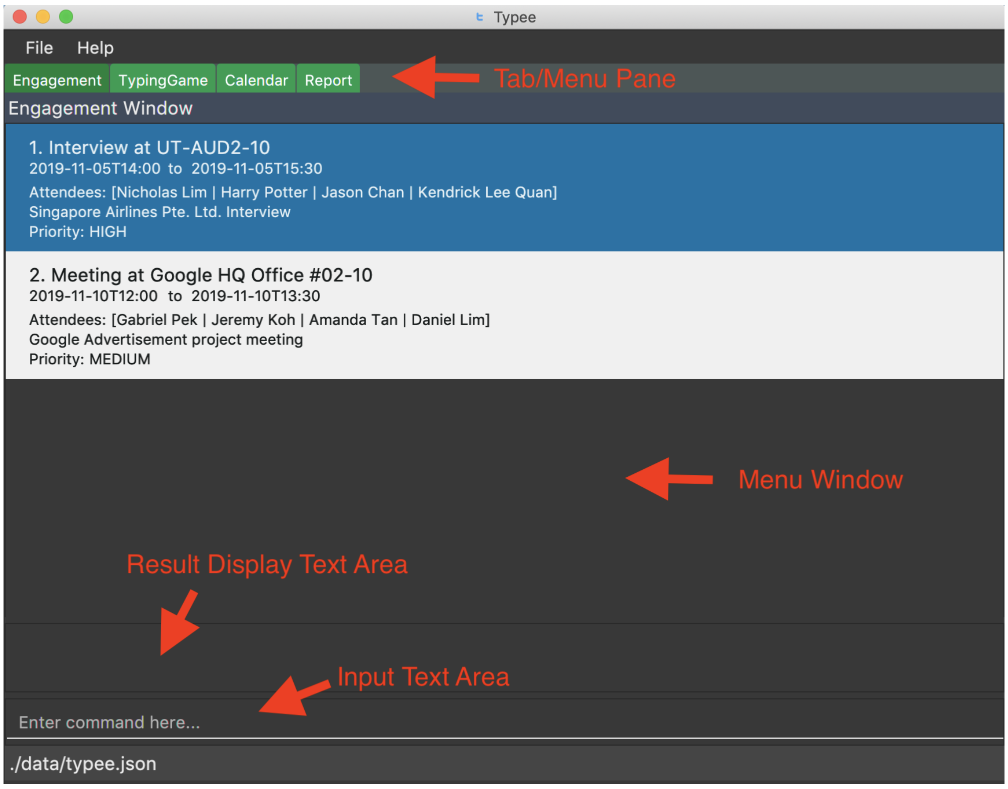
Ko Gi Hun - Project Portfolio

CS2103T Team Project: Typee

Overview

Typee is an engagement management application built on JavaFX framework, where users can manage engagements and schedule accordingly. Our main users are targeted in corporation's receptionists and secretaries whose main tasks is to manage and arrange engagements.



1. Interview at UT-AUD2-10
2019-11-05T14:00 to 2019-11-05T15:30
Attendees: [Nicholas Lim | Harry Potter | Jason Chan | Kendrick Lee Quan]
Singapore Airlines Pte. Ltd. Interview
Priority: HIGH

2. Meeting at Google HQ Office #02-10
2019-11-10T12:00 to 2019-11-10T13:30
Attendees: [Gabriel Pek | Jeremy Koh | Amanda Tan | Daniel Lim]
Google Advertisement project meeting
Priority: MEDIUM

Figure 1. default window of Typee upon start-up

Main UI is composed of 4 parts;

- **Tab/Menu Pane**: List of Name/Feature/Menu is displayed here.
- **Menu Window**: Each feature/menu window is displayed and can be interacted.
- **Command Output Display**: Command output after execution displays here.
- **Command Input text field**: User enters command
- Typee divides engagements into 3 types; *Appointments*, *Meetings* and *Interviews*.
- Typee application interacts with the users by separating application features into different menu windows. Each menu name is displayed in the left-end of the application window, where user can switch to different windows anytime by using command.
- Typee is a CLI (Command Line Interface) based application, with a mixture of simple GUI interacting features.

Summary of contributions

- **Role**: Software Engineer [UI Design, Implementation]
- **Major enhancement**:
  1. added **the ability to switch windows to other menu windows**
     - What it does: allows the user to switch the main window to load different menu windows to utilise different features any time.
     - Justification: In terms of user interface, by placing different feature interactions and forms in separate windows makes the application look more organised.
     - Highlights: New class `Tab` was implemented in the model and existing class `CommandResult` had to check if the command if the command type is `tab` command. This was a necessary implementation as the `MainWindow`, which is the root UI component of the main UI had to load external fxml file to the empty component during the command process. Structure of the MainWindow had to be modified appropriately so that each separate fxml files are linked with different JAVAFX controller classes other than `MainWindow` controller class.
     - Credits: SEDU team, where much of the architecture of the Main Window UI structure was referenced in order to customise our UI architecture to allow performing tab switching command.
  2. added **the ability to generate reports into a document form (pdf) based on the selected engagement**.
     - What it does: allows users to generate a report into pdf format document, where user can simply select the engagement that the user wants to generate into document and attach it to email to the corresponding receiver.
     - Justification: Secretaries and receptionists are not responsible only in task management. They have other tasks that consume alot of their time during office work. In order to improve their task performance more time efficiently, the feature allows users to save time instead of manually typing the document while referring to the engagement information one by one. With a simple 1 line command, typee generates a document, so that the user only need to attach the document their email. Document template is fully customisable, hence it makes the application flexible so that any corporations can use our application to fit their company document conventions formats.
     - Credits:
       - Commons-IO : file extension validation method is used for
       - iTextPdf
- **Minor enhancement**: Updated the existing AB3 CSS dark theme file to customise our Typee UI color schemes; TreeListView, Functional button skins (Delete button, Refresh button, Next/Previous month button) (Pull requests #223, #65)
- **Code contributed**: [Functional code]
- **Other contributions**:
  - Enhancements to existing features:
    - Wrote additional tests for existing features to increase coverage by roughly 9% (Pull

        requests #201, #200, #226)
    - Tools:
        - Integrated a third party library (iTextPdf, common-IO) to the project

Contributions to the User Guide

Given below are sections I contributed to the User Guide. They showcase my ability to write documentation targeting end-users.

Switch to different windows: `tab`

Switches to a different menu in the application window.
Format: `tab b/MENU_NAME`

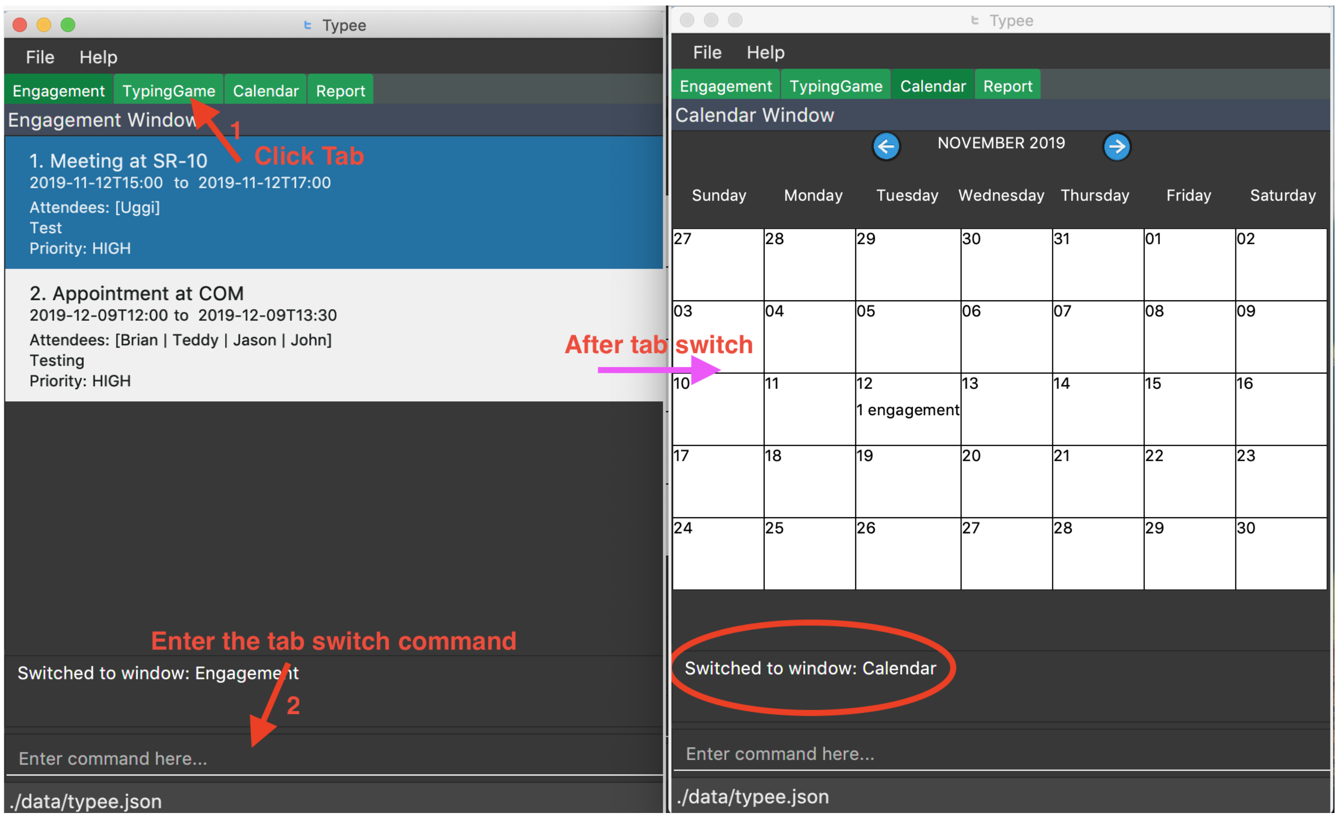
- Typee has 4 major features/menus. They comprise of:
    - Engagements List
    - Typing Game
    - Calendar View
    - Report Generator Users can switch between respective windows in order to make full use of the application.

NOTE `tab b/game` is used instead of `tab b/typinggame` to simplify typing.
`Engagement` window is set to default window upon starting the application.

If the user wants to switch to different windows, simply enter the tab command with refer to the menu name listed on the left end of the application window.

For example, user enters `tab b/calendar` , which the system will switch to calendar view window with a result display message set to "Switched to window: Calendar". Refer to the screenshot below.

NOTE Users who prefer using mouse over typing can alternatively click the green tabs on the top to switch to different menu windows.



```
1. Meeting at SR-10
2019-11-12T15:00  to  2019-11-12T17:00
Attendees: [Uggi]
Test
Priority: HIGH

2. Appointment at COM
2019-12-09T12:00 to  2019-12-09T13:30
Attendees: [Brian | Teddy | Jason | John]
Testing
Priority: HIGH
```

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
27	28	29	30	31	01	02
03	04	05	06	07	08	09
10	11	12 1 engagement	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

```
Switched to window: Engagement
```

```
Switched to window: Calendar
```

Generating a PDF file of engagement : `pdf`

Usage

Pdf Command allows user to create a document of selected engagement in a given format of document template. Document template can be customised based on the customers’s requirements, however, default document format will be in an email format, where the user can set who the user is going to send this document to inform an engagement.

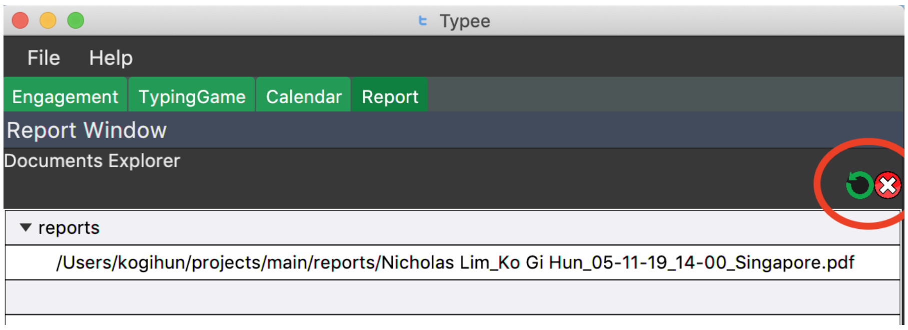
Format: `pdf i/LIST_INDEX t/RECEIVER f/FROM`

For example, if the user wants to create a document of an engagement, which has a list index of 1, which can by observed in `engagement` window. User sets the sender as `John`, which is the user’s name, and sets `receiver` as `Harry`. Hence, user enters `pdf i/1 t/Harry f/John` to generate the document.

NOTE	User can generate multiple report documents with the same engagement, but with different SENDER and RECEIVER names. System will not allow generating documents that already exists in the directory.

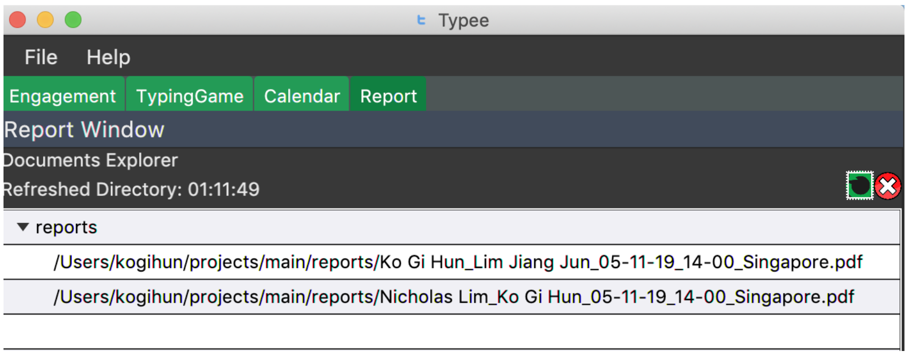
Once user enters the command, system will automatically open the generated document and display the command result in the output panel, showing "Engagement Report successfully generated."

NOTE	User can alternatively click the green refresh button on top right of the documents explorer to refresh the documents directory.



/Users/kogihun/projects/main/reports/Nicholas Lim_Ko Gi Hun_05-11-19_14-00_Singapore.pdf

Figure 2. refresh button and delete button.



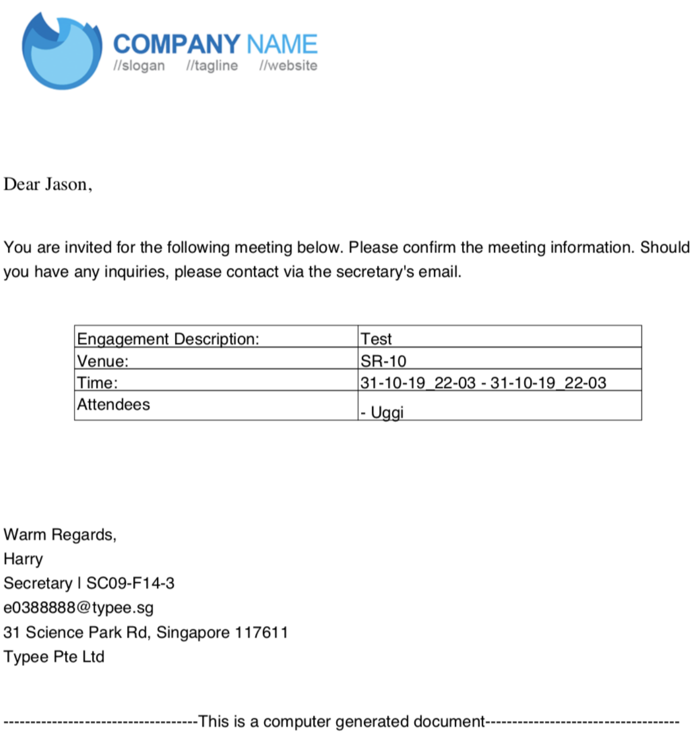
Refreshed Directory: 01:11:49

/Users/kogihun/projects/main/reports/Ko Gi Hun_Lim Jiang Jun_05-11-19_14-00_Singapore.pdf

/Users/kogihun/projects/main/reports/Nicholas Lim_Ko Gi Hun_05-11-19_14-00_Singapore.pdf

Figure 3. After clicking refresh button

NOTE Do not manually modify the document file name in the `reports/` directory as it might cause system failure in recognising the documents.



COMPANY NAME
//slogan //tagline //website

Dear Jason,

You are invited for the following meeting below. Please confirm the meeting information. Should you have any inquiries, please contact via the secretary's email.

Engagement Description:	Test
Venue:	SR-10
Time:	31-10-19_22-03 - 31-10-19_22-03
Attendees	- Uggi

Warm Regards,
Harry
Secretary I SC09-F14-3
e0388888@typee.sg
31 Science Park Rd, Singapore 117611
Typee Pte Ltd

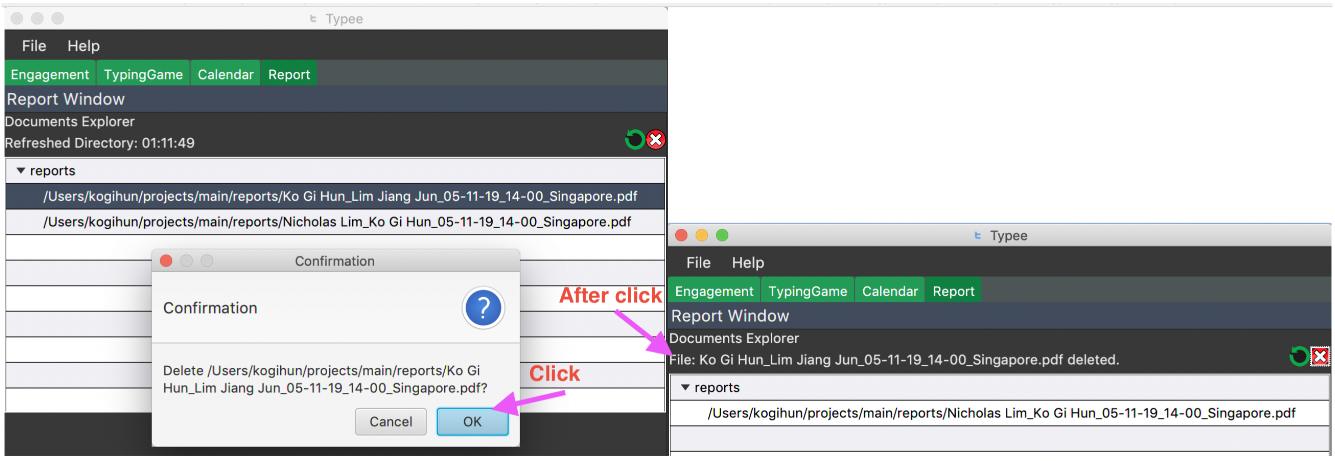
------------------------------------This is a computer generated document------------------------------------

Figure 4. pdf document sample

User can also double click the list item in the documents explorer tree view to open the document file on their local computer file system. Below is the sample of generated document from the system.

Deleting Documents
Instead of directing the actual directory in the local system, user can simply click the red `x` button, next to the refresh button to delete the selected document list item. Once system displays the popup message to confirm the user's decision, user will click the `OK` button to confirm deletion.

Delete function will only available when user has pre-selected the document list item and the selected item must be a .pdf form, not directory. Below is the screenshot after user clicks the delete button.

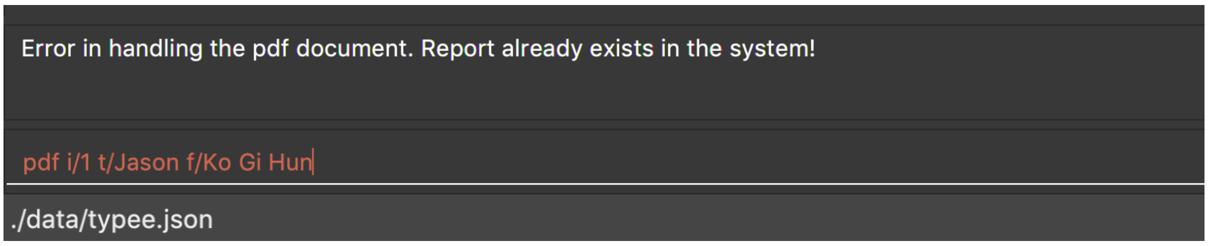

Typee
File Help
Engagement TypingGame Calendar Report
Report Window
Documents Explorer
Refreshed Directory: 01:11:49
reports
/Users/kogihun/projects/main/reports/Ko Gi Hun_Lim Jiang Jun_05-11-19_14-00_Singapore.pdf
/Users/kogihun/projects/main/reports/Nicholas Lim_Ko Gi Hun_05-11-19_14-00_Singapore.pdf
Confirmation
Confirmation
Delete /Users/kogihun/projects/main/reports/Ko Gi Hun_Lim Jiang Jun_05-11-19_14-00_Singapore.pdf?
Cancel
OK
After click
Click
Typee
File Help
Engagement TypingGame Calendar Report
Report Window
Documents Explorer
File: Ko Gi Hun_Lim Jiang Jun_05-11-19_14-00_Singapore.pdf deleted.
reports
/Users/kogihun/projects/main/reports/Nicholas Lim_Ko Gi Hun_05-11-19_14-00_Singapore.pdf
Figure 5. pdf delete pop up message and after click

Now, once system successfully deleted the selected document, system will display the status message above the documents explorer. Below is the screenshot of the system status message after deletion.

NOTE PDF generation and deletion operations cannot be undone via `undo` command.

Error Handling

1. Documents Explorer in Report Window only displays files with .pdf format in reports/ directory. Files with different format will not be displayed in the explorer.

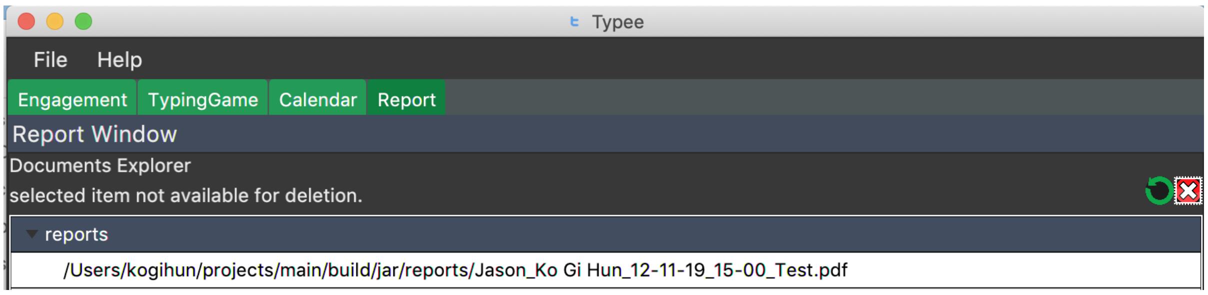

Error in handling the pdf document. Report already exists in the system!

pdf i/1 t/Jason f/Ko Gi Hun

./data/typee.json

Figure 6. duplicate file exception message displayed in status text field.

2. System will not allow user to delete directories in the documents explorer tree view.
3. System will throw an exception message in the status field if user locks the document.



selected item not available for deletion.

reports

/Users/kogihun/projects/main/build/jar/reports/Jason_Ko Gi Hun_12-11-19_15-00_Test.pdf

Figure 7. file deletion exception message displayed in the status text field.

Contributions to the Developer Guide

Given below are sections I contributed to the Developer Guide. They showcase my ability to write technical documentation and the technical depth of my contributions to the project.

Tab Switch Feature

Tab switch feature is a type of Command that allows users to switch to respective windows for using different features in the system.
Tab switch function is implemented in using both CLI and GUI approach; user can execute tab-switch by typing command in the input text-area or by interaction with the UI component (Button).

Implementation Structure

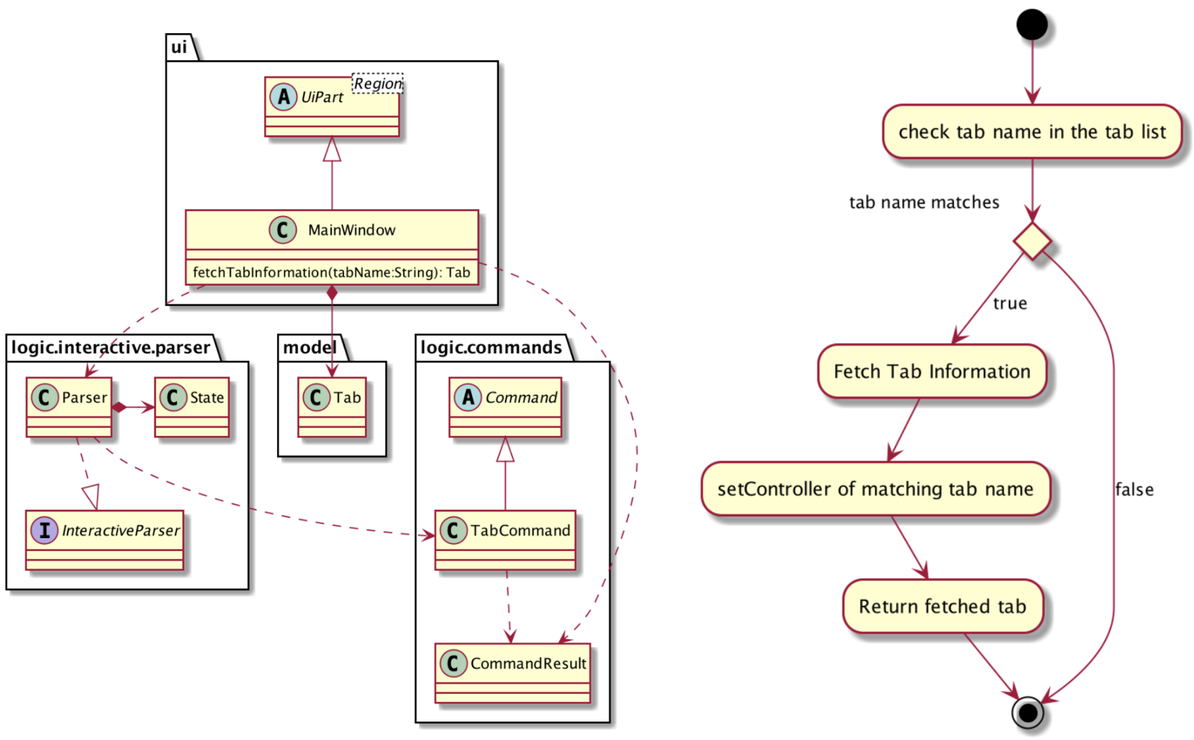


Figure 8. TabCommand Class Diagram and fetchTabInfo Activity Diagram

1. User has to use a specific `Command`; `TabCommand` in order to switch the window to another window.
    - Upon first startup of the system, by system default, system will display the `EngagementList` window.
2. New Ui Model class `Tab` is implemented to contain the respective fxml controller classes in a OOP manner.

MainWindow will have an additional method `fetchTabInformation(tabName)`. After the parser executes the `TabCommand`, it will return a CommandResult with `Tab` property.
Next, system checks whether the input name matches with one of the tab names in the system. If there is a match, system will fetch the tab information and return the `Tab` object. If no match is found, system will return withn an empty tab with only name attached.

Engagement Report Generation

This feature allows user to generate a pre-selected engagement in to a report and save it as a document file. The document file will be created in a .pdf format.

Implementation Structure

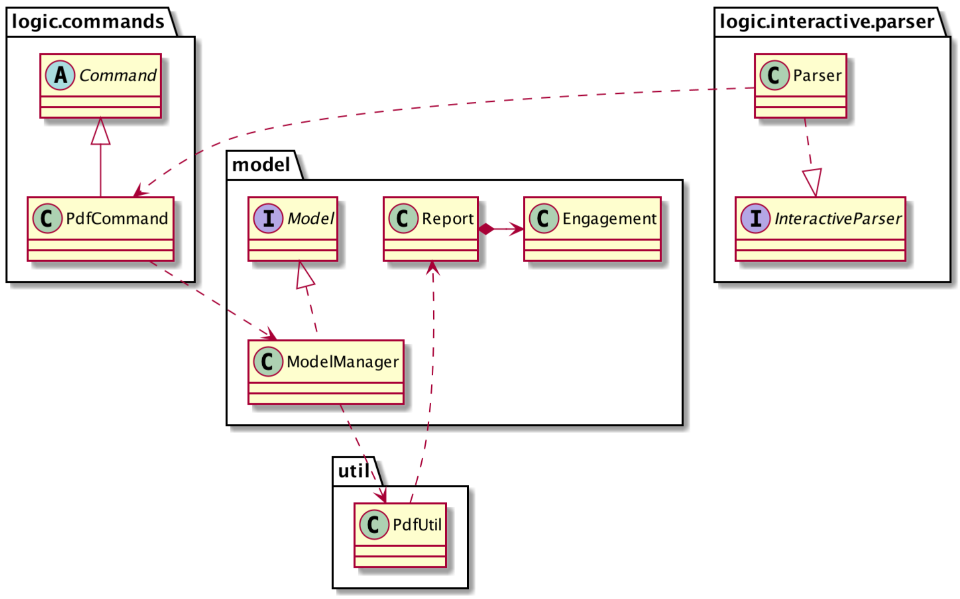


Figure 9. PdfCommand Implementation Class Diagram

The feature will be implemented as an additional type of Command; `PdfCommand`

- Proposed syntax of the PdfCommand is as follow: `pdf i/ENGAGEMENT_LIST_INDEX t/RECEIVER f/SENDER`.

Util class `PdfUtil` will be implemented for handling all pdf document creation related methods. It will be implemented under `util` package and it will be able to deliver few essential features that are necessary for document creation.

- Able to generate a full report document based on the `Engagement` as input.
- Use different templates for each other types of `Engagement` such as `Appointment`, `Interview` and

`Meeting`. Document template will follow general email format: Receiver, Engagement Information written in a table, Sender and signature with address and company logo.

To fulfill the document format, `Report` class needs to be implemented in order to model all necessary properties that has to be in the document. It 3 following properties;

- `engagement`: specific engagement information to include in the document
- `to`: A `Person` who is receiving the document
- `from`: A `Person` who is either a receptionist or a secretary who is sending the document.

Below is the sequence diagram for document generation which displays the concept of how this feature will be implemented by multiple interaction between different architecture components.

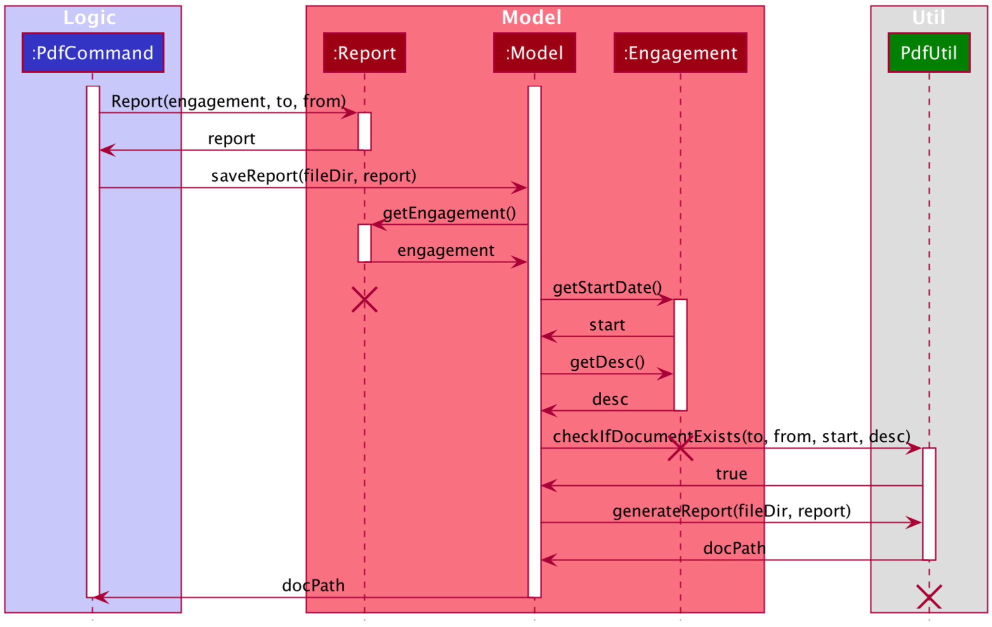


Figure 10. Sequence diagram for document generation.

UI Design

`ReportWindow` will be the UI container which helps the user to interact using `PdfCommand`. The window will include dynamic tree-view that is directly linked to the external directory in the file-system; `reports/`

UI Components & Features

Table below explains the components that are included in the `ReportWindow` with its purpose and features.

Table 1. Report Window UI Components

	UI Component	Feature	Purpose
`Refresh Button`	Button	Refreshes the tree-view file directory.	To help user to refresh the directory when there is a system glitch or manually adds document (.pdf) in the `reports/` directory.
`Delete Button`	Button	Deletes the selected document item in the tree-view file directory.	To assist users who prefer using mouse (GUI over CLI) deletes document easily.
`File_Explorer`	Scrollable Tree View	1. Displays the list of document generated previously and stored under the directory `reports/`. 2. Each list item is clickable with a MouseClick action of opening the document.	To allow user to manage documents more time efficiently.

Use case: (UC15) Generate appointment document in PDF format

MSS

1. User enters information for generating PDF of an engagement for a selected engagement.
2. System generates a PDF file in a specific external directory
3. System shows successful message
4. System opens the generated document.

   Use case ends.

Extensions

2a. System receives invalid input.

2a1. System shows error message.

Use case resumes at step 1.

2b. User inputs duplicate information that already exists in the directory.

2b1. System shows error message.

Use case resumes at step 1.