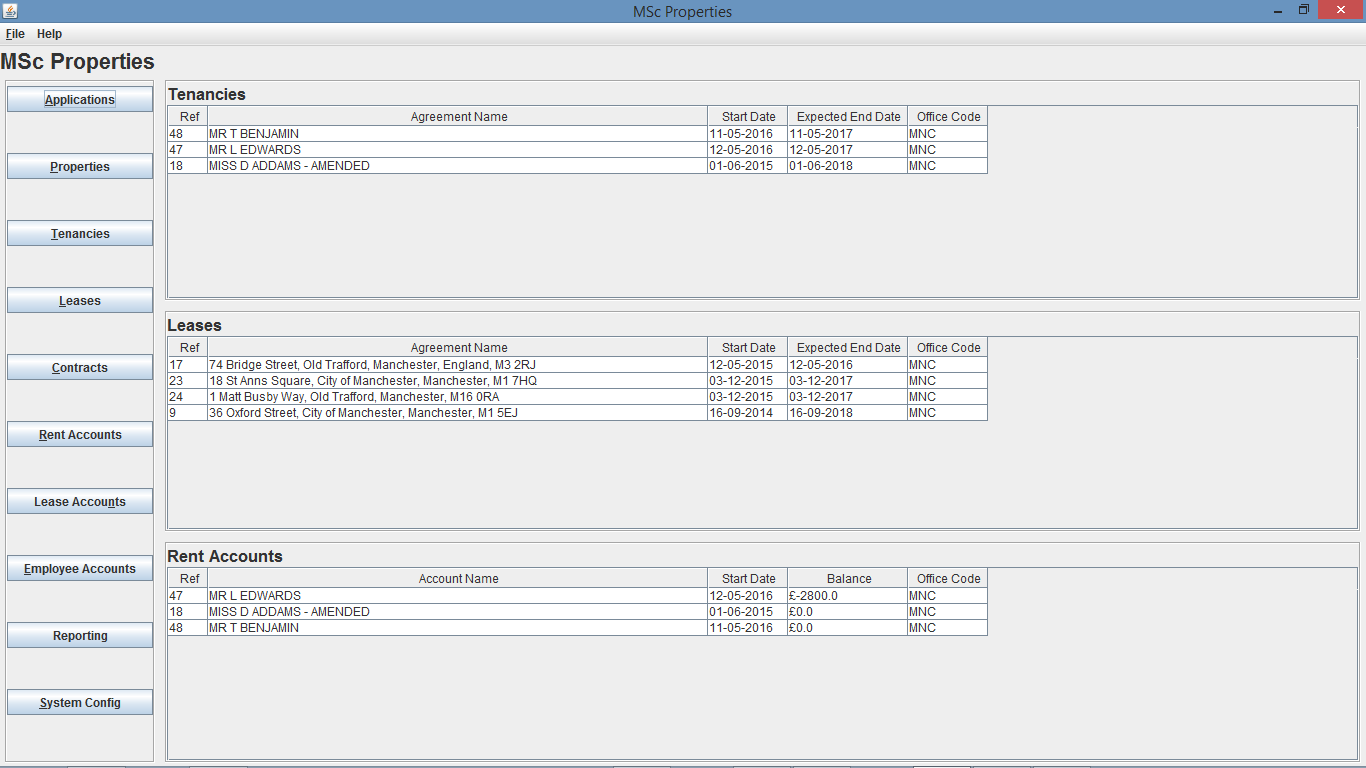


Fig. 48 - Home Screen of user from “MNC” office after amendment of Lease Ref 17

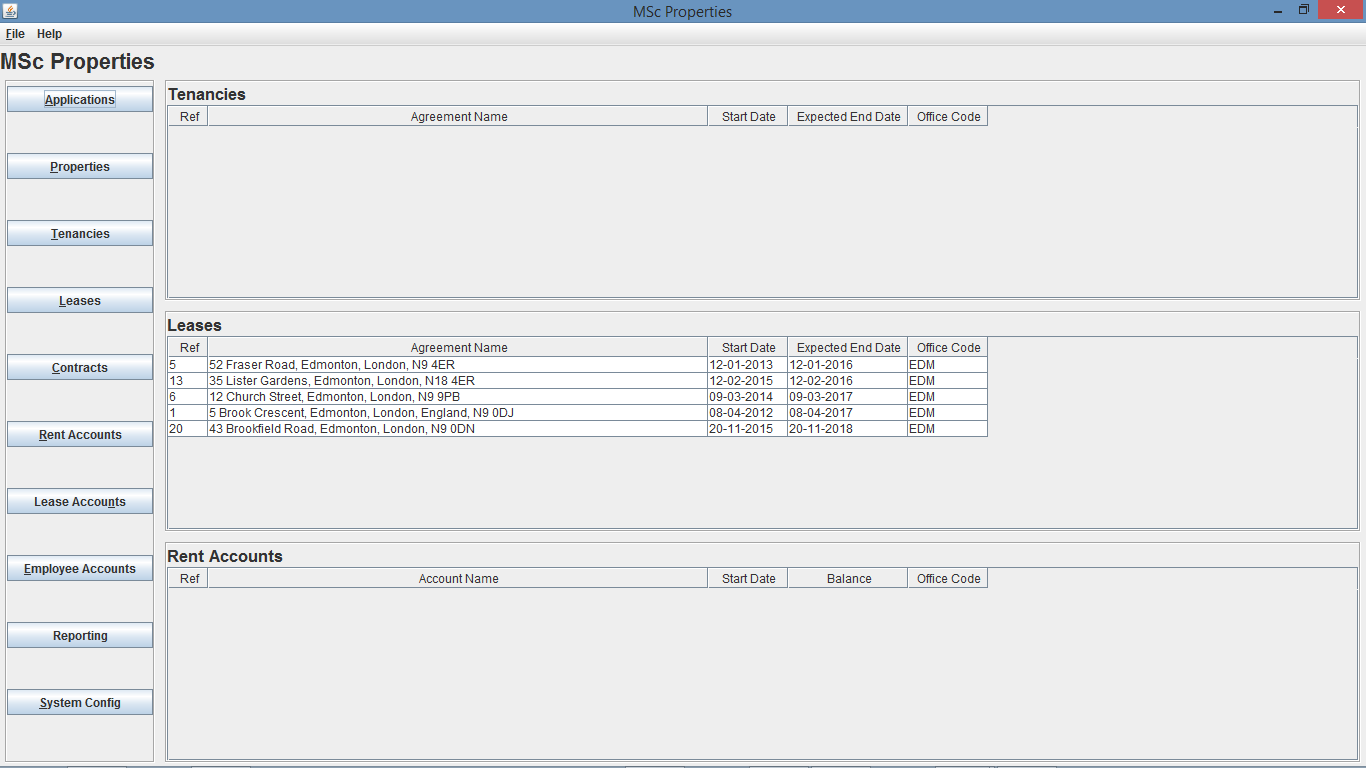


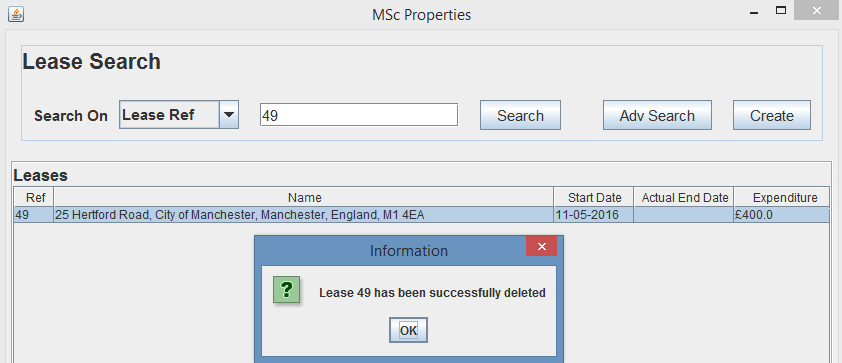
Fig. 49 - Home Screen of user from “EDM” office after amendment of Lease Ref 17

As explained I was expecting the user from office “MNC” to now show the amended Lease on the home screen, as well as the updated list position in the Leases and as expected I will not see the amendment on the home screen of the user from office “EDM” and this is shown in Fig. 48 and Fig. 49.

My third test is going to be to delete a lease, again using the 3rd user account, where the deletion of a lease will be reflected in the leases table.

I can delete a lease through two ways, either by going to the lease details screen, or from the lease search screen. For this testing I am going to delete a lease through the lease search screen, again by navigating to it from the home screen by clicking the Leases button. Once I am there I have to search for the lease, and again as I need to select a lease from a particular office, but that does not have any dependent records, I decide to use the standard search and search for lease 49 which was created earlier, as this is from office “MNC”, but any search method could have been used.

I then need to right click on the lease I want to delete, and select Delete Lease, a yes/no dialog will then appear asking me to confirm the deletion of lease 49, and if I select yes this will then delete Lease 49 and a confirmation dialog will appear as shown in Fig. 50.

 Fig. 50 – Lease Deletion Dialog

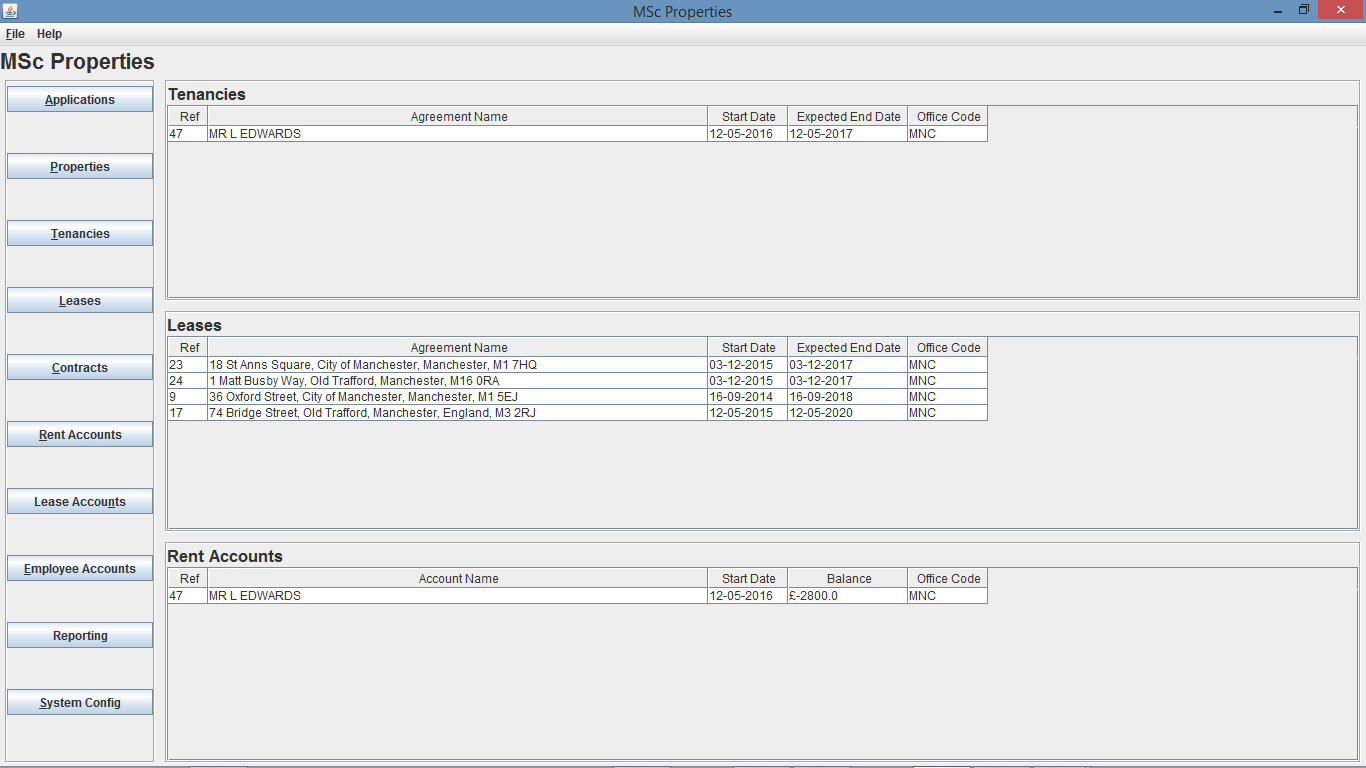


Fig. Fig 51 - Home Screen of user from “MNC” office after deletion of Lease Ref 49

As you can see from Fig. 51 the deletion of Lease Ref 49 occurred and are now not shown on the home screen of user from office “MNC”. Also as you can see from Fig.52 the home screen of the user from office “EDM” was not changed.

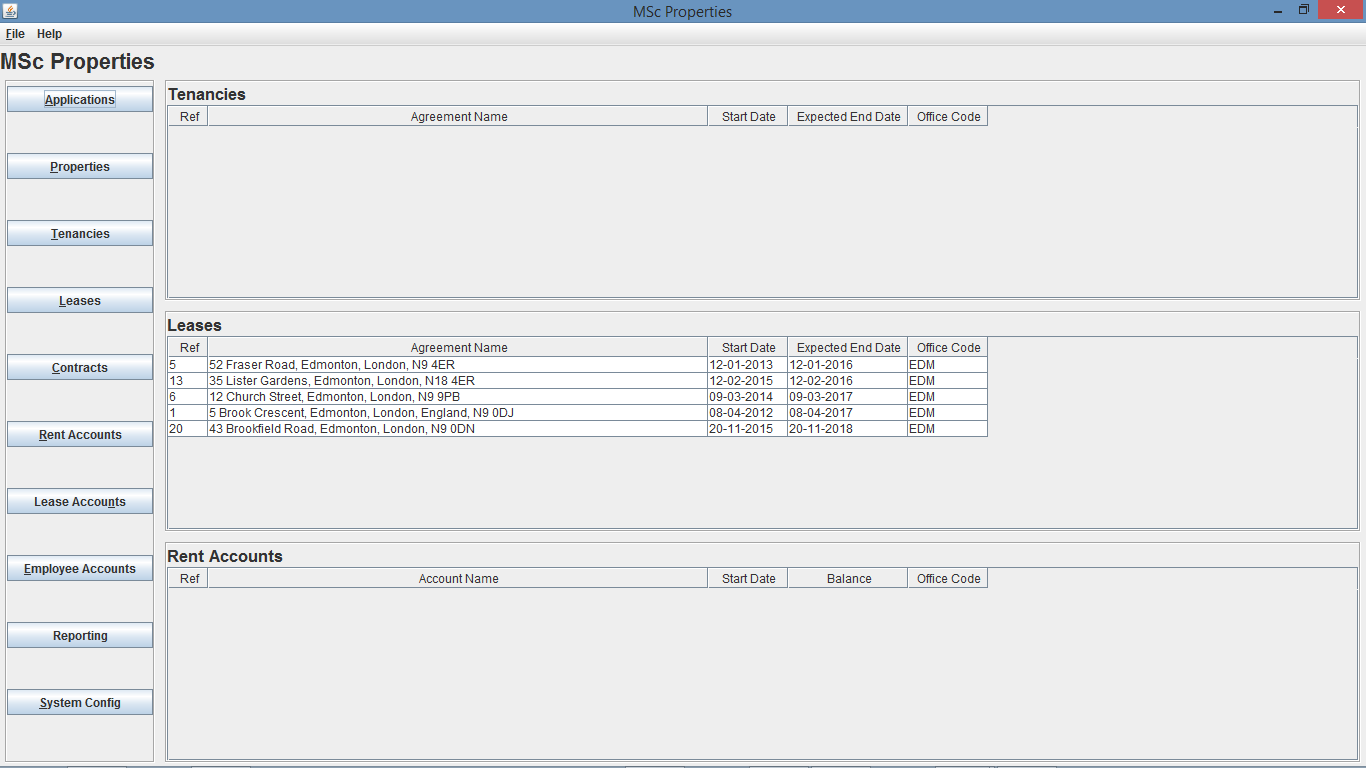


Fig. 40 - Home Screen of user from “EDM” office after deletion of Lease Ref 49

My final test is going to be to end a tenancy, again using the 3rd user account, and ending a lease will be reflected in the leases table.

I can end a lease through two ways, either by going to the lease details screen, or from the lease search screen. For this testing I am going to delete a lease through the lease search screen, again by navigating to it from the home screen by clicking the Leases button. Once I am there I have to search for the lease, and again as I need to select a lease from a particular office, I decide to use the standard search and search for lease 17, as this is from office “MNC”, but any search method could have been used.

I then need to right click on the lease I want to end, and select End Lease, a date entry dialog will then appear asking me for the end date, if I click OK, a yes/no dialog will then appear asking me to confirm the ending of lease 17, and if I select yes this will then end Tenancy 18 and a confirmation dialog will appear as shown in Fig. 53.

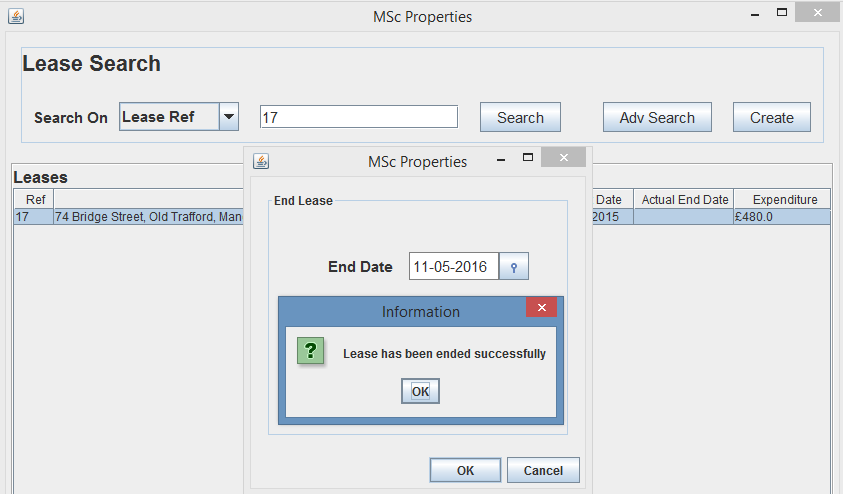


Fig. 53 – End Lease Dialog

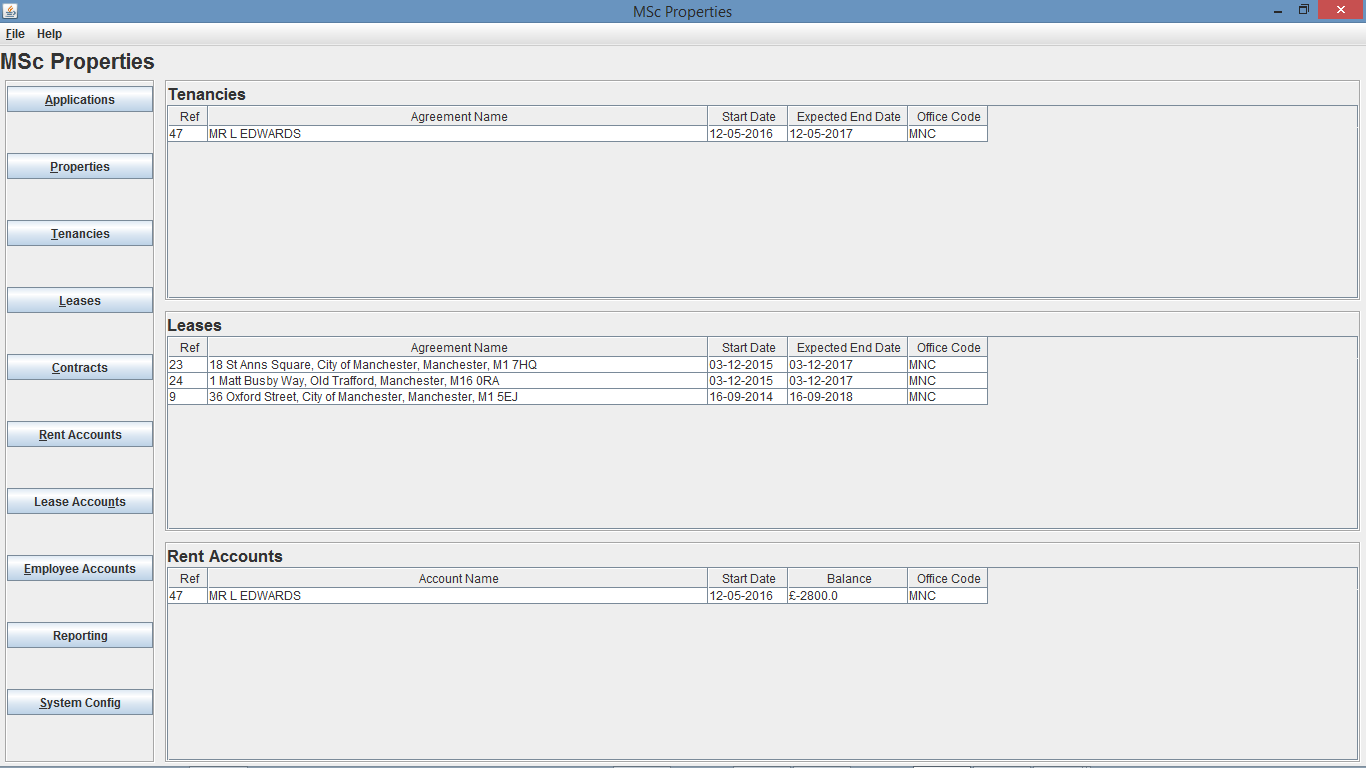


Fig. 42 - Home Screen of user from “MNC” office after ending Lease Ref 17

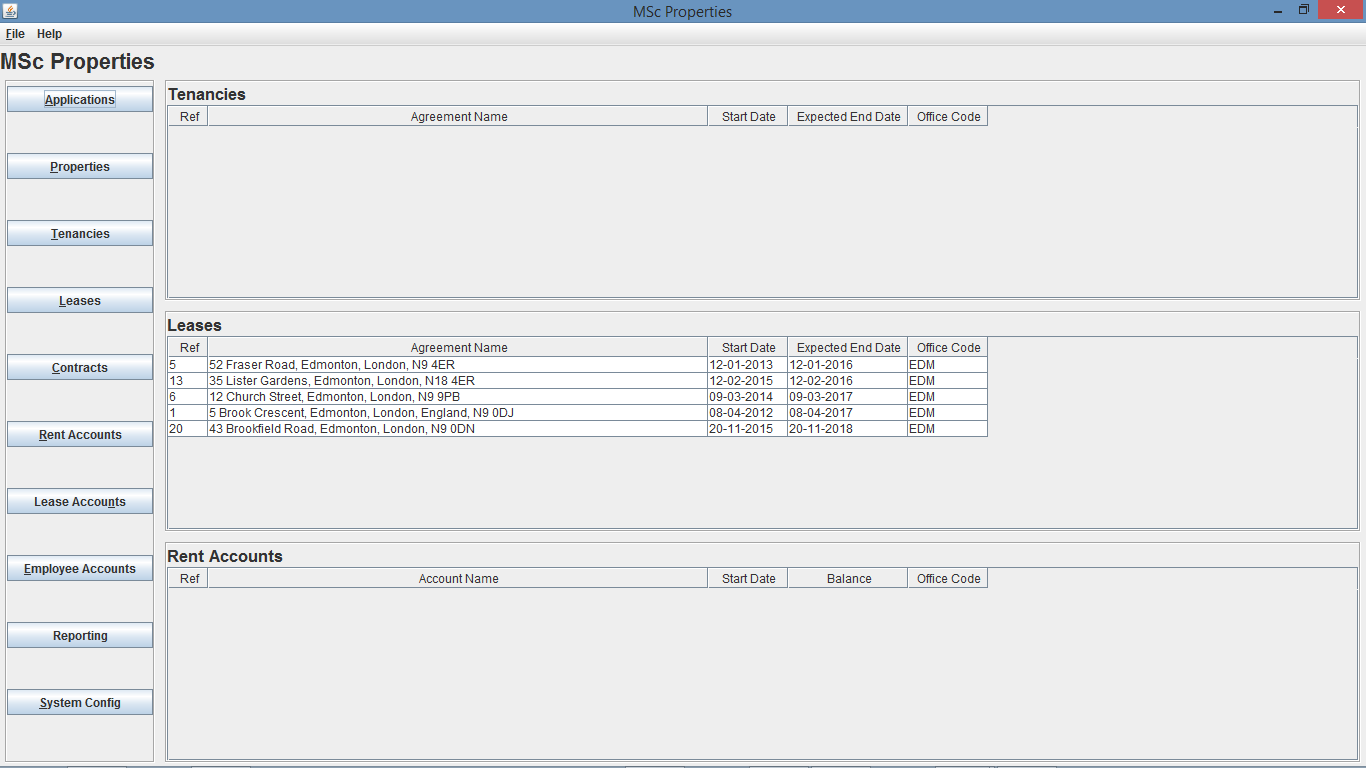


Fig. 43 - Home Screen of user from “EDM” office after ending Lease Ref 17

* 1. **Create Transaction**

My next final testing of the home screen will cover creating rent transactions for rent accounts to ensure the updates are sent through to the home screen of users who are from the office

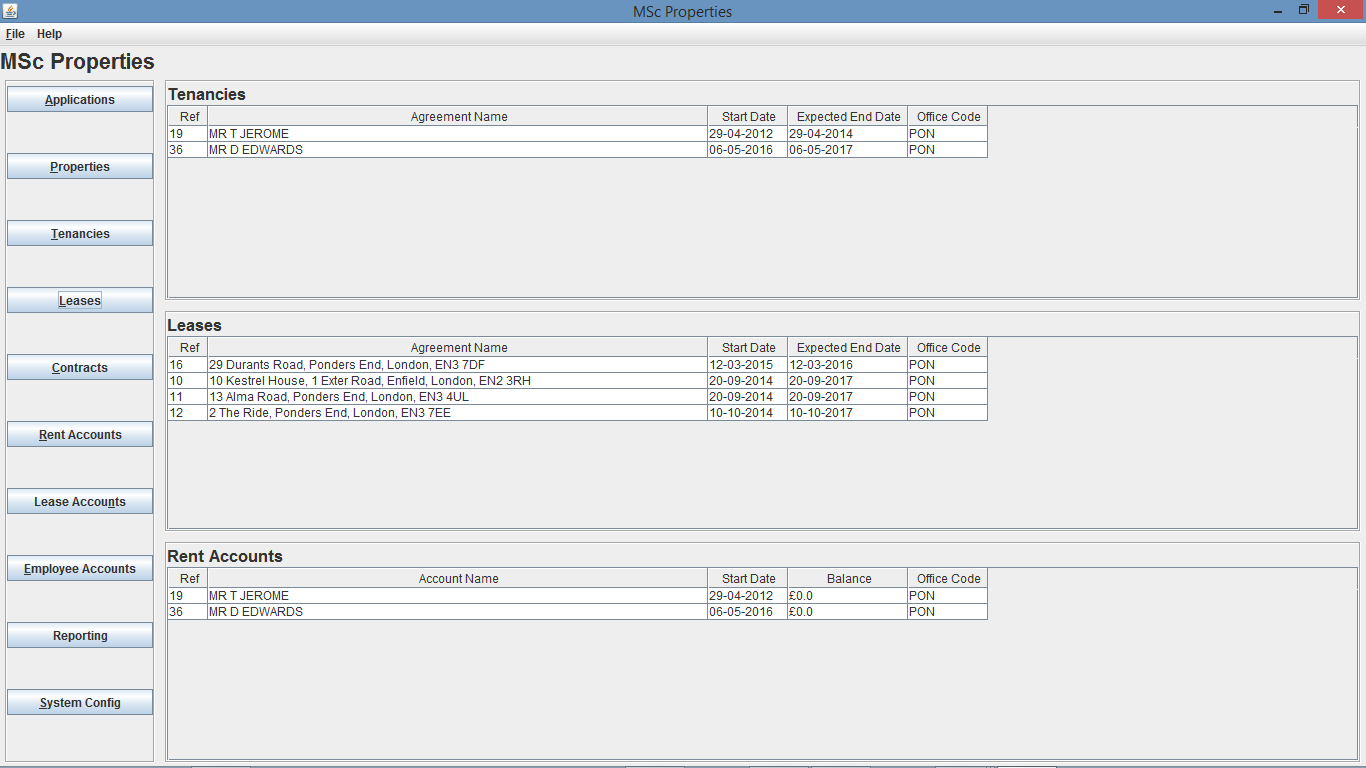


Fig. 44 – Home Screen of user from “PON” office prior to rent transaction creation

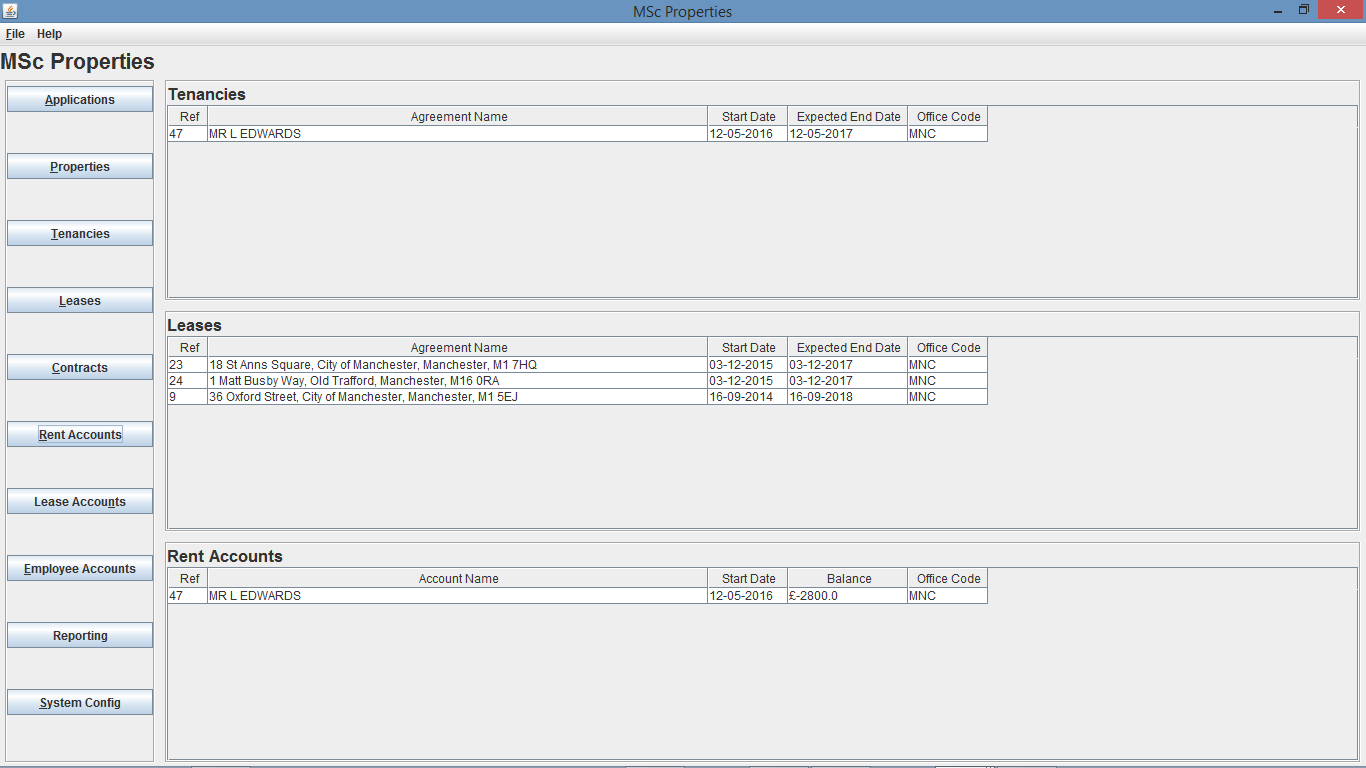


Fig. 45 – Home Screen of user from “MNC” office prior to rent transaction creation

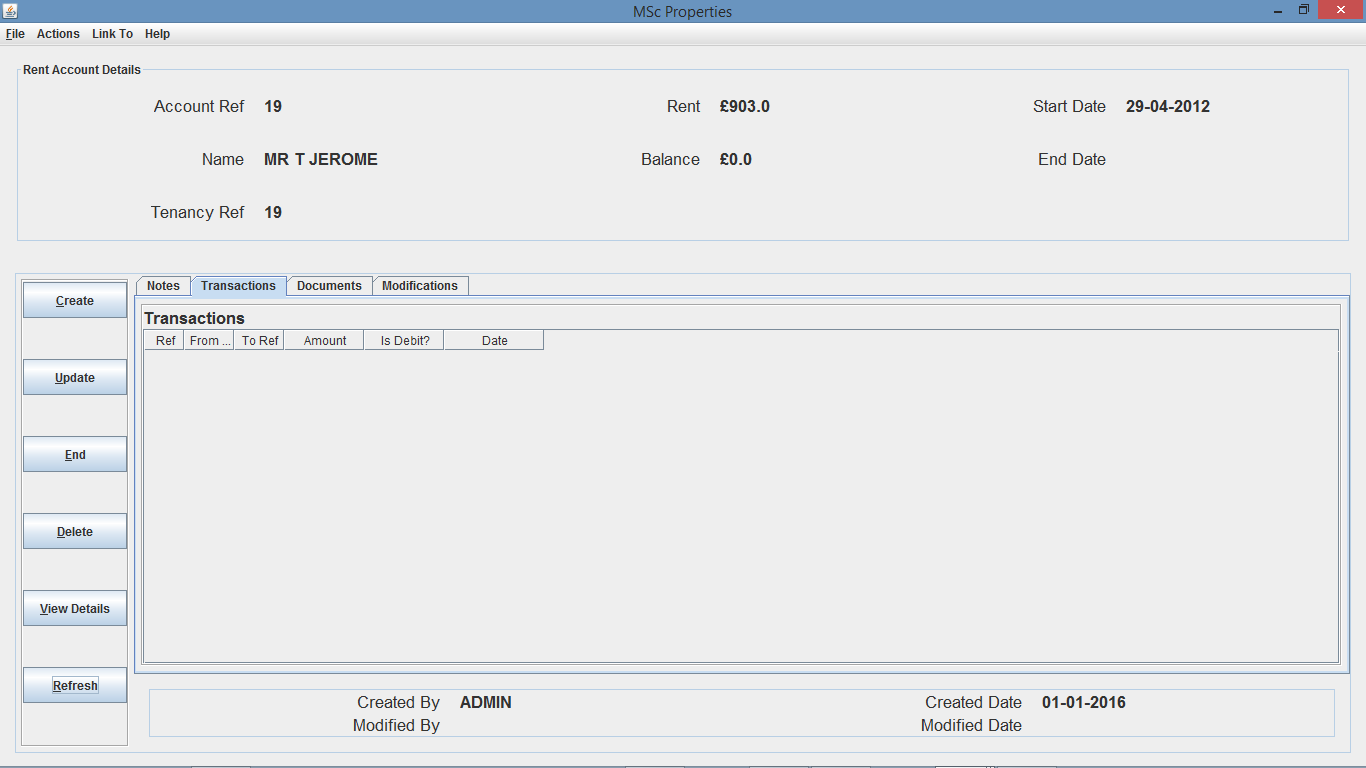


Fig. 46 – Rent Account Details Screen (on Transactions tab)

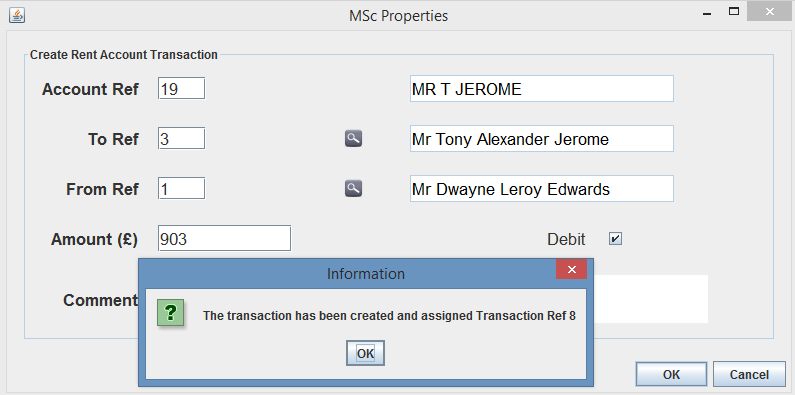


Fig. 47 – Create Transaction Dialog

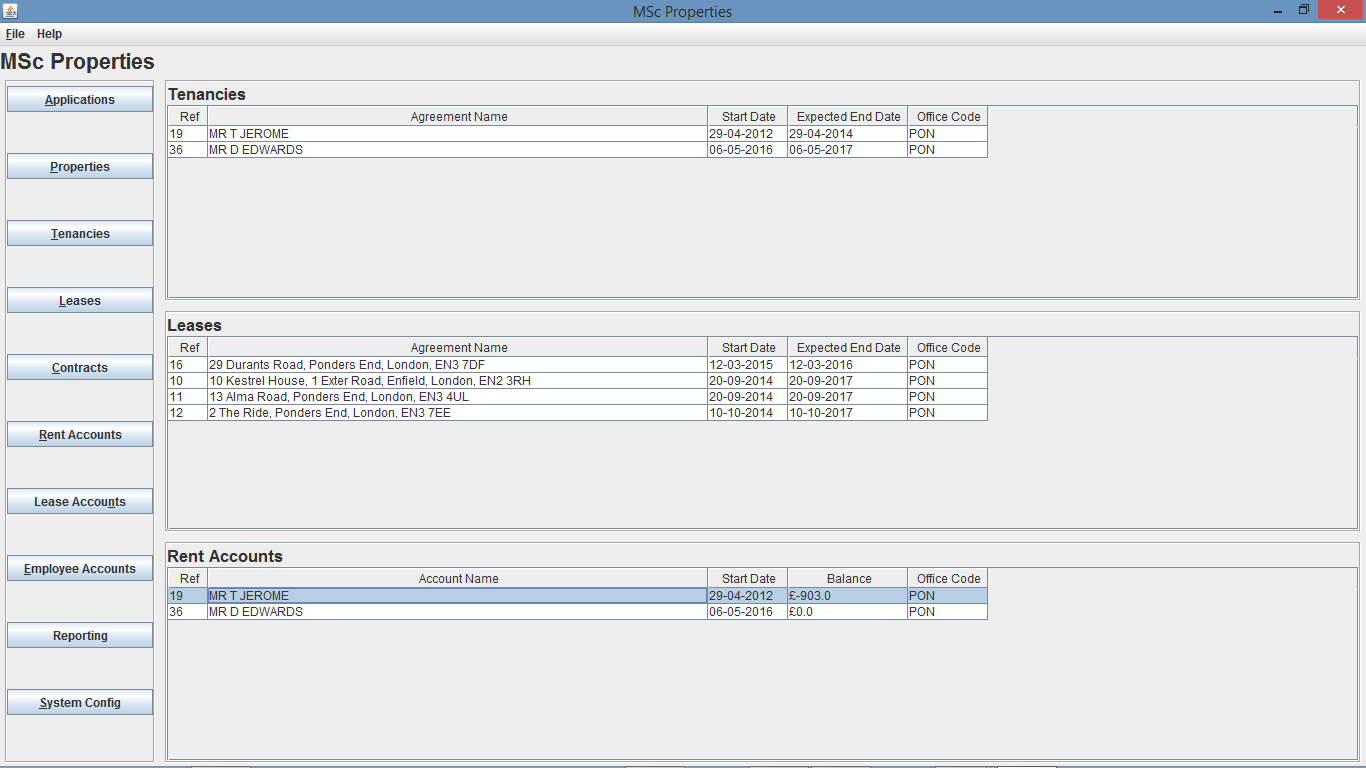


Fig. 48 - Home Screen of user from “PON” office after creation of rent transaction 8

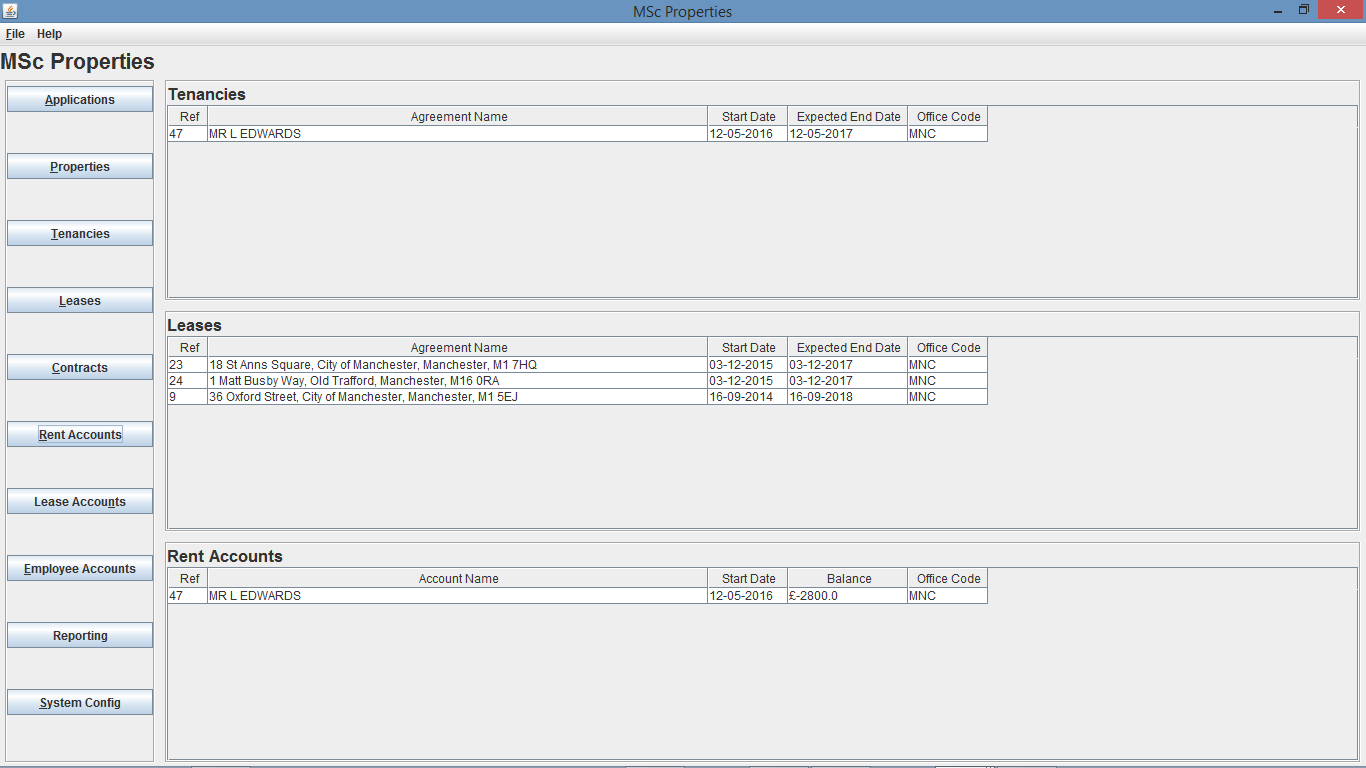


Fig. 49 – Home Screen of user from “MNC” office after creation of rent transaction 8

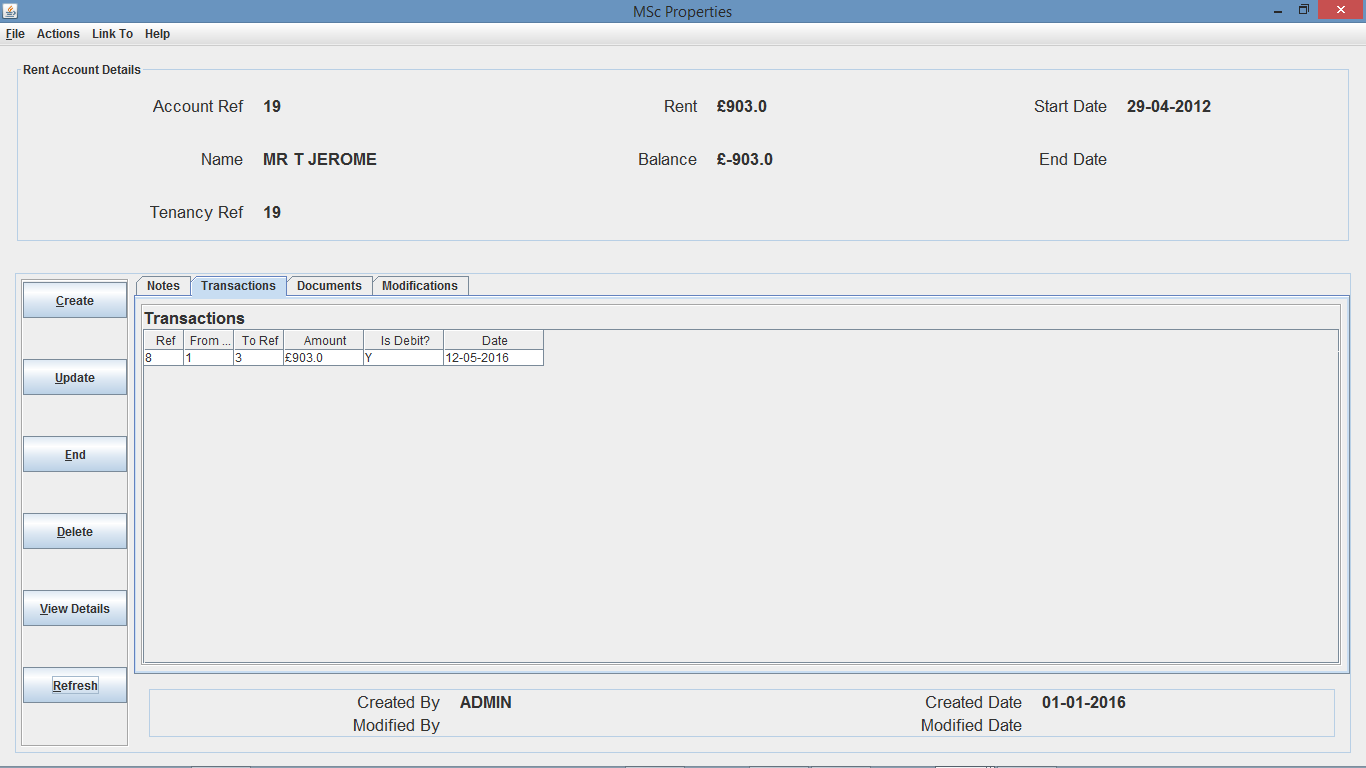


Fig. 50 - Rent Account Details Screen after creation of rent transaction 8 (on Transactions tab)

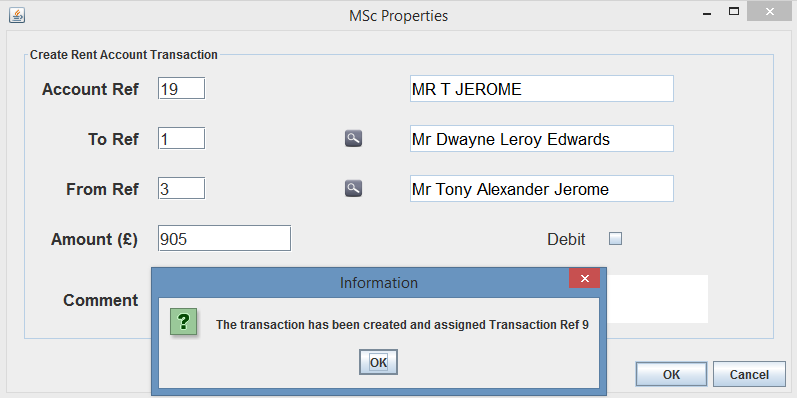


Fig. 51 – Create Transaction Dialog

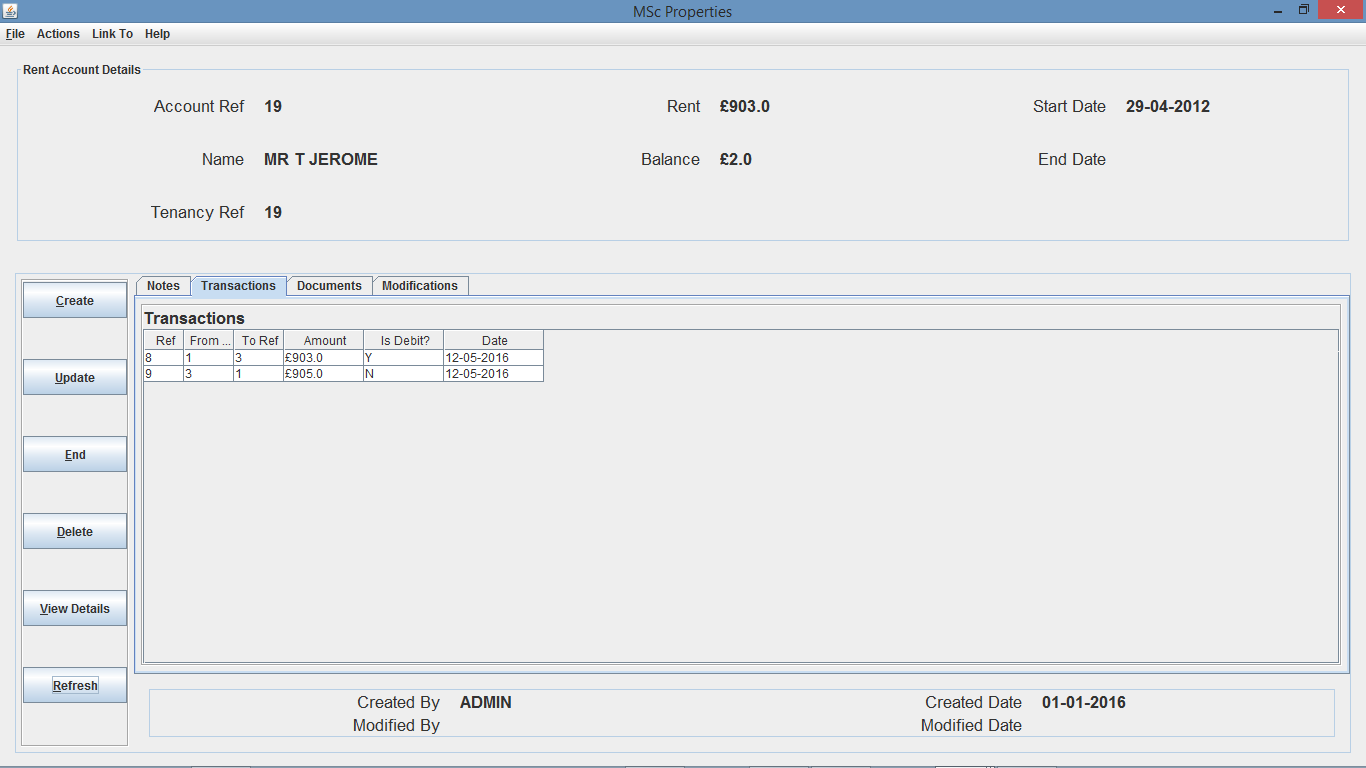


Fig. 52 – Rent Account Details screen after creation of rent transaction 9 (on transactions tab)

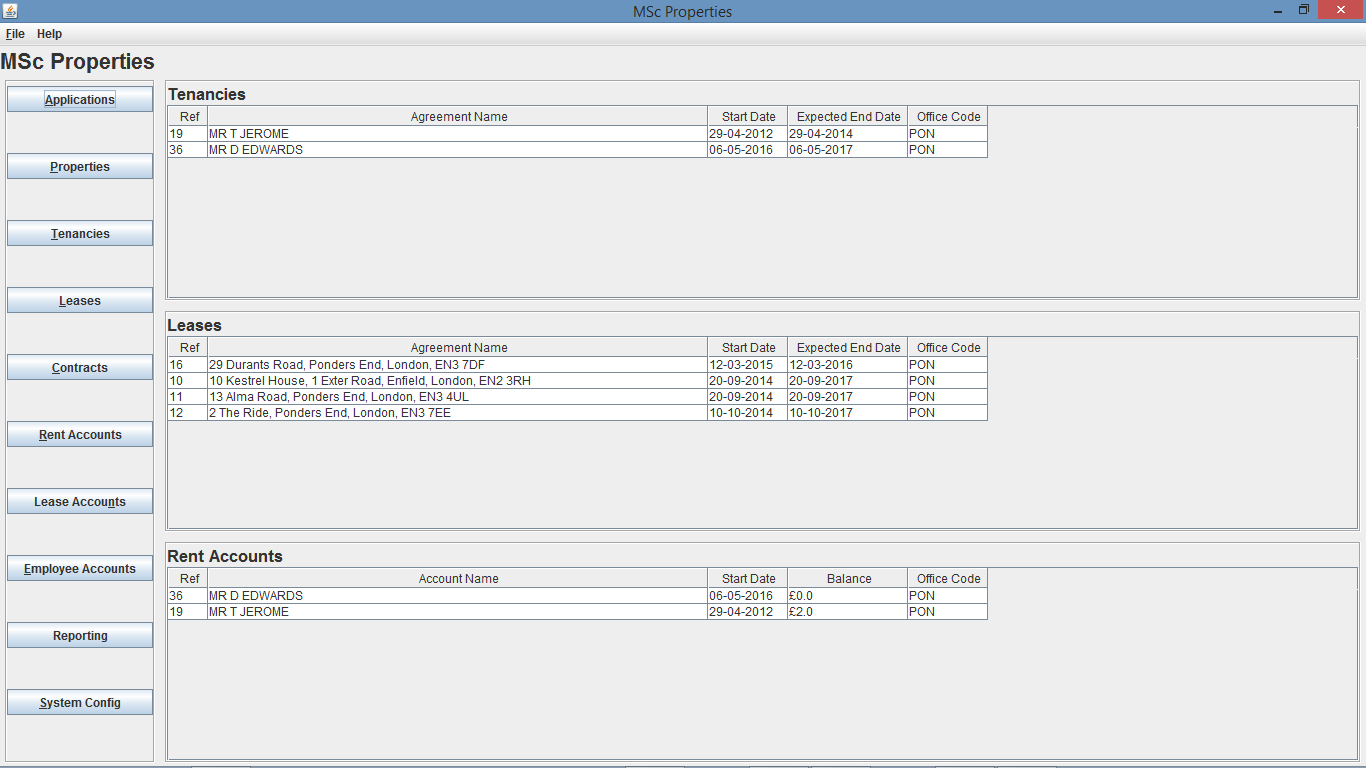


Fig. 53 – Home Screen of user from “PON” office after creation of rent transaction 9

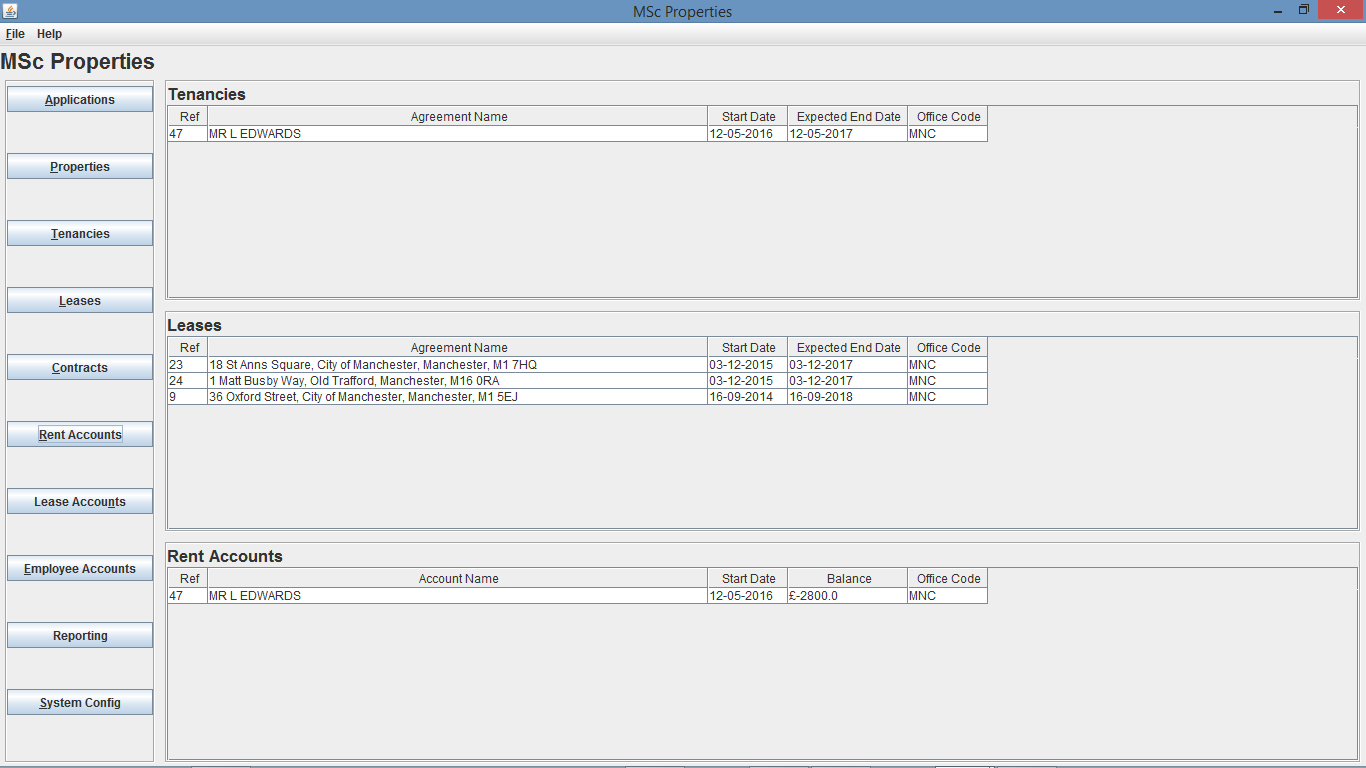


Fig. 54 – Home Screen of user from “MNC” office after creation of rent transaction 9

* 1. **Home Screen Limit**

1. **Create Note**
2. **Update Note**
3. **Delete Note**
4. **Document Management functionality**
5. **Advertise Business Services through Website**
6. **Service Requests through Website**