Meeting with Thief Team (25th November 2015)

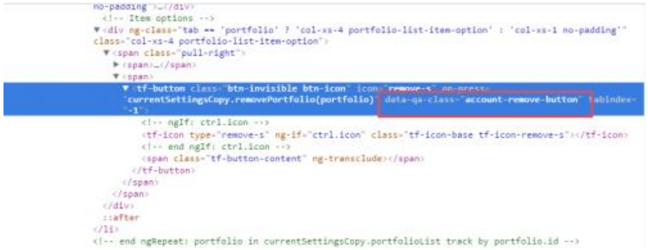
Wednesday, November 25, 2015 5:52 PM

ListBox changes

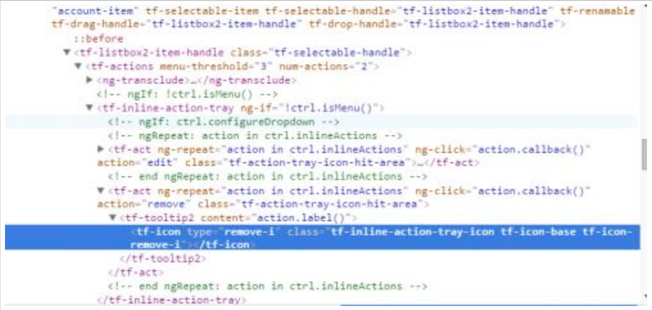
```
WeightOptionsColTabAddRemove.prototype.expandElementTree = function( elementPath, excludeElements ) {
   var arrElements = elementPath.split( '|' ), xpathExpandButton, i;
   var arrExcludedElements;
   if ( excludeElements === undefined ) {
      arrExcludedElements = undefined;
   } else {
      arrExcludedElements = excludeElements.split( '|' );
   }
   xpathExpandButton = this.xpathAvailableContainer;

   if ( arrElements.length === 1 ) {
      xpathExpandButton += '/ul/li/div[.="' +arrElements[0]+ '"]//a';
      element( by.xpath( xpathExpandButton ) ).click();
   } else {
      for ( i = 0; i < arrElements.length; i++ ) {
         xpathExpandButton += '/ul/li/div[.="' +arrElements[i]+ '"]//a';

         if ( arrExcludedElements === undefined ) {
            element( by.xpath( xpathExpandButton ) ).click();
         } else if ( arrExcludedElements.indexOf( arrElements[i] ) < 0 ) {
            element( by.xpath( xpathExpandButton ) ).click();
         } else {
            arrExcludedElements.splice( arrExcludedElements.indexOf( arrElements[i] ), 1 );
         }

         if ( i !== arrElements.length - 1 ) {
            xpathExpandButton += '/ancestor::li[1]';
         }
      }
   }
};
```

```
WeightOptionsColTabAddRemove.prototype.expandElementTree = function( elementPath, excludeElements ) {
   var arrElements = elementPath.split( '|' ), xpathExpandButton;
   var arrExcludedElements;
   if ( excludeElements === undefined ) {
      arrExcludedElements = undefined;
   } else {
      arrExcludedElements = excludeElements.split( '|' );
   }
   xpathExpandButton = this.xpathAvailableContainer;

   if ( arrElements.length === 1 ) {
      xpathExpandButton += '//*[@data-qa-class="available-tree-item"]' +
         '//*[normalize-space(.)="' +arrElements[0]+ '" and contains(@class, "no-wrap")]' +
         '/ancestor::*[contains(@class, "selectable-handle")]//*[contains(@class, "icon-tree")]';

      this.makeElementVisible( element( by.xpath( xpathExpandButton ) ) );

      // Expand the element
      element( by.xpath( xpathExpandButton ) ).click();

   } else {
      for ( var i = 0; i < arrElements.length; i++ ) {
         xpathExpandButton += '//*[@data-qa-class="available-tree-item"]' +
            '//*[normalize-space(.)="' +arrElements[i]+ '" and contains(@class, "no-wrap")]' +
            '/ancestor::*[contains(@class, "selectable-handle")]//*[contains(@class, "icon-tree")]';

         if ( arrExcludedElements === undefined ) {
            this.makeElementVisible( element( by.xpath( xpathExpandButton ) ) );

            // Expand the element
            element( by.xpath( xpathExpandButton ) ).click();
         } else if ( arrExcludedElements.indexOf( arrElements[i] ) < 0 ) {
            this.makeElementVisible( element( by.xpath( xpathExpandButton ) ) );

            // Expand the element
            element( by.xpath( xpathExpandButton ) ).click();
         } else {
            arrExcludedElements.splice( arrExcludedElements.indexOf( arrElements[i] ), 1 );
         }

         if ( i !== arrElements.length - 1 ) {
            xpathExpandButton += '/ancestor::*[contains(@class, "selectable-handle")]' +
'/following-sibling::*[@data-qa-class="available-tree-group"]';
         }
      }
   }
};
```

Special case where QA attributes are deleted



```
no-padding">…</div>
  <!-- Item options -->
▼<div ng-class="tab == 'portfolio' ? 'col-xs-4 portfolio-list-item-option' : 'col-xs-1 no-padding'"
class="col-xs-4 portfolio-list-item-option">
  ▼<span class="pull-right">
    ▶<span>…</span>
    ▼<span>
      ▼<tf-button class="btn-invisible btn-icon" icon="remove-s" on-press=
      "currentSettingsCopy.removePortfolio(portfolio)" data-qa-class="account-remove-button" tabindex=
      "-1">
          <!-- ngIf: ctrl.icon -->
          <tf-icon type="remove-s" ng-if="ctrl.icon" class="tf-icon-base tf-icon-remove-s"></tf-icon>
          <!-- end ngIf: ctrl.icon -->
          <span class="tf-button-content" ng-transclude></span>
        </tf-button>
      </span>
    </span>
  </div>
  ::after
</li>
<!-- end ngRepeat: portfolio in currentSettingsCopy.portfolioList track by portfolio.id -->
```



```
"account-item" tf-selectable-item tf-selectable-handle="tf-listbox2-item-handle" tf-renamable
tf-drag-handle="tf-listbox2-item-handle" tf-drop-handle="tf-listbox2-item-handle">
   ::before
  ▼<tf-listbox2-item-handle class="tf-selectable-handle">
   ▼<tf-actions menu-threshold="3" num-actions="2">
     ▶<ng-transclude>…</ng-transclude>
      <!-- ngIf: !ctrl.isMenu() -->
     ▼<tf-inline-action-tray ng-if="!ctrl.isMenu()">
        <!-- ngIf: ctrl.configureDropdown -->
        <!-- ngRepeat: action in ctrl.inlineActions -->
      ▶<tf-act ng-repeat="action in ctrl.inlineActions" ng-click="action.callback()"
      action="edit" class="tf-action-tray-icon-hit-area">…</tf-act>
        <!-- end ngRepeat: action in ctrl.inlineActions -->
      ▼<tf-act ng-repeat="action in ctrl.inlineActions" ng-click="action.callback()"
      action="remove" class="tf-action-tray-icon-hit-area">
        ▼<tf-tooltip2 content="action.label()">
            <tf-icon type="remove-i" class="tf-inline-action-tray-icon tf-icon-base tf-icon-
            remove-i"></tf-icon>
          </tf-tooltip2>
        </tf-act>
        <!-- end ngRepeat: action in ctrl.inlineActions -->
      </tf-inline-action-tray>
```

Radio Button changes

```
AttributionOptionsGroupingsAddRemoveNewReference.prototype.getRadioButton = function( btnName ) {
   // Variable(s)
   var xpathRadioButton;

   if ( btnName.toLowerCase() === 'new' ) {
      xpathRadioButton = this.xpathNewReference + '//*[@data-qa-id="new-radio-button"]';
   } else if ( btnName.toLowerCase() === 'reference' ) {
      xpathRadioButton = this.xpathNewReference + '//*[normalize-space(.)="' +btnName+ '"]/descendant::
*[contains(@class, "radio")]';
   } else {
      xpathRadioButton = this.xpathNewReference + '//*[@data-qa-id="radio-button-'
+btnName.toLowerCase().replace( /\s/g, '-' )+ '"]';
   }

   return element( by.xpath( xpathRadioButton ) );
};
```

```
AttributionOptionsGroupingsAddRemoveNewReference.prototype.getRadioButton = function( btnName ) {
   // Variable(s)
   var xpathRadioButton;

   if ( btnName.toLowerCase() === 'new' ) {
      xpathRadioButton = this.xpathNewReference + '//*[@data-qa-id="new-radio-button"]/
*[contains(@class,"radio")]';
   } else if ( btnName.toLowerCase() === 'reference' ) {
      xpathRadioButton = this.xpathNewReference + '//*[normalize-space(.)="' +btnName+ '"]/descendant::
*[contains(@class, "radio")]';
   } else {
      xpathRadioButton = this.xpathNewReference + '//*[@data-qa-id="radio-button-'
+btnName.toLowerCase().replace( /\s/g, '-' )+ '"]';
   }

   return element( by.xpath( xpathRadioButton ) );
};
```

Checkbox changes

```
AttributionOptionsGroupingsAddRemoveNewReference.prototype.getCheckbox = function( cbName ) {
   var xpathCheckbox = this.xpathNewReference + '//*[@data-qa-id="checkbox-'
+cbName.toLowerCase().replace( /\s/g, '-' )+ '"]';
```

```
AttributionOptionsGroupingsAddRemoveNewReference.prototype.getCheckbox = function( cbName ) {
   var xpathCheckbox = this.xpathNewReference + '//*[@data-qa-id="checkbox-'
+cbName.toLowerCase().replace( /\s/g, '-' )+ '"]' +
```

return element(by.xpath(xpathCheckbox));
};
	'/*[contains(@class,"checkbox")]';
	return element(by.xpath(xpathCheckbox));
	};

Meeting with Bryan (thief)

Tuesday, June 14, 2016 12:07 AM

- Original issue why helper constants came into picture.
- How frequent they'll update the thief-helper library based on our requests to add new functions.
- Referring spec and source code to understand the function usage.
- Documentation is not in synch as this is our first point of reference. Also, the description of some functions seems to be copied.

Subject	**RE: Application developers should not have to put non domain qa-classes on elements**
From	Bryan C. Chaiklin
To	Bhupender Agarwal
Cc	Adam Herrmann; Ken D. Morris; Eyme Ann Mathew
Sent	Friday, June 10, 2016 12:29 AM

Yea, that's not great. It does not really seem worth having the constants in that case. I can't think of a way to improve it right now.

I missed your documentation questions before:

1. Documentation for existing thief helper functions are not clear enough for everyone to get started on and one has to refer to the source code to use the function. Showing sample usage of these functions would ease the process of converting our existing functions to use thief library.
   a. **The examples would be nearly identical to what we have in our spec files such as https://gitlab.factset.com/web-infrastructure/thief-angular-testing/blob/master/src/accordion/index.spec.js. It would be nice if this could be automatically inlined with the main documentation, but we don't have such a tool. For now can we just advice testers to refer to these specs. I actually find it easier to learn by reading these specs and completely ignoring the primary documentation.**
2. Existing documentation seems to be outdated as the source code seems to have different function names compared to what's mentioned in the documentation.
   a. **Please file issues for the specific cases. That's odd, I would not expect that to occur.**
3. Some functions directly perform action on element rather than returning the reference of it
   For example, in my function **clickNewButton** above, the line transferBox.source['items'](); directly perform click on + icon button in application. But we also have scenarios where we just want to verify the existence of **tf-transferbox-source-control-items** rather than performing action on it.
   a. **You could just run a query for `tf-transferbox-source-control-items` to check that is exists. We could also provide a new top level function. I'm not sure which is better. Eventually I think we are moving to a world where we have less abstractions for controls like this so you would most likely be writing DOM queries for specific elements in the future, so maybe stick with that in this case.**

Thank you,
Bryan

From: Bhupender Agarwal
Sent: Thursday, June 09, 2016 12:14 PM
To: Bryan C. Chaiklin

Cc: Adam Herrmann; Ken D. Morris; Eyme Ann Mathew
Subject: RE: Application developers should not have to put non domain qa-classes on elements

Hi Bryan,

One possible approach to update helper constants to target either of thief directive could be, separating them with pipe symbol. But, this approach will complicate the page object functions and will drastically increase the number of lines of code(as it affects every single po function).

For example, if HelperConstasnts.TransferBox has to target either ‘tf-transferbox’ or ‘tf-new-transferbox’ then my approach to update the constant file would look like this:

```
var HelperConstants = function() {
            this.TransferBox = 'tf-transferbox| tf-new-transferbox ';
            this.DropDown = 'tf-dropdown';
            this.Menu = 'tf-menu';
}

module.exports = new HelperConstants();
```

Now, in order to make this work, my function clickNewButton() will look like this:

```
/
*********************************************************************************
***/
/* Function:
clickNewButton
           */
/* Description: This function is used to click on "New (+)"
button.                                                           */
/* Return: Returns the reference of "New"
button.                                                                              */
/
*********************************************************************************
***/
TileOptionsGroupings.prototype.clickNewButton = function() {

          var arrConstantsTransferBox = HelperConstants.TransferBox.split( '|' );

       baseXpath = '//*[@data-qa-id="options-container" and not(contains(@class, "ng-hide"))]//';

       xpathTransferBox = baseXpath + arrConstantsTransferBox[0]
       for ( var i = 1; i < arrConstantsTransferBox.length; i++ ) {
                 xpathTransferBox += '|' + baseXpath + arrConstantsTransferBox[i];
       }

          // Get the reference of transferbox
          var transferBox = new TestHelpers.Transferbox( by.xpath( xpathTransferBox ) );

          // Click on "Add(+ icon)" button
          return transferBox.source['items']();

};
```

Please let me know if you have any better approach towards this? This method will increase the complexity of PO functions and script execution time.
Also, can you please look into the documentation concerns I mentioned in my previous thread?

Thanks,
Bhupender Agarwal
Senior Team Leader, QA Automation
FactSet Systems India Pvt. Ltd.
Work: +91 91 00 931356
Cell: +91 90 30 337694

From: Bryan C. Chaiklin
Sent: Wednesday, June 08, 2016 7:02 PM
To: Bhupender Agarwal
Cc: Adam Herrmann; Ken D. Morris; Eyme Ann Mathew
Subject: RE: Application developers should not have to put non domain qa-classes on elements

Is it possible using this technique to allow HelperConstants.TransferBox to target either `tf-transferbox` or let's say `tf-new-tranferbox`. How could you modify the constant to allow this? I think this approach in general gets most of its value if it is flexible enough to do that type of thing. Otherwise, I think the value is minimal.

From: Bhupender Agarwal
Sent: Tuesday, June 07, 2016 3:14 PM
To: Bryan C. Chaiklin
Cc: Adam Herrmann; Ken D. Morris; Eyme Ann Mathew
Subject: RE: Application developers should not have to put non domain qa-classes on elements

Hi Bryan,

Sorry for the delay in responding back to your last email as I was on vacation.

I have used a few helper functions from **trasnferbox** class to convert some of my utility functions. I have also created a sample helper constant file which resolves to the thief directives.
Here is how my sample helper constant file looks like:

```
var HelperConstants = function() {
            this.TransferBox = 'tf-transferbox ';
            this.DropDown = 'tf-dropdown';
            this.Menu = 'tf-menu';
}

module.exports = new HelperConstants();
```

Converted functions:

```
/
****************************************************************************************
***/
/* Function:
clickNewButton
            */
/* Description: This function is used to click on "New (+)"
button.                                                       */
```

```
/* Return: Returns the reference of "New"
button.                                                                          */
/
**************************************************************************************
***/
TileOptionsGroupings.prototype.clickNewButton = function() {

            // XPATH to point to transferbox
            var xpathTransferBox = '//*[@data-qa-id="options-container" and not(contains(@class, "ng-
hide"))]//' + HelperConstants.TransferBox;

            // Get the reference of transferbox
            var transferBox = new TestHelpers.Transferbox( by.xpath( xpathTransferBox ) );

            // Click on "Add(+ icon)" button
            return transferBox.source['items']();

};

/
**************************************************************************************
*****/
/* Function:
selectNewDropDownItem
               */
/* Description: This function is used to select required option from "New' drop
down.                                                */
/* Params: 1. itemName -> Name of the item to
select.                                                                          */
/* Return: Returns any reject message if any, while selecting the required
item.                                                     */
/
**************************************************************************************
*****/
TileOptionsGroupings.prototype.selectNewDropDownItem = function( itemName ) {
   // Variable(s)
            var xpathDropDownItem = '//' + HelperConstants.DropDown + '//' +HelperConstants.Menu;
            var menuItem = new TestHelpers.Menu( by.xpath( xpathDropDownItem ) );

            // Calling helper function
            return menuItem.selectItemByText( itemName );

};
```

Lines highlighted above depicts the helper constants usage in customized XPATH. Please let me know if this is what you were expecting?
Do you want me to show you the sample DOM and the script where I am using these functions?

If this approach for helper constant is correct, I have few queries:

1. Where the helper constants will be maintained?
2. Does helper constant file get downloaded when we install thief-helper module?
3. In the sample helper constants above, I am wrapping the constants under **HelperConstants** function, would you be following the same?

> **NOTE:** If I do not wrap these constants into a function, customized XPATH will not be able to identify it and will throw exception.

While working on these helper functions I made some observation as follows:

1. Documentation for existing thief helper functions are not clear enough for everyone to get started on and one has to refer to the source code to use the function. Showing sample usage of these functions would ease the process of converting our existing functions to use thief library.
2. Existing documentation seems to be outdated as the source code seems to have different function names compared to what's mentioned in the documentation.
3. Some functions directly perform action on element rather than returning the reference of it
   For example, in my function **clickNewButton** above, the line transferBox.source['items'](); directly perform click on **+** icon button in application. But we also have scenarios where we just want to verify the existence of **tf-transferbox-source-control-items** rather than performing action on it.

Please let me know if anything is not clear here. I am open to meet and discuss regarding the same.

Thanks,
Bhupender Agarwal
Senior Team Leader, QA Automation
FactSet Systems India Pvt. Ltd.
Work: +91 91 00 931356
Cell: +91 90 30 337694

From: Bryan C. Chaiklin
Sent: Friday, May 20, 2016 9:30 PM
To: Bhupender Agarwal; Eyme Ann Mathew
Cc: Rebecca Karger; Adam Herrmann; Ken D. Morris
Subject: RE: Application developers should not have to put non domain qa-classes on elements

1. It would probably be something like `testHelpers.selectors.TEXTBOX`. That would simply resolve to `tf-textbox`, which can then be combined into your more complicated selector.
2. If the attributes are not useful they should be removed, but it is not a priority to do so. I'd leave that up to the application engineers to decide when/if they want to clean those up.
3. Yes.

If we want to move forward this I'd prefer that you test it locally by defining your own constants that reference the element selectors, before we update thief-angular-testing. I want to make sure everything works as both parties expect. After that if you can file an issue against thief-angular-testing recapping this discussion and showing your tested samples we should be able to move forward.

Thanks,
Bryan

From: Bhupender Agarwal
Sent: Friday, May 20, 2016 10:57 AM
To: Bryan C. Chaiklin; Eyme Ann Mathew
Cc: Rebecca Karger; Adam Herrmann; Ken D. Morris
Subject: RE: Application developers should not have to put non domain qa-classes on elements

Hi Bryan,

That sounds perfect to me. Adding constant from your end would reduce the scripter's work to identify appropriate location to add qa attributes. With these new constants added, we can rely on it to initiate thief helper functions and as you said it'll be more robust as we do not have to worry about change in the component’s parent structure. However, I have few questions:

1. What naming convention will you follow to name these constants?
2. Will it be ok to retain existing qa attributes in the application code though we are not going to rely on it to locate thief components?
3. Would you be adding the constants used for each components in the thief-angular-testing documentation?

As we are discussing about thief helper functions, I would like to bring to your notice that documentation for thief helpers seems to be out of synch. For example, when I checked source file for Accordion class (https://gitlab.factset.com/web-infrastructure/thief-angular-testing/blob/0.5.0/src/accordion/index.js) it has methods – isSectionOpenedById(), _getSectionById() and toggleSectionById() but the documentation (https://gitlab.factset.com/web-infrastructure/thief-angular-testing#Accordion) shows it has methods isOpened() and toggle().

Please let me know if you need any other information from us.

Thanks,
Bhupender Agarwal
Senior Team Leader, QA Automation
FactSet Systems India Pvt. Ltd.
Work: +91 91 00 931356
Cell: +91 90 30 337694

From: Bryan C. Chaiklin
Sent: Friday, May 20, 2016 5:33 PM
To: Bhupender Agarwal; Eyme Ann Mathew
Cc: Rebecca Karger; Adam Herrmann
Subject: Application developers should not have to put non domain qa-classes on elements

Bhupender, Eyme,

This email is a follow up to a discussion that Rebecca, Adam and I had regarding qa-classes. Adam had shown some qa-classes being added to an application here:
https://gitlab.factset.com/analytics-libraries/account-dialog/merge_requests/19/diffs

We don’t want application developers to add classes like `textbox` and `checkbox` to `tf-textbox` and `tf-checkbox`. This is not adding any unique information that is relevant to their application. It is just boilerplate for them. Instead, I think we should do one of the following have QA use the already existent generic element selectors of `tf-textbox` and `tf-checkbox`. Presumably after selecting an element like this, the next step is to initiative the Thief Helper. If the Thief Helper was initialized on an element that matched `data-qa-class=”textbox”`, but was not a `tf-textbox` it would most likely fail anyways.

If we want to, we could define constants in `thief-angular-testing` for these element selectors. This would have the added benefit that if we were to once again in the future have a new version of an element (`tf-textbox2`) we could make that constant match `tf-textbox|tftextbox2`. We would also have to ensure that the helper was updated to handle both. This has the potential to reduce the number of reasons that could cause the tests to need to be modified.

Let me know your thoughts.

Thanks,
Bryan

Discussion on thief vs application code changes

Wednesday, June 22, 2016 4:41 PM

Subject	**Discussion on thief vs application code changes**
Date and Location	Thursday, June 23, 2016 9:30 PM - 10:30 PM, Hyderabad 506 VideoConference; Norwalk 601 206 VideoConference
Attendees	Eyme Ann Mathew; Ken D. Morris; Thomas Crescenzi; David Seah; Evan Wolbach; Rebecca Karger; Hyderabad 506 VideoConference; Norwalk 601 206 VideoConference;
Message	Hi All, I am blocking your calendar to discuss about the changes seen in application DOM due to thief version updates. I have gone through last 6 month changes made by my team for each thief version released and found that those changes are more related to restructure in application DOM but not due to thief version updates. Thus, even if we implement thief-helper functions that'll not reduce the maintenance work we do. Hence, wanted to discuss the other possible approach we might take to reduce maintenance work. Thanks, Bhupender

Questions:

- After going through past 6 month script changes, I figured out that most of the changes observed in DOM were application side changes made in reference to thief version upgrades.
- I have seen that engineering team started using new thief components replacing the old component which resulted in script maintenance. For example, button -> tf-button, ui, li changed to tf-listbox-group, tf-listbox-item respectively etc.
- There are many components which do not have qa attribute (probably because it was controlled by thief **or** the HTML equivalent elements were not present in files when we were adding qa attributes) due to which script is brittle to changes made to existing attributes we relied to find the element reference. In addition, we build customized XPATH such that it is traced from the parent thus requiring us to use relationship concept between the elements in the DOM. If we have unique qa attributes for each element we can use the same to identify elements.
-
- While trying to convert sample script to use helper functions, I found that there are many functions which are missing from the library which we might need to interact with component on webpage. For example, in Listbox class we have method only for expanding the group under Available section but we might need function to collapse and also functions to verify the status of group (if it is already expanded/collapse).
- We have scenarios where we need only reference of particular component but would not like to perform any action on it. For example, we may want to hover over a textbox or any other component on webpage just to switch focus from one component to another, we may need the list of options in the drop down rather than selecting one from it etc.
- When component name changes in thief library would you be updating the thief-helper class names to reflect the same? If so, that would invite maintenance work again.

Notes

Points to Discuss

Friday, October 07, 2016 5:58 PM

Deekshith:

- Able to guide new hires on Agile Project.
- Active participation in AT4.
- Decision to split available resources for different agile boards.
- Helped maintaining code standards.
- Active participation in Meetings conducted to find solutions to Automation challenges.
- Took initiative to document about Automation Workflow w.r.t RPD.
- Explored and documented usage of slick-grip api functions.
- Smooth transitioning of Security Analysis to Core team.
- Actively participated in giving training to team on using GIT.
- Initiative to document FAQs related to automation tools which would help for business continuity.
- Able to manage and support other agile boards like PDB, BPM2 (was part of agile for a while).
- Kept team spirit high which helped people to be more productive.
- Good collaboration with Engineers, PDs and Analysts.
- Good work ethics.
- Amicable person which helps to create healthy relationship with team members.
- Unbiased constructive feedback.
- Focused on improving process.

Tribhuvan:

- Able to manage and assist people working on different agile boards.
- Helped team to understand the required standards and implement the same.
- Gave regular feedback to address the issue sooner than later.
- Actively participated in meeting and added valuable points which helped to find solution to automation challenges.
- Good relation with analysts and stakeholders.
- Took initiative to discuss about SIT to engineers and analyst.
- Insisted engineers to create more unit test so as to catch bugs at early stage.
- Researched on protractor-cucumber and gave demo to Jay.
- Helped analysts to write better feature files.
- Focused on maintaining QA standards.
- Actively worked on Asset Allocation project while managing the team.
- Extended probation of Komal and Ranvijay and came with plan to improve their performance.
- Wise decision to take Prashanth for PS.
- Worked on checking automation feasibility of GIPS and provided report for the same.
- Respects individual preferences and guide them accordingly.
- Good knowledge on QA Process and hence volunteered to train campus hires.

Thief grid discussion

Friday, October 14, 2016 4:10 PM

- Need way to access multi-headers using api.
- Data returned from function call should be in consistent format.

Wednesday, April 26, 2017 6:12 PM

- Thief-Angular:
  - https://gitlab.factset.com/web-infrastructure/thief-angular-testing/issues/56 - Request to have collapse() function for virtual list box.
- Thief-Grid:
  - Existing helpers are not supporting when we have secondary headers (found recently).

Automation Failure Best Practice

Tuesday, November 14, 2017 7:25 PM

Meeting Date: 11/14/2017 9:30 PM
Location: Norwalk 601 204, Divyasree 1109
Link to Outlook Item: *click here*
Invitation Message
Participants

- Angela Yacovelli (Meeting Organizer)
- Rajul Shrivastava
- Bhupender Agarwal
- Priyanka Muddalapuram
- Asha Veerapaneni
- Alisa C. Berry
- Vasu Nakka
- Surendra Babu Parchuru

Points to discuss

Submitting re-run on failure

1. Configuration was wrong during first schedule.
2. If issue is intermittent either with environment
3. Test Not Run shouldn't be the reason for a re-run.
4. If application was slow during first instance and is behaving normal now given
    a. You have confirmed with engineer regarding the first instance.

Filing RPDs

1. Application works fine but still script failed.
2. Received Job Terminated email with reason as logs not found. File TCR RPD else it should be against App QA product.

Meeting with Jim

Tuesday, November 28, 2017 2:21 PM

- Hiring people with technical background even for Associate level.
- Test coverage by product wise should be known to testers. Identify the buggier functionality and target your testing accordingly.
- Testing of functionality should be owned by single person rather than dividing them among Manual Testers, Functional Analyst and Automation Scripter.
- Promotion expectation.
- Revising QA Standards
    - Num. of verifications which should be part of each test case.
    - Current limit of max. num. of steps in a case is 25 which should be reduced to 15.
- Testing should be focused to uncover bugs and thus should invest time where bugs are usually appear.