

Documentations



May 12, 2017

The university of nottingham

Contents

[Introduction 3](#_Toc482197958)

[The challenges 3](#_Toc482197959)

[Contributions 4](#_Toc482197960)

[Requirements 6](#_Toc482197961)

[Types of people using the system 6](#_Toc482197962)

[Main Requirements 6](#_Toc482197963)

[How “Travel Dashboard” should work? 6](#_Toc482197964)

[Project Management Approach 7](#_Toc482197965)

[User Case Study 8](#_Toc482197966)

[Risk Analysis 10](#_Toc482197967)

[Beginning stage of the project 10](#_Toc482197968)

[Development stage of the project 10](#_Toc482197969)

[Maintenance stage of the project 11](#_Toc482197970)

[User Guide 13](#_Toc482197971)

[Log in Screen 13](#_Toc482197972)

[Registration 14](#_Toc482197973)

[Registration Code 15](#_Toc482197974)

[Main Dashboard Page 16](#_Toc482197975)

[Upload Dashboard Page 20](#_Toc482197976)

[Uploading File 21](#_Toc482197977)

[Removing Records 23](#_Toc482197978)

[Report Dashboard Page 24](#_Toc482197979)

[Generating Report 29](#_Toc482197980)

[APPENDIX 30](#_Toc482197981)

[Spreadsheet Data Format 30](#_Toc482197982)

[Installation Guide 33](#_Toc482197983)

[How to Import in Java 34](#_Toc482197984)

[How to Produce .JAR File 37](#_Toc482197985)

[Producing .exe File 42](#_Toc482197986)

[Helpful Sources- 52](#_Toc482197987)

[Database 53](#_Toc482197988)

[Software Used 53](#_Toc482197989)

[Database Diagram 53](#_Toc482197990)

[User Table 55](#_Toc482197991)

[Excel File Table 56](#_Toc482197992)

[Queries from the Table 60](#_Toc482197993)

[Budget Table 62](#_Toc482197994)

[Queries from the Table 62](#_Toc482197995)

[Bibliography- 65](#_Toc482197996)

# Introduction

Atos is one of the leading global companies that provides services to millions of their customers. Their teams from all over the world collaborate with each other to maintain their reputation. This means that the employees need to travel quite often from one branch to another. This, however, has brought some issues for high level managers who manages teams from different branches. Although, Atos has a very sophisticated system that allows employees to claim expenses for their travel, however, the system produces a huge report in excel file format that manually needs to be analysed. This makes it very challenging for the higher managers who want to see their team's travel expenses to prevent them from going beyond their budget. Atos has reached out to us to solve this issue by helping them building a “Travel Dashboard” system. Atos have requested to build a system that can help them upload their yearly expense reports in the system. They also want a system that can produce charts that categorises the expenses in different sections such as- how much an individual spent or how much they spent on an element, for example- the hotel bills, meals etc. So, when Atos came to us, our team has promised them to not only constitute a system that shows charts but also helps them to analyse their expenses in more details. Our team has finally delivered a system that combines both simple, user friendly design along with sophisticated algorithms that calculates and categorises their expenses effectively. The “Travel Dashboard” system can take one or multiple excel files and calculates the total cost that can be found in those excel files. The higher managers can also search how much an individual has spent in a team. The system is very reliable and secured because it does not allow everyone to use the system unless they are signed up in the system.

## The challenges

There are a lot of challenges that have been encountered while completing this project. However, these challenges have motivated us to make more precise decisions and make the system more reliable for our clients. These challenges include-

- **Communication barrier-** At the beginning of this project, there were some communication errors which lead the team to a lot of confusion, nevertheless the team had regular meetings with the clients and supervisors. However, after analysing our progress in our requirement stage we identified our mistakes. These mistakes include- not reviewing the requirements properly, not requesting our clients for useful resources/example files, not working as a team etc. After careful observation, we decided that we would have to change the way we communicated with our clients. Our main priority was to understand their requirements. To do that, we came up with some user stories and explained it to our clients to confirm the requirements. We also produced Gantt charts to predict the stages that we should go through once we started this project.

-  **Not working as a team-** When we started this project, we were not effective enough in terms of communicating with each other. We worked on the project, however, did not document our progress as much as we should. This diverted us from making significant progress at the beginning of this project. However, we managed to overcome these barriers by using our project planning skills. We organised meetings every week with our clients, team and our supervisor and made a comprehensive documentation of our meeting minutes. We also used trello where we listed our tasks each week and gave ourselves a deadline. Changing our job role in the team also helped us raise the standard of our work.

- **Problems with the files-** At the beginning of this project, we did not receive enough resources from our clients. This prevented us from making any progress with our system. The reason for this was because we needed to design our database that would fit our client's requirements. However, our clients needed to anonymous their data and we could not make a start without these resources. However, we figured out that we could ask them some questions regarding the schema that we might expect from their data. This helped us to design our database with the most important fields that could be present in the excel file that would be uploaded in our system.

## Contributions

**Anthony-**

* + Designing the signup, login, main page, upload page and report page
  + Programming the signup, login, main page, upload page and report page
  + Designing and programming the PDF report
  + Allocating tasks for the other team members
  + User stories
  + Documenting the requirements
  + White box testing

**Jaber-**

* + Documenting the requirements
  + User stories for the clients
  + Meeting presentations for the clients
  + Dealing with the clients and organising meetings
  + Designing the Report page along with Anthony
  + Database documentations, user guide (Along with Lukas), installation guide, meeting minutes.
  + Allocating tasks for the team members (Along with Anthony)
  + Designing the team leader, project leader page.
  + Programming the team leader and project leader page.

**Lukas-**

* + Working on the FXML of the main page, upload page and report page
  + Designing the poster
  + User Guide
  + FXML Documentation

**Niall-**

* Testing
* Helping with database design

**James-**

* Attending the meetings

# Requirements

We have mainly gathered the requirements based on the project details that has been provided to us at the beginning of this module. However, the requirements have been analysed more in depth by having a face to face meeting with our external supervisors. These are the final requirements we have assembled based on our meeting

## Types of people using the system

There are three types of users who are most likely to use the software-

* Finance
* Managers
* Procurement Manager

These are the employees in the company who are most likely to use the software. The main reason for **the finance** to use this software is to plan the budget for each team in the company. **The managers** are going to use the software to find out the individuals who spent the most while travelling and how much they are spending every single time. **The procurement managers** are most likely to check the dashboard to find cheap deals for the employees, for example- if an employee goes to a specific location frequently then the procurement managers can manage a deal with the hotel nearby that place and get discount for the employee.

## Main Requirements

The company is mostly interested in looking at

* The cost/expenses of each staff member (in terms of travelling)
* The expense report needs to be approved by the managers before the employees get their money.
* The locations they have travelled the most (i.e. the hotels/ a specific branch) Mainly interested in representations of data (charts) in order to see where are they spending most?
* Generating all the charts in a pdf format

## How “Travel Dashboard” should work?

1. **Login System-** Login system should contain the username and the password of the users. This is to help differentiating the different users for the software.
2. **Setting Budget-** The allowance should show the managers how much employees spend and how much they are allowed to spend. This should be visible to both the managers and the employees.
3. **Notifications-** The status of the process (whether the report has been finished or submitted) should be displayed by the system.
4. **Uploading Reports-** The excel report needs to be submitted so that the managers can see the expenses.So, the system should contain a section that indicates whether the report has been submitted by the managers.
5. **Individual Report-** Making notes of the hotel stays and the expenses any other relevant details. So, there has to be a specific page that lists out all the expenses for an individual employee and also how much all the employees of a specific manager have spent on travelling.

## Project Management Approach

**Analyse**

**Plan**

**Deploy**

# User Case Study

This is a user story that has been formed based on the requirements that we have collected from Atos. This has been made to help us confirm the requirements with our clients.

The User Case Specification (based on the solution that the software has the claim functionality.)

**The term `team` is like a department, the term `project` has the same hierarchy with team, i.e. a team can have several projects, and a project can be cross-team.**

**Higher Manager (Finance Director and CFO/Approver BU Manager/Procurement Manager)**

**As a higher manager, I would like to:**

1. Be able to assign a team leader for each team.

2. Be able to add/remove the users to every team.

3. Be able to create/delete project, and allocate it to specific teams.

4. View every team’s cost summary through different graphs and charts.

5. View the summary of every project in various format, just as team summary.

6. Be able to see everyone’s expense which is categorised as time and the type (for example hotel cost). View can be through standard statistics such as showing of median, mode and max in various format of view. (Whether the member has spent more than the maximum/allowable expense.)

7. See the software generating a ‘PDF’ version file based on team/project summary.

The summary in higher manager’s dashboard would contain:

1. View the max/min/average cost of the total cost by all teams per week/month/quarter/year/all (label)

2. The total cost by all teams per w/m/q/y/a (line chart)

3. The total cost by all teams in receipts type per w/m/q/y/a (bar chart and pie chart)

4. Every team’s cost per week/month/quarter/year/all (line chart)

5. Every team’s cost in receipts type per w/m/q/y/a (bar chart and pie chart)

6. Every project’s cost per week/month/quarter/year/all (line chart)

7. Every project’s cost in receipts type per w/m/q/y/a (bar chart and pie chart)

8. Compare the team’s cost with the average cost (bar chart)

9. View the places(hotel) where the teams cost while travelling (table)

10. View the vehicles the teams cost while travelling (table)

**Team leader**

**As a team leader, I would like to:**

1. Be able to add/remove the users to every team (?)

2. View the team’s cost summary through different graphs and charts.

3. View the summary of projects that are allocated into the team in various format, just as team summary.

4. Be able to see his team member’s expense which is categorised as time and the type (for example hotel cost). View can be through standard statistics such as showing of meaden, mode and max in various format of view. (Whether the member has spent more than the maximum/allowable team expense.)

5. See the software generating a ‘PDF’ version file based on team/project summary.

The summary in team leader’s dashboard would contain:

1. View the max/min/average cost of the total cost by team per week/month/quarter/year/all (label)

2. The total cost by team per w/m/q/y/a (line chart)

3. The total cost by team in receipts type per w/m/q/y/a (bar chart and pie chart)

4. The total cost by project per w/m/q/y/a (line chart)

5. The total cost by project in receipts type per w/m/q/y/a (bar chart and pie chart)

6. Every team member’s cost per week/month/quarter/year/all (line chart)

7. Every team member’s cost in receipts type per w/m/q/y/a (bar chart and pie chart)

8. Compare the team member’s cost with the team’s average cost (bar chart)

9. View the places(hotel) where the team members cost while travelling (table)

10. View the vehicles the team members cost while travelling (table)

# Risk Analysis

This section of the report covers the risks we might face at different stages of the project. In this section, we have identified some of the common issues that we might encounter and how we are planning to solve these issues when they arise.

## Beginning stage of the project

**Misunderstanding the requirements** – Gathering requirements effectively should be the main priority for a team which is working on a new project. This is what makes a huge difference between a successful and an unsuccessful project. Understanding the requirements of the clients is important because the main purpose of each project is to satisfy the customers’ needs and solve unique problems. However, if the requirements are not gathered properly then it will be very difficult to produce a product that will meet the customers’ needs. The requirements can be misunderstood if the team members do not communicate with the clients regularly or ask them relevant questions regarding the project. To solve this issue, it is very important to make sure that our team has regular meeting with the clients. We have also planned to ask them relevant questions regarding the project so that we can analyse their requirements. We will be reviewing their requirements before every meeting until we receive and understand their requirements.

**Miscommunications between team members -** Miscommunication between the team members can slow down the progress of the whole team. If someone in the team does not understand their role in the project, then the project will result in failure.  This is because the part that will be specified for that specific member will be missing or will be lacking in a lot of sections as they will not have any idea of how to complete that section. In order to mitigate this issue, we have planned to have a team meeting every week after our meeting with our clients and our internal supervisor. We will review what each other think of the meeting every week so that we can justify each other’s understanding of the project and more importantly the parts which they have been assigned to do.

**Not understanding the whole project –** Not understanding the whole purpose of the project can prevent the team from executing the deliverables successfully. The team will suffer a lot in terms of managing the projects and when it comes to delivering the correct product. To prevent this issue, we will be documenting our clients’ requirements. We will also work on user stories and review this with our clients and supervisor to confirm their requirements. This will help us reflect on some of the most essential functionalities that we might have left out accidently.

## Development stage of the project

**The developer crisis –** The developer crisis can impact on the whole system as the product will not be built in time if our developer does not produce the system. This can cause either because of the misunderstanding of the requirements or because of the lack of communication between the team members. The team needs to collaborate to make it easier for everyone to contribute to this project. However, if the work is not distributed accordingly then it will be harder for the team members to complete the whole project successfully. This can also mean that some members in the team must put more effort than the others which can also have a massive impact on the quality of the final product. To prevent this issue, we will make sure that two of us are working in each section which means that if one member does not understand then the other team member can help them. Although, this might seem a bit time consuming, however, by splitting a part into two sections and combining them together can make every team member’s job easier, instead of putting too much pressure on one team member.

**Developing the wrong product –** It can be very easy to develop a wrong product if the team members are not careful. There are many factors that can have an impact on making a wrong product. These can include – not understanding the requirements of the clients, clients not explaining the requirements correctly, not completing the program in the correct sequence etc. To prevent this issue, we will review our prototype with our every time we change it. This will help us avoid adding/removing any essential features from the system.

**Server failure –** Server failure can be a great issue as the software needs connection to the server to connect to the database. This can cause a great issue since we will be using database for our system. To prevent this issue, we will be using a cloud based server along with the school server so that we can always retrieve the data even when we can’t access the school server.

**Losing the work –** It is very common to lose work especially when the system is stored in one’s computer. It is also very easy to mislead other team members if one does not share their progress with the rest of the team. To prevent this issue, we will be using GitHub where we will share what each of us will be improving in the system. This will also help us retrieving our system if we lose it from our computer accidently.

## Maintenance stage of the project

**Not providing clear documentations –** The documentations for this project must be structured in a way that will help the Atos technical team to maintain this project appropriately. However, if the documentations do not elaborate all the essential features and functionalities of the system then it will be very hard for Atos’ technical team to understand the system. Hence, it will be very challenging for them to maintain/improve the system in the future. To prevent this issue, we will request our clients and our supervisor to test our system and ask us questions regarding the system. This way we can determine what we need to include in our documentations. We will also divide the documentations in different sections such as user guide for users and installation guide for the technical team in order to cover both the graphical sections and also the coding section of our program.

**Not delivering the required product –** The required product needs to fulfil all the requirements of the clients. This will be very challenging if the requirements are not documented correctly. However, there are many situations where the product might fail to live up to the clients’ expectation even when the requirements are documented perfectly. This situation arises when the system is not tested against the requirements which are listed at the beginning of the project.  To prevent this issue, we will use the user stories as a check list and test the final product against the requirements that will be listed in the user stories.

# User Guide

Travel Dashboard is an application that allows users to analyse travel as provided by the Atos system for reimbursement of travel expenses for their staff.

Application will ultimately connect directly to the Atos system but that is not enabled with the version of the Travel Dashboard implemented and delivered in April 2017 by students at the School of Computer Science, University of Nottingham.

Travel Dashboard is designed to analyse and generate reports about imported data in the form of Excel file with travel data from the Atos system. The format of the data is fixed and import files must adhere to that format for analysis to be correctly performed. File formats are described in the Appendix.

This document provides information about the features and use of software. Travel Dashboard application does not require prior skills in statistical analysis or Excel spreadsheet operations.

In order to use the application, the user must have a login account. That is required, even though the application is standalone and not connected with the Atos system. Login must be obtained in advance from the Atos technical support team.

## Log in Screen

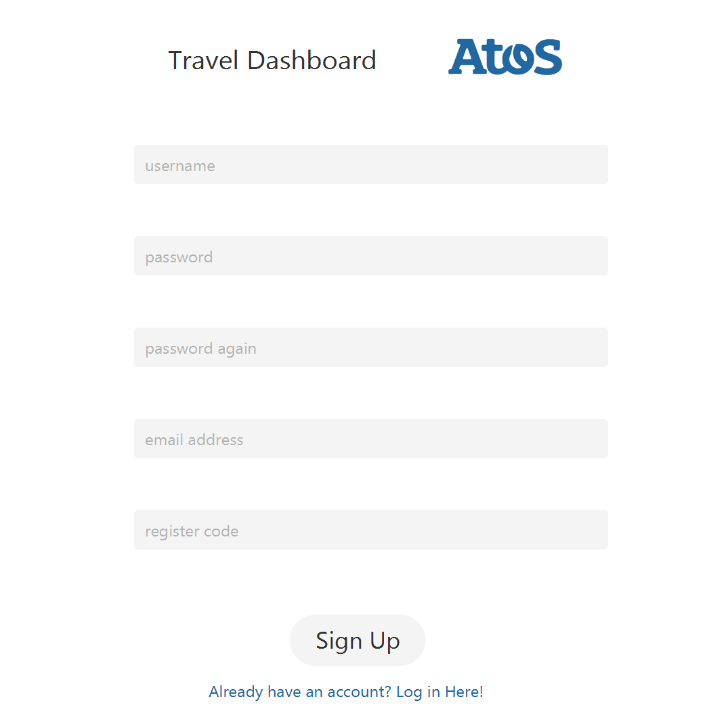
The log in screen is used by approved members to log into the Travel Dashboard application to generate, analyse travel data or make a claim for the travel expenses. The log in form of Travel Dashboard looks like this:

|  |  |
| --- | --- |
| This first box is for the member username that was provided by the company or made by the users. The second box is for the password that is linked with the username. The *“Log in”* makes a connection with the database. |  |
| If the username and/or password is incorrect or it is not formatted correctly, then a specific box will show up telling that the member has entered wrong details. | https://scontent.flhr3-2.fna.fbcdn.net/v/t34.0-12/18191636_1486202584801008_1552790960_n.png?oh=1e3fe56b6cc9d129e517b4be26ec53ac&oe=59054E69 |

## Registration

The registration screen appears when the member presses text *“Not yet have an account? Sign up here”* located in the log in screen. The form contains four boxes – username, password, repeat password and registration code. See the image below:

New application member must enter his/her username into the first box it will be the same username for the log in form. Unique ID will be created and linked with this username.



Password must be at least 6 characters long and it is case sensitive. (Use of both characters and numbers is recommended.)

Repeat the same password.

Enter the company email address.

To enter the registration code, contact the technical team and they will provide a unique code that can be entered to successfully sign up.

## Registration Code

Registration code is a special/encrypted code that will be mainly assigned by the technical team. This code is used to prevent unauthorized users to access the system by matching the code (that you put in the registration code box) with the database. Therefore, it is important to acquire this code from the technical team in order to sign up to the system successfully.

The sign-up button initializes the creation of the account and adds the details to the database.

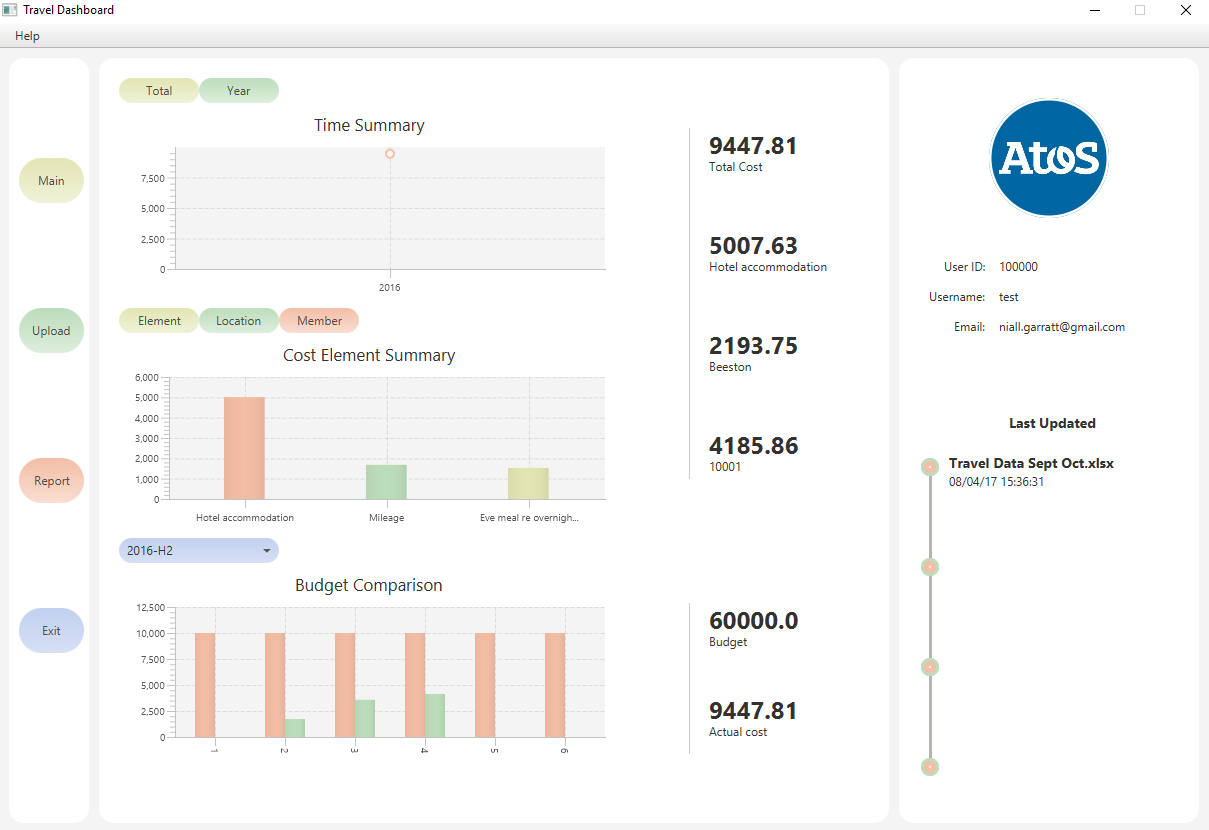
|  |  |
| --- | --- |
| If every detail in the form was filled-up correctly, then a box will pop-up saying that the *“Account created successfully!”*. See the image on the left. |  |

From now on the member can log in into Travel Dashboard using the log in form.

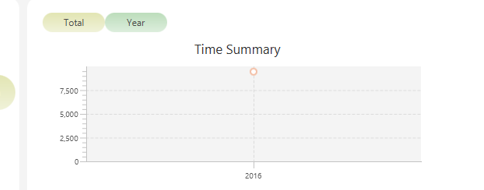
|  |  |
| --- | --- |
| If the password and the repeated password does not match, then a box will show up saying that *“Password not matched”*. To solve this issue, the member must retype password in both inputs and make sure that they match. |  |

## Main Dashboard Page

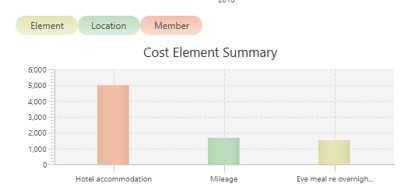
The main page of the Travel Dashboard contains three charts representing statistics of the data that was uploaded into the system and analysed.



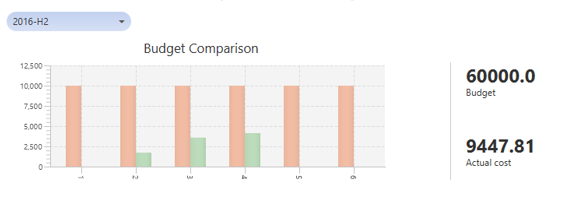
The first chart represents the *“Time Summary”*. There is an additional filter above the chart to choose whether to represent “Total” or “Year” summary.

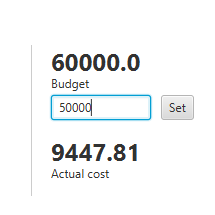


The second chart can be chosen from three different options – *“Element”, “Location”* or *“Member”*. *“Element”* – the chart will show cost element summary. *“Location”* – the summary of location data. *“Member”* – summary of team member.

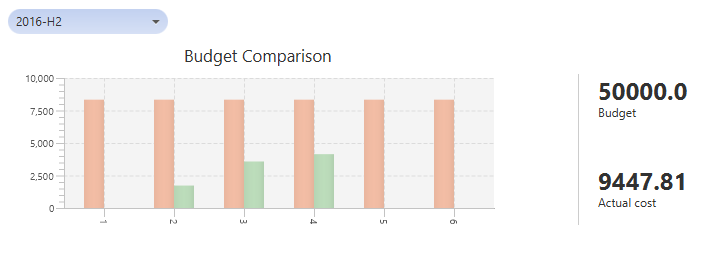


The third chart represents the *“Budget”* and *“Actual cost.”* The users can choose a specific budget by clicking on the Number on top of the *“Budget (highlighted by the rectangle shape)”* text.

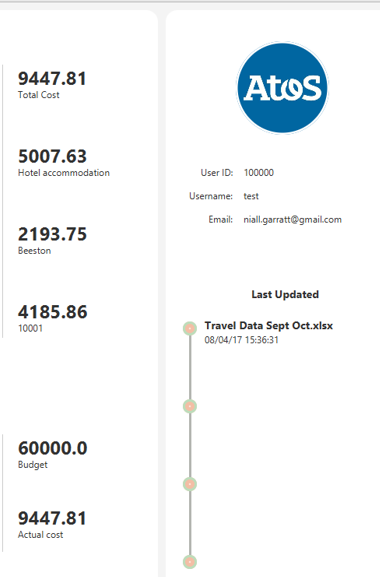




This is how the users can change the budget. They can simply insert the number and then click set.



As it can be seen from the image above, the chart changes based on the budget set by the user. This chart shows the two halves of the year. This is the second half of the year as shown by the ***lookup menu (shown using oval.)*** The budget is divided by six and it compares the actual cost per month with the budget per month (that is calculated based on the number set by the user.)



Last Activity of the user

v

v

v

The employee who spent the most

The Overall cost

Most Expensive Element

Location where most of the money were spent

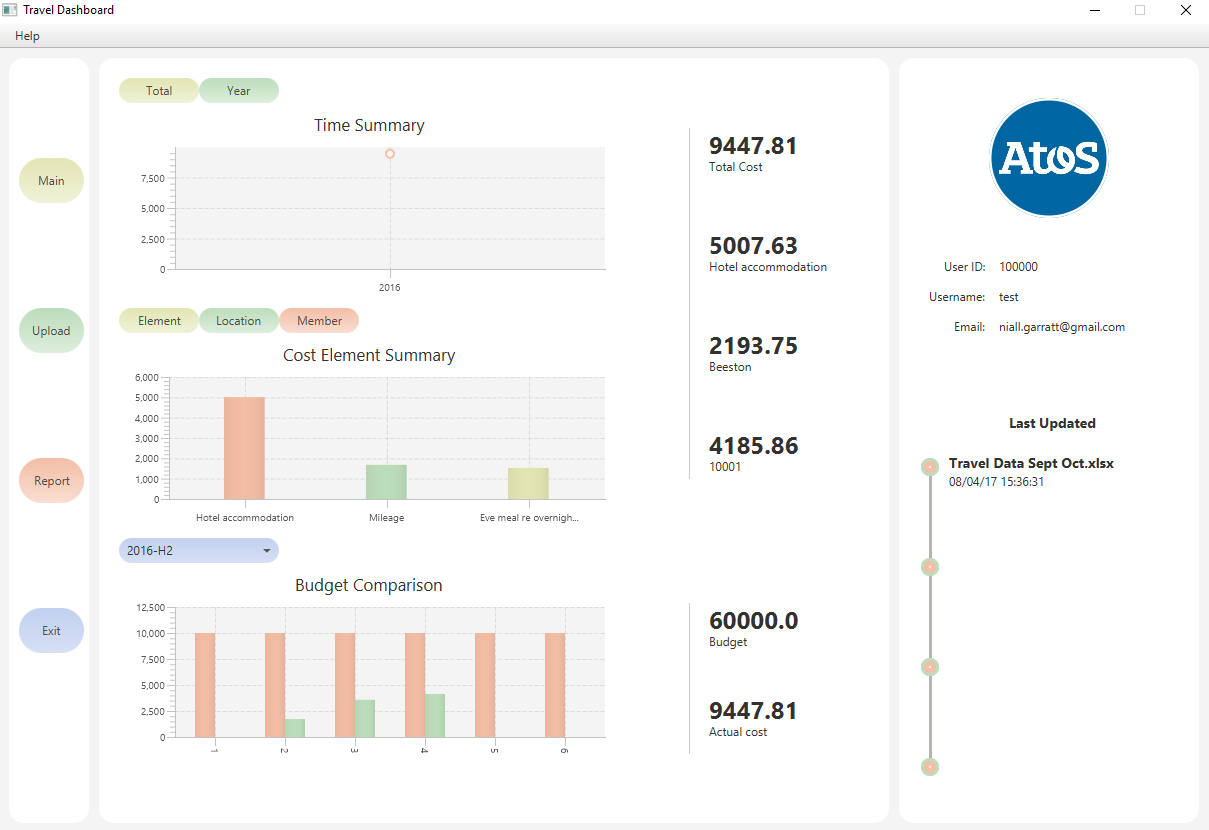
v

v

User Details

v

## Upload Dashboard Page



To upload an excel file click this option.

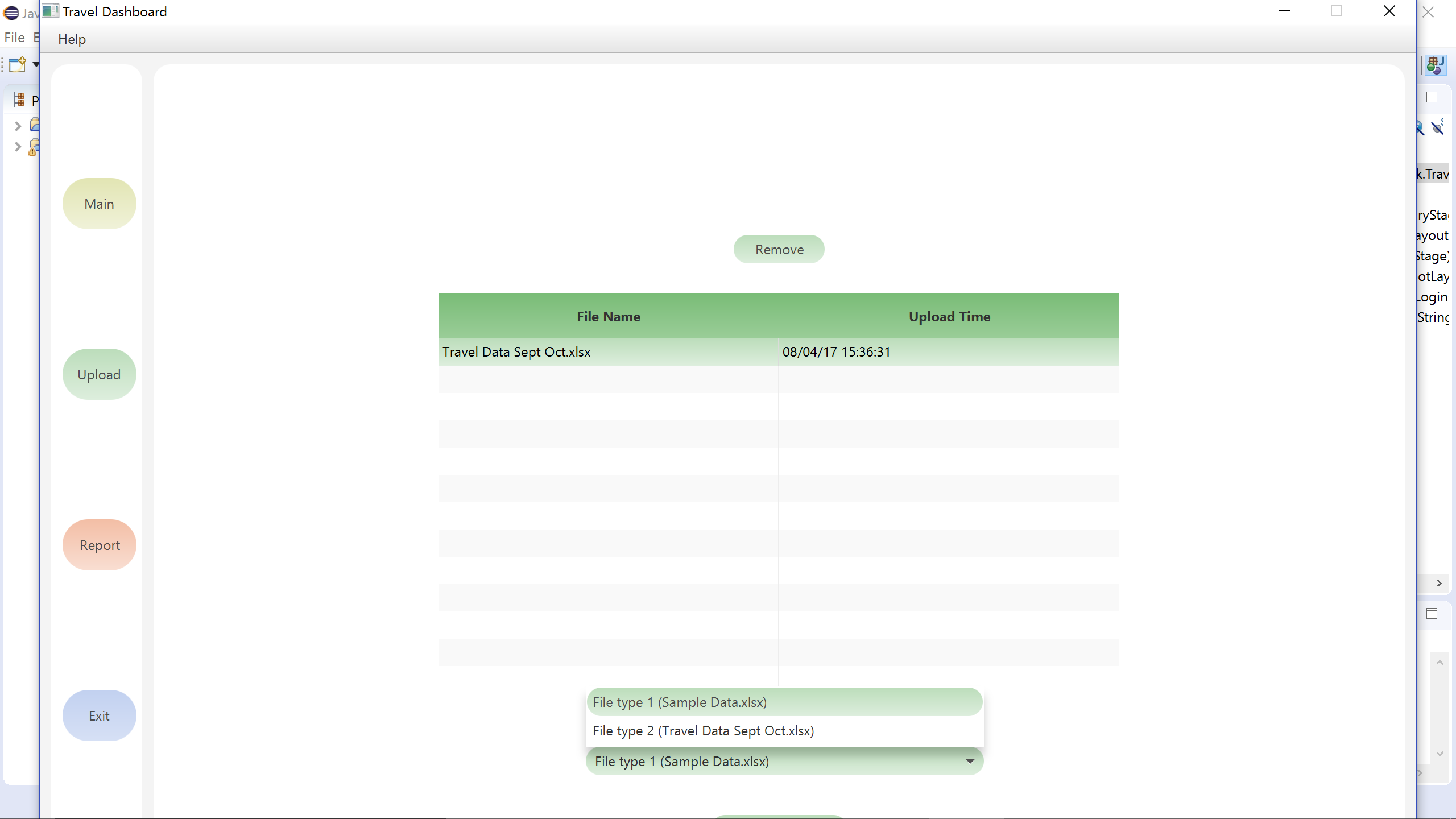
The upload page is used to upload travel data into the application. A report will be generated based on the data that was uploaded into the application. There is a table *“Upload History”* in the center of the page showing file names and date when they were uploaded in the application. The page also contains *“Remove”* and *“Upload Excel”* buttons as well as dropdown for choosing the *“File type”*. See the image below.



**NOTE- The remove button only appears when a file is chosen/clicked. The “Upload Excel” file only appears when the file type is chosen from the lookup menu.**

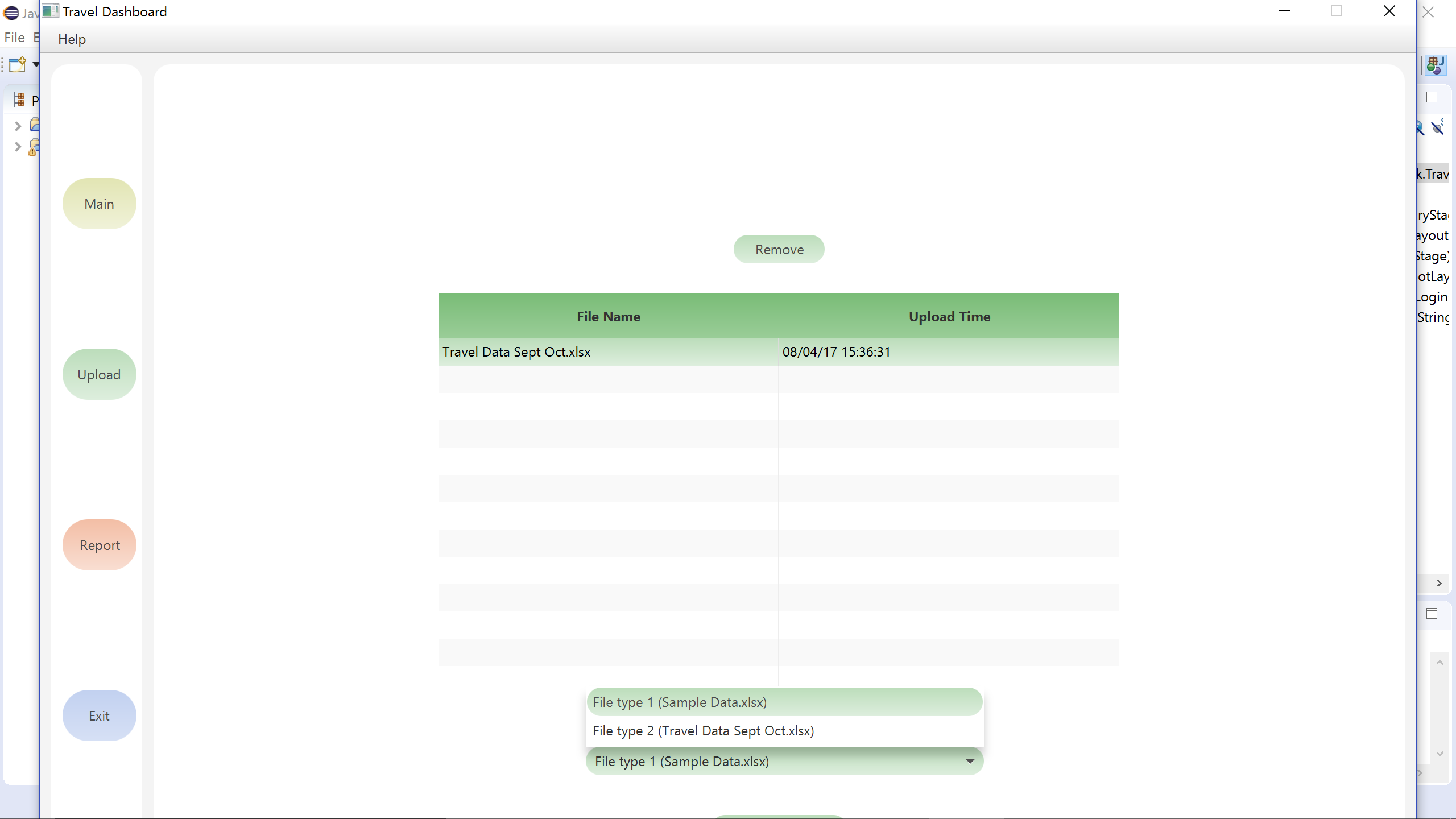
### Uploading File

To upload a file into the application a member must choose the type of the file from a dropdown and press *“Upload Excel”*.



|  |  |
| --- | --- |
| These are the two types of schema the system accepts. |  |
|  |  |
| A window will open where a member should choose which file does he want to upload into the application. |  |
| Before finishing the upload into the application there is option to review the information that is going to be uploaded into the system. If the information is correct, then a member should press *“Confirm”* button and the information will be uploaded. |  |
|  |  |

### Removing Records

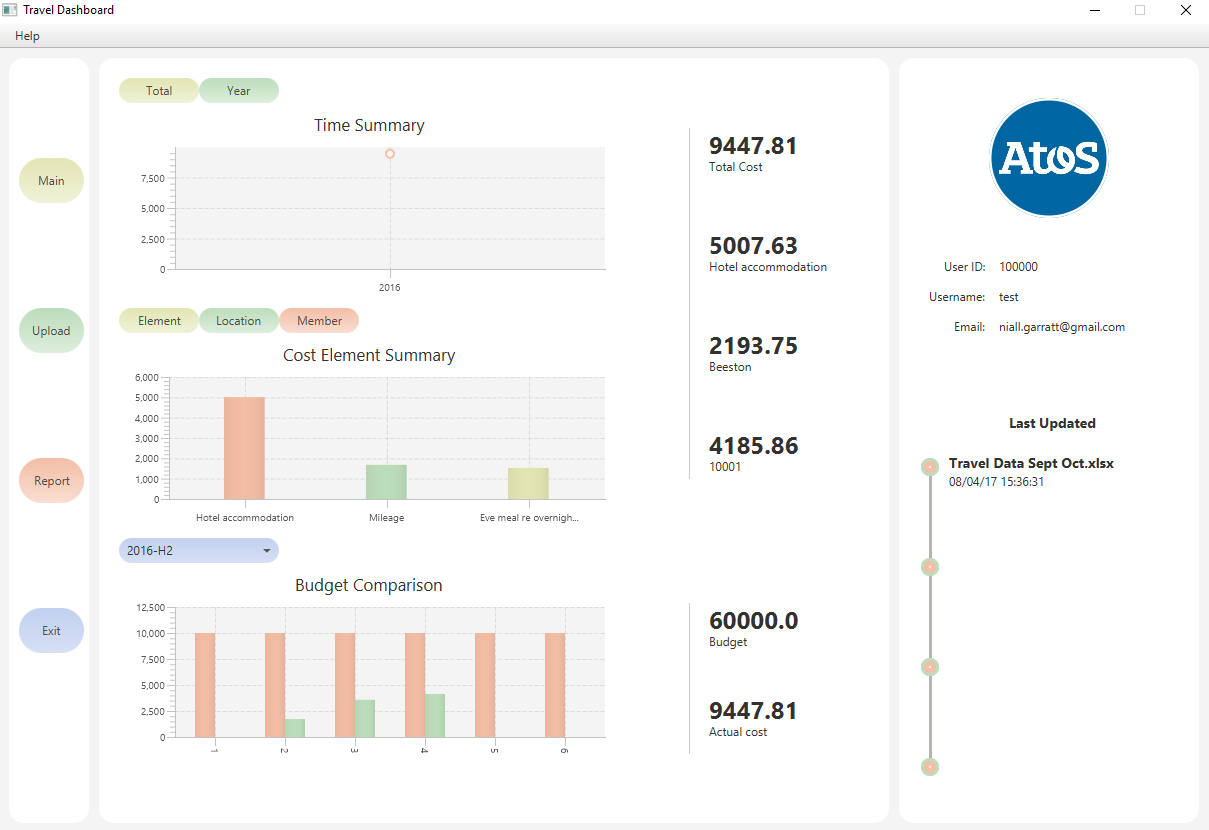


If a record in the upload page is incorrect or the information is no longer relevant, then it can be deleted from the system.

To delete a record, choose the file name that needs to be deleted and press the *“Remove”* button. See the image.

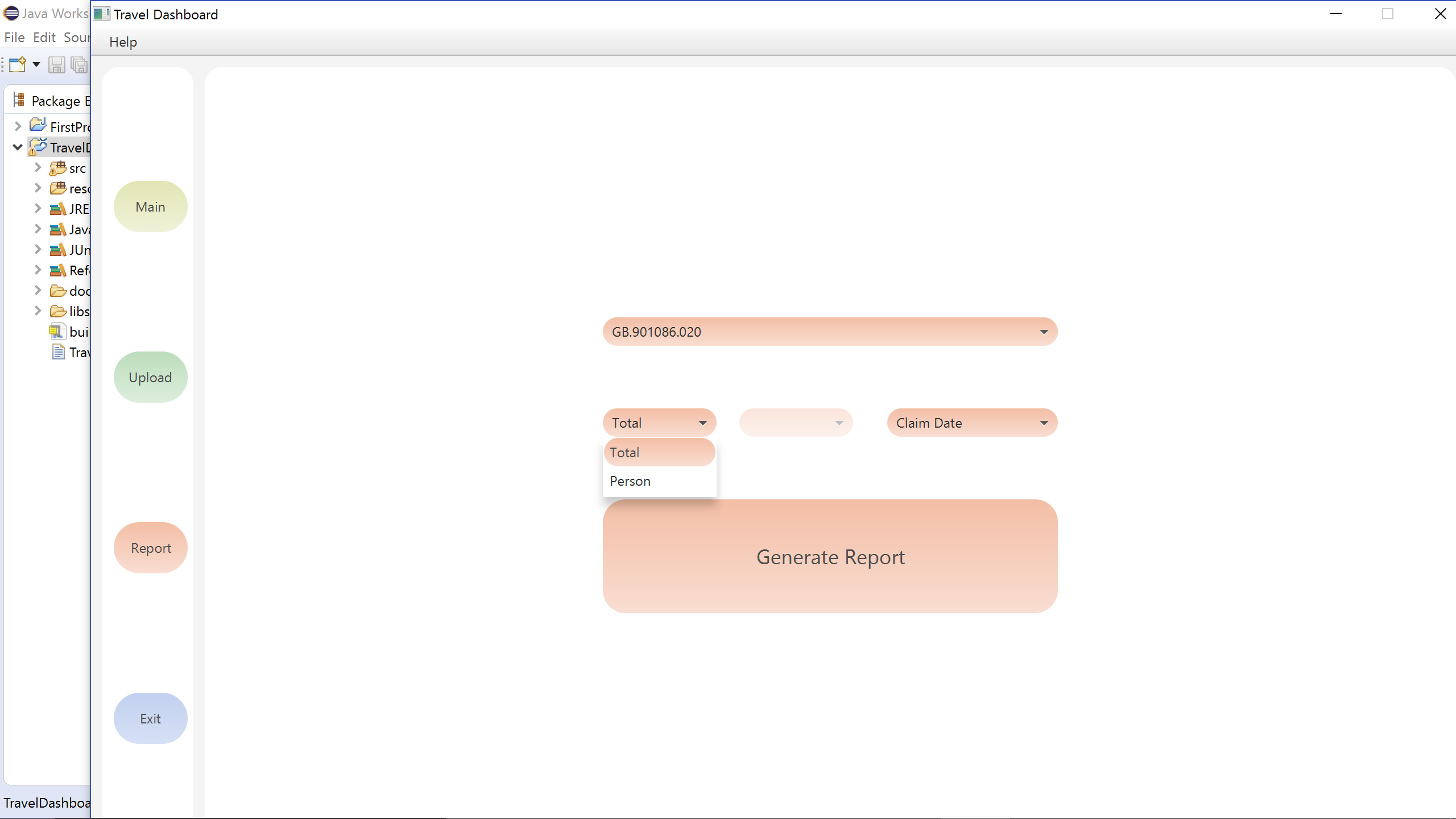
|  |  |
| --- | --- |
| A box will show up saying that *“You are trying to remove the record”*, if the member presses *“OK”* then the record will be removed from the system. |  |

## Report Dashboard Page



To generate report after importing a file, click this option.

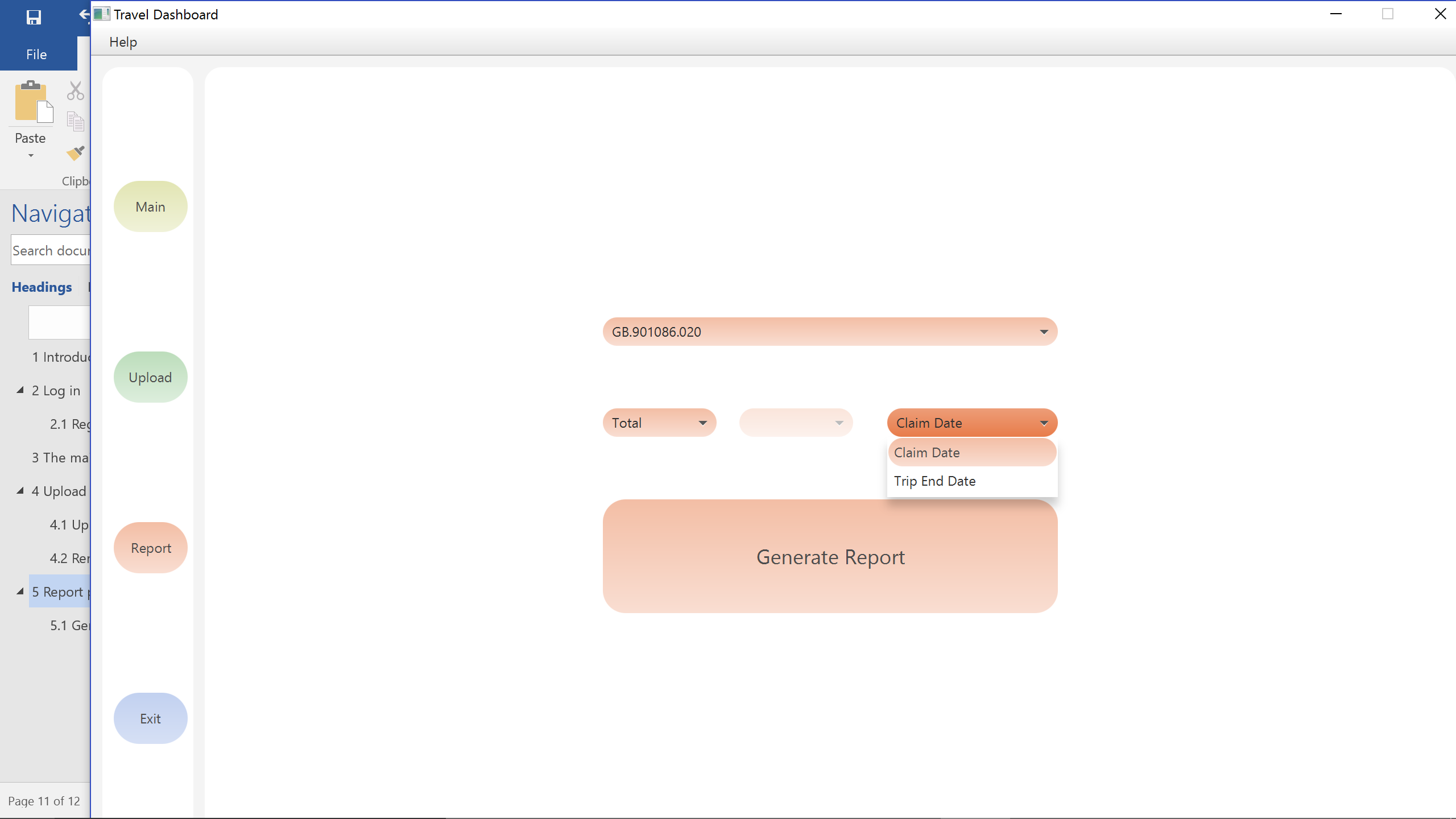
The report page generates a report on a specific criterion that is chosen by the member. The report is made from the data that was uploaded in the upload page. The report page contains three dropdown boxes – *“Total/ Person”, “Claim Date/ Trip End Date”*, one for choosing specific value from the data and a button to *“Generate report”*.



Produces report for a specific employee.

Produces report for the whole file.

This is the WBS code that can be used to specifically identify a project and produce report for that project.



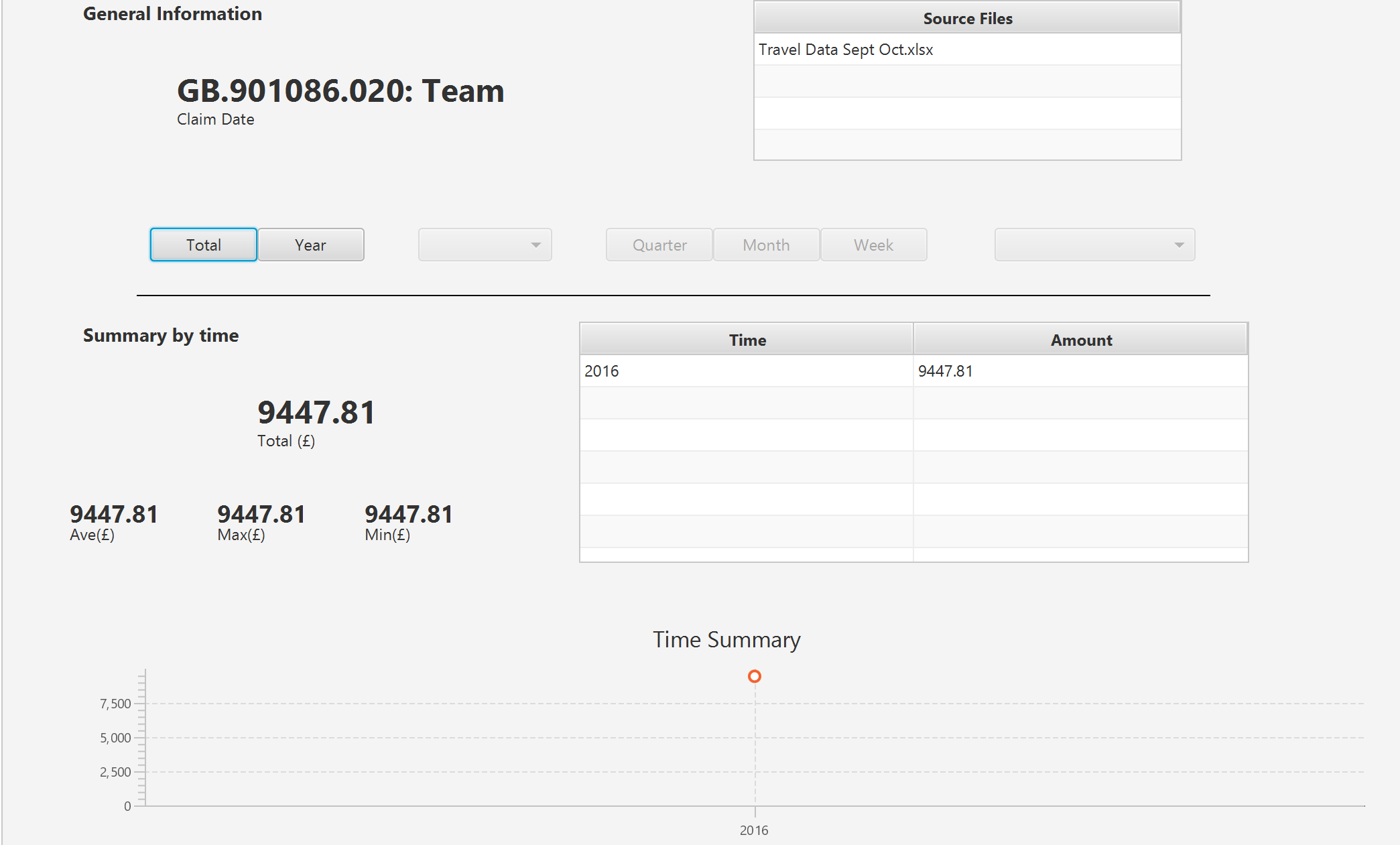
Produces report based on the day the trip end date.

Generates the report which is shown and discussed below.

Produces report based on the day the employee claimed the expenses.

The report contains several sections which are discussed below-

These are the files which are analysed and used to generate report.

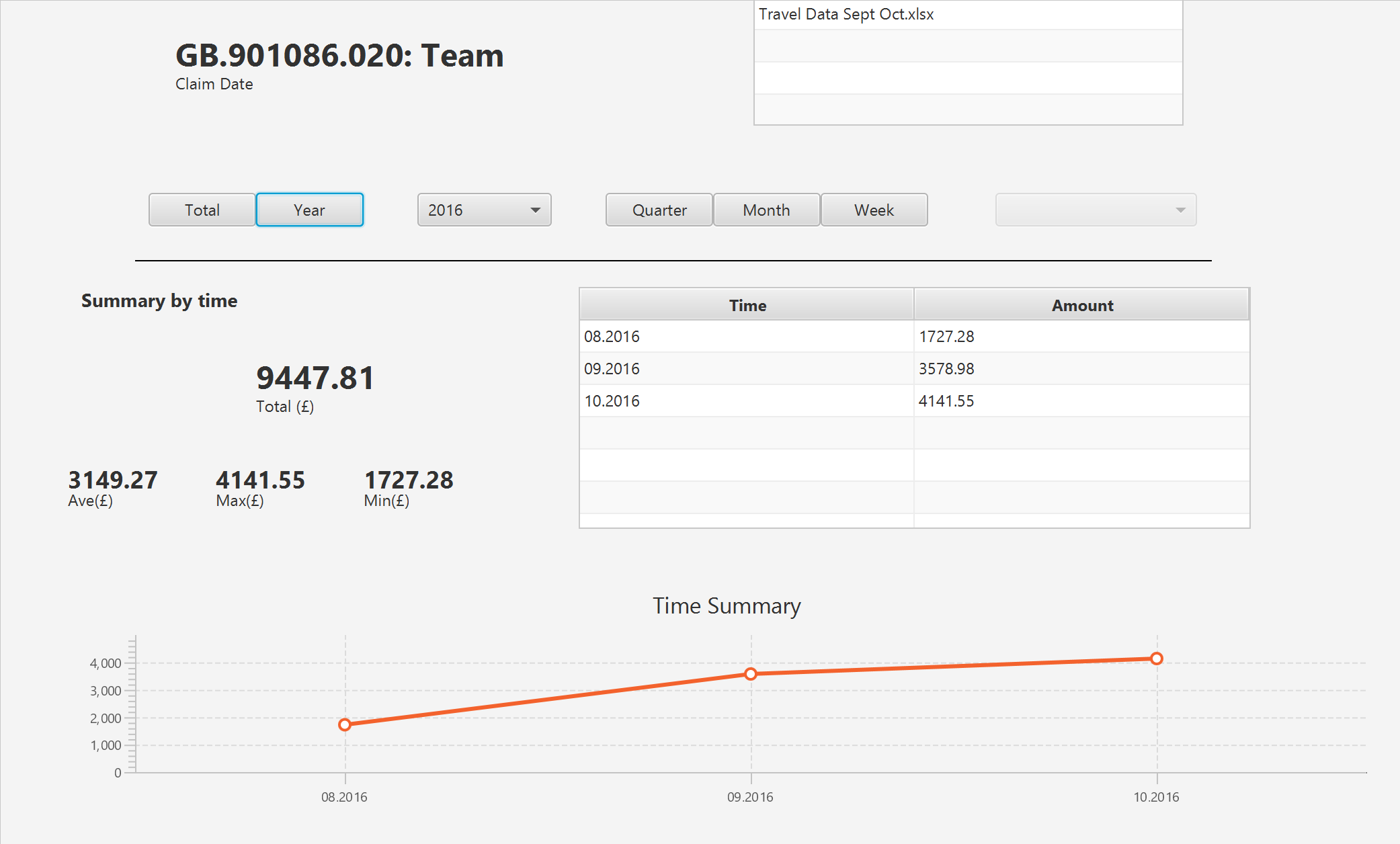


The total of all the expenses found in the file.

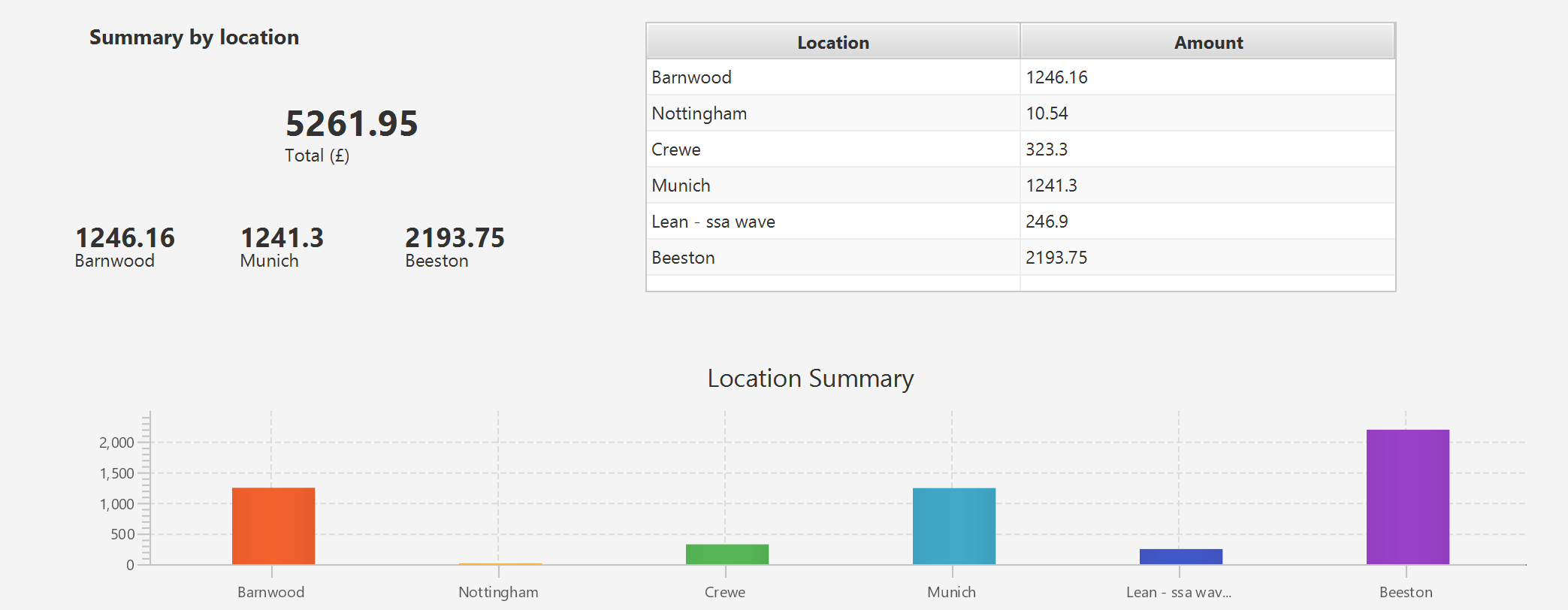
Shows weekly expenses

Shows monthly cost

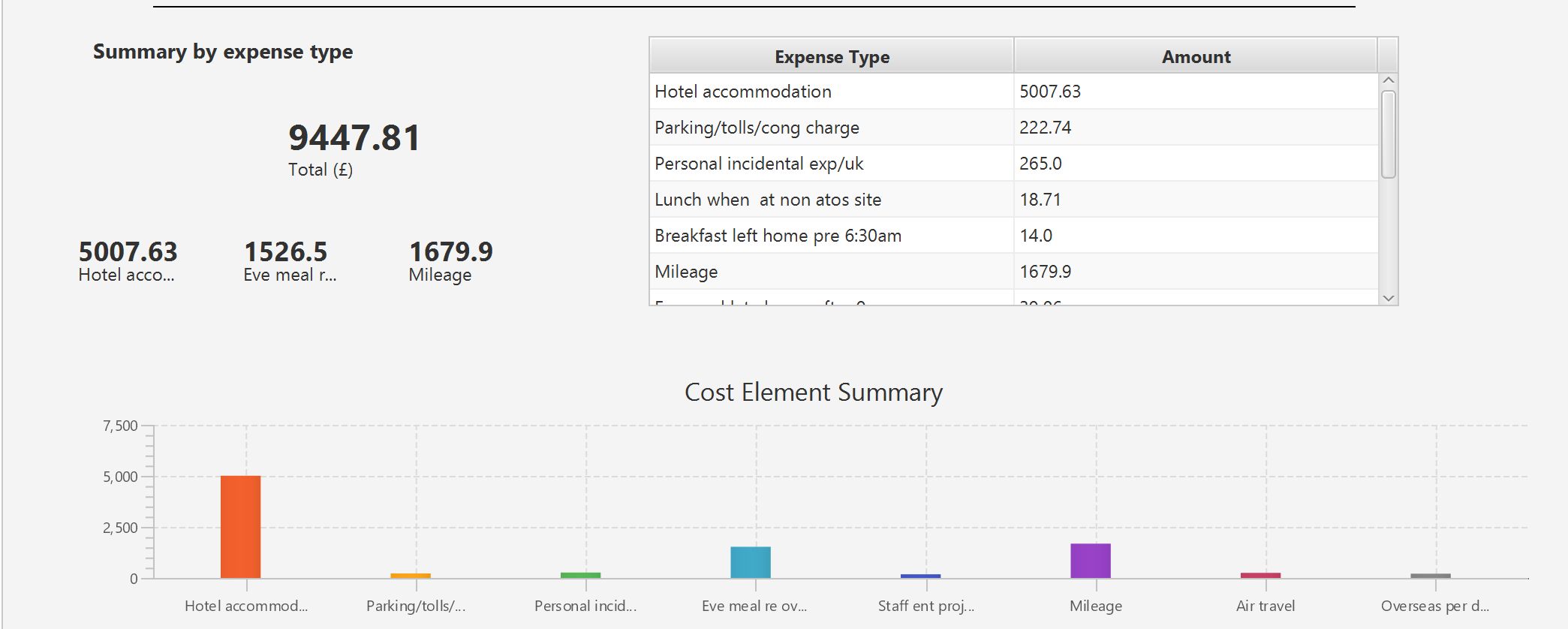
Shows the Quarterly expense result.



This option shows the monthly expenses in a year.



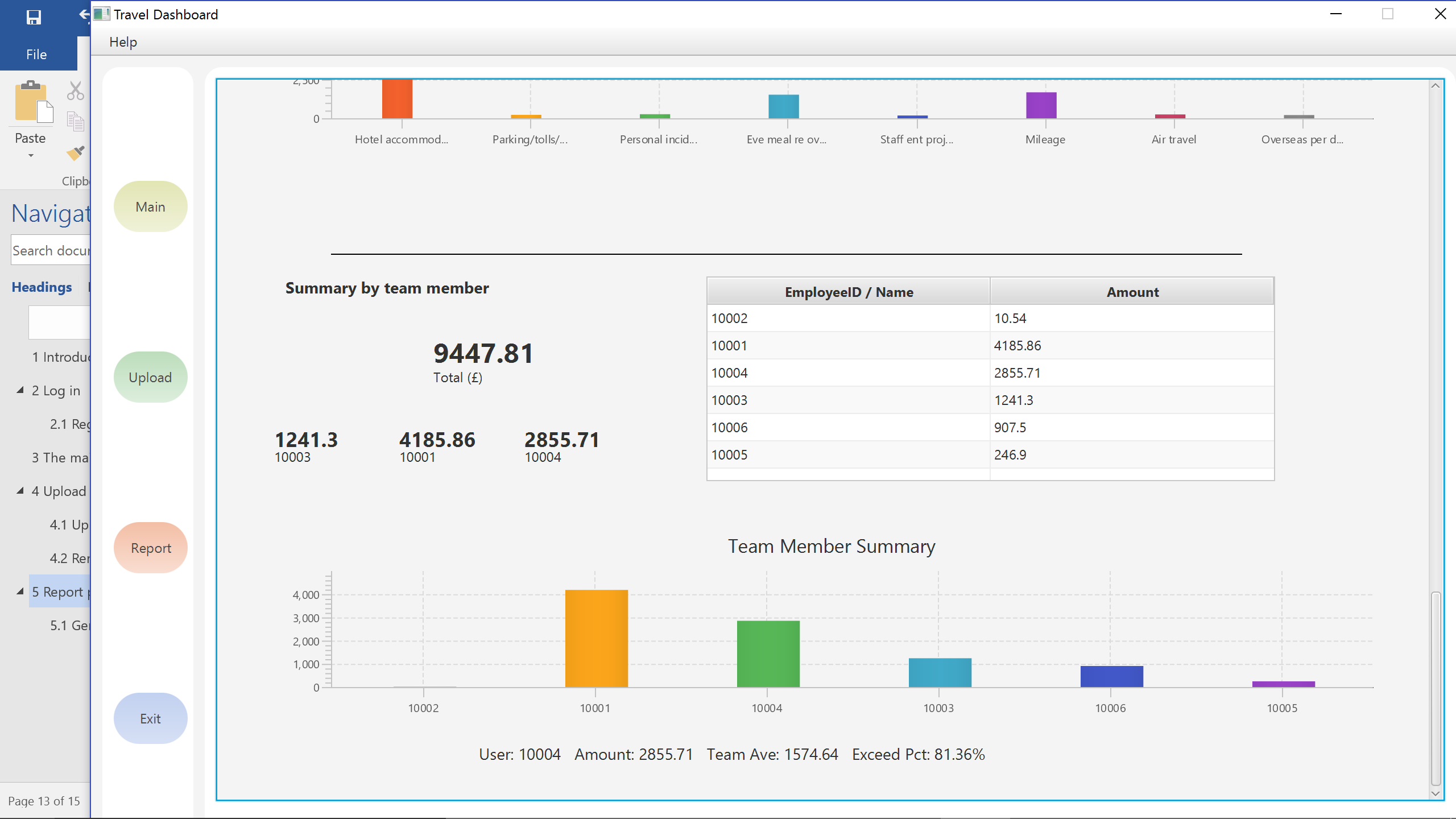
The above image shows the section that illustrates the expenses by location.



The next section shows the expense of each element.

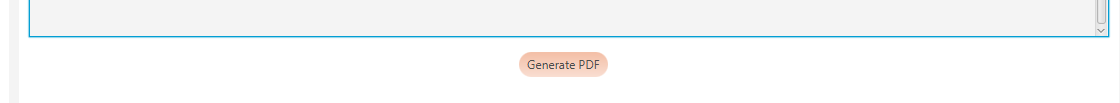


This section shows the team members’ expenses. This also categorises the team members by their expenses (i.e.- finds the employee who spent the most and employee who spent the least etc.)

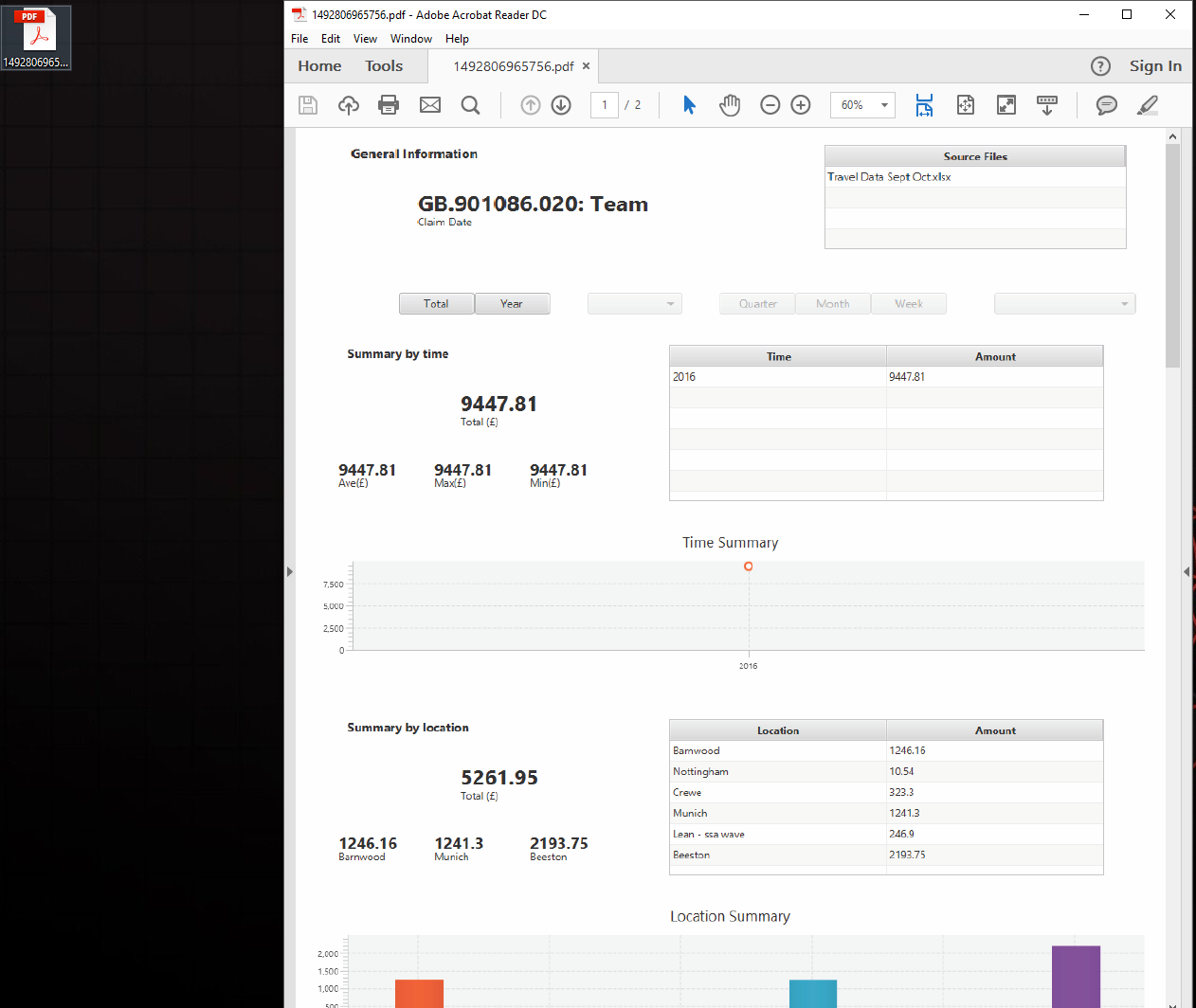


The bars in this chart can be hovered over to view each of the employee’s expenses and other information. This information includes how much they have spent compared to the average of their other team members in the team.

### Generating Report



This is the last part of the Report page which generates PDF report for the user. Once this button is clicked the system generates a pdf report in the desktop of the user as it can be seen below-



# APPENDIX

## Spreadsheet Data Format

Travel Dashboard application is compatible with two spreadsheet formats: FileType1 and FileType2 which differ in the number and order of columns used for analysis.

**File Type 1**

1. Doc. Date
2. WBS Element
3. Cost Elem.
4. ValCOArCur
5. COCr
6. Value TCur
7. TCurr
8. Last/First Name
9. UM
10. Created on
11. DocumentNo
12. Cost element name
13. Cost element descr.
14. Doc.Header Text
15. DocTyp
16. Per
17. Name
18. RefDocNo
19. Quantity
20. Vbl. value
21. Val/ObjCur
22. Finance Process
23. AInd
24. Month
25. Time Month
26. Person Reference Number

**File Type 2**

1. Expense Date
2. Payment Date
3. DisplayDate
4. Week
5. Month
6. Year
7. Category Group
8. Expense
9. Employee with SAP ID
10. Employee with DAS ID
11. Organizational Unit
12. Category
13. Summary
14. DocNo
15. Mileage
16. Location
17. GLCODE
18. Trip Activity Type
19. Trip Start Date
20. Trip End Date
21. Payment Currency
22. Payment Settlement Date
23. Number of Receipts
24. Trip Cost
25. Mileage Expenses
26. Atos Company Code
27. Receipt Date
28. Receipt Location
29. Receipt for No of Staff
30. Expense Type
31. GL Category
32. P&L Category
33. Employee Profit Centre
34. WBS Profit Centre
35. Rechargable Indicator
36. Activity Type
37. Country
38. Service Line
39. Tower A
40. Tower B
41. Tower C
42. Tower D
43. Tower E
44. WBS
45. WBS Billable Indicator
46. WBS Owner with SAP ID
47. WBS Owner with DAS ID
48. Customer Number
49. Customer Name
50. Customer Type
51. Project
52. Type of expense
53. Level 2 Sector Name
54. Level 3 Market Name
55. Level 5 Customer Name
56. Cost Centre
57. Cost Centre Description
58. Cost Centre Owner with DAS ID

**Note- Please make sure the columns (in the excel file) match one of the file type discussed above.**

# Installation Guide

This document provides instructions for installing the Travel Dashboard Software for Atos developed through a group project by students in the School of Computer Science at the University of Nottingham.

Travel Dashboard is a standalone desktop application that enables the user to visualize statistics from uploaded data file in Microsoft Excel format. The application was tested for Excel 2007-2016.

Travel Dashboard application requires MySQL database in order to store and process the data. It has been tested with MySQL version 5.6.35. Ultimately, the application will connect to Atos database. However, that has not been enabled as part of the project.

***Hardware requirements***

**Recommended hardware** for installing the software comprises

1. Core i3/i5 processor (or higher)
2. 4 GB RAM (or higher)
3. 128 GB hard drive (or higher).

**Software requirements**

Application requires the following **software**:

1. Microsoft OS Windows 7 or later
2. Java Development Kit JDK 8 (Oracle)
3. Eclipse Neon 2 (www.eclipse.org).

**Outline**

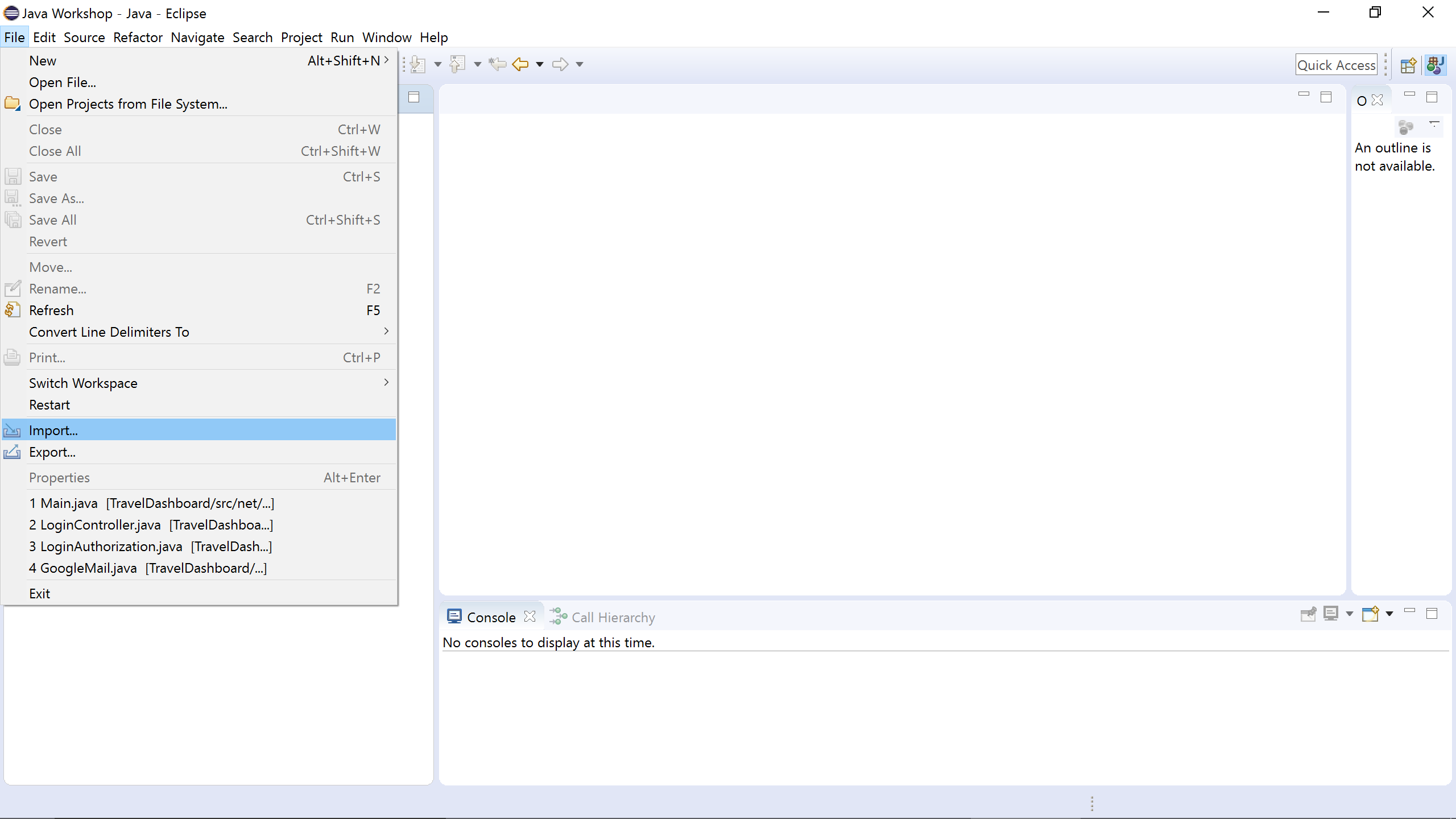
The software is distributed to Atos through a Zip file that includes source code of the application. It does not include MySQL software, which can be obtained from <https://www.mysql.com/>.

If the source code is modified, one needs to generate new .jar using Eclipse and then create a new .exe using a converter from .jar to .exe. For that latter, one can use Luanch4J (<https://sourceforge.net/projects/launch4j/files/latest/download>)

In order the install the software, one should follow the three steps described in this document. They involve:

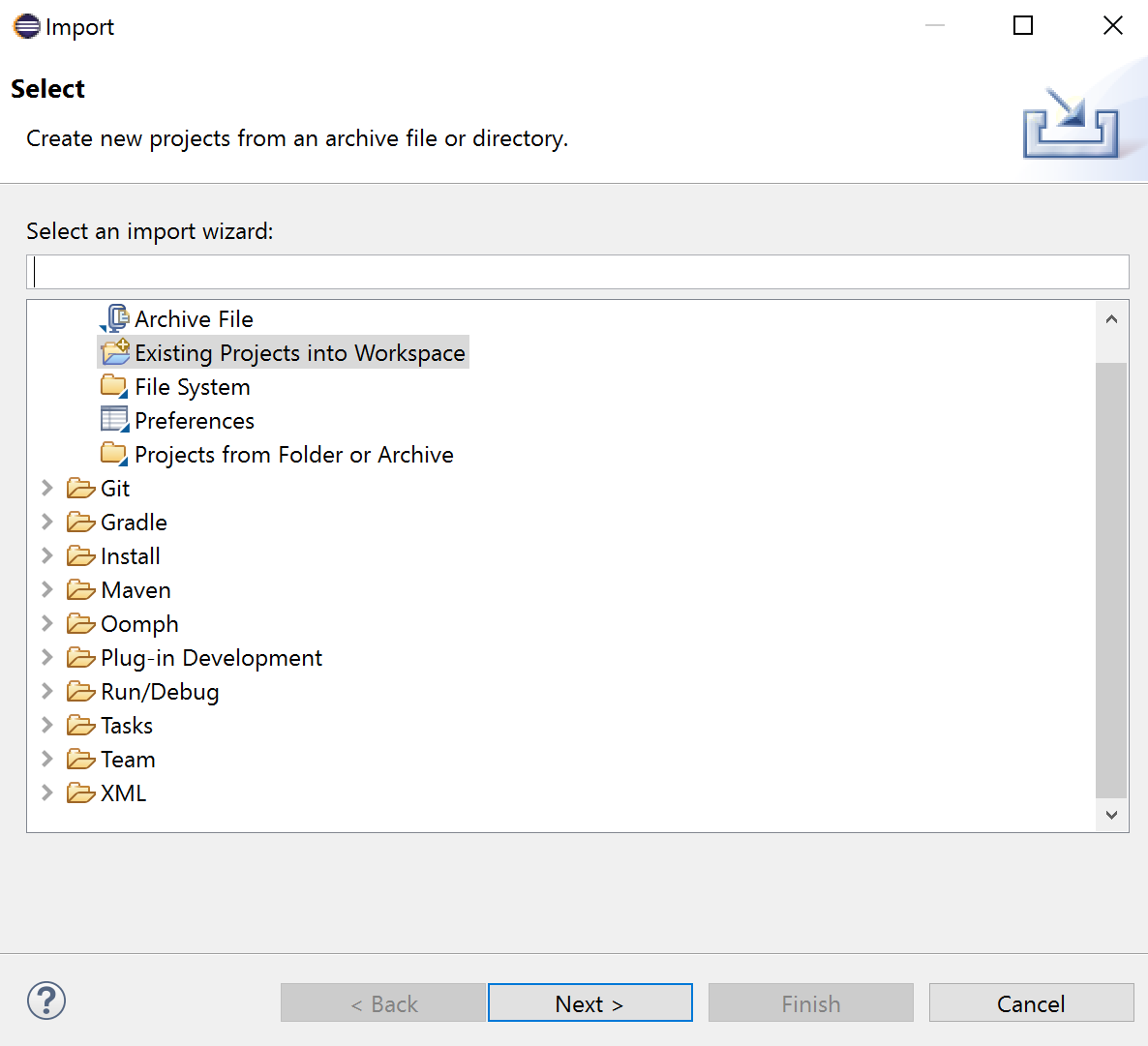
1. Import of Java file into Eclipse
2. Create .jar file using Eclipse
3. Create .exe file using Launch4J

## How to Import in Java

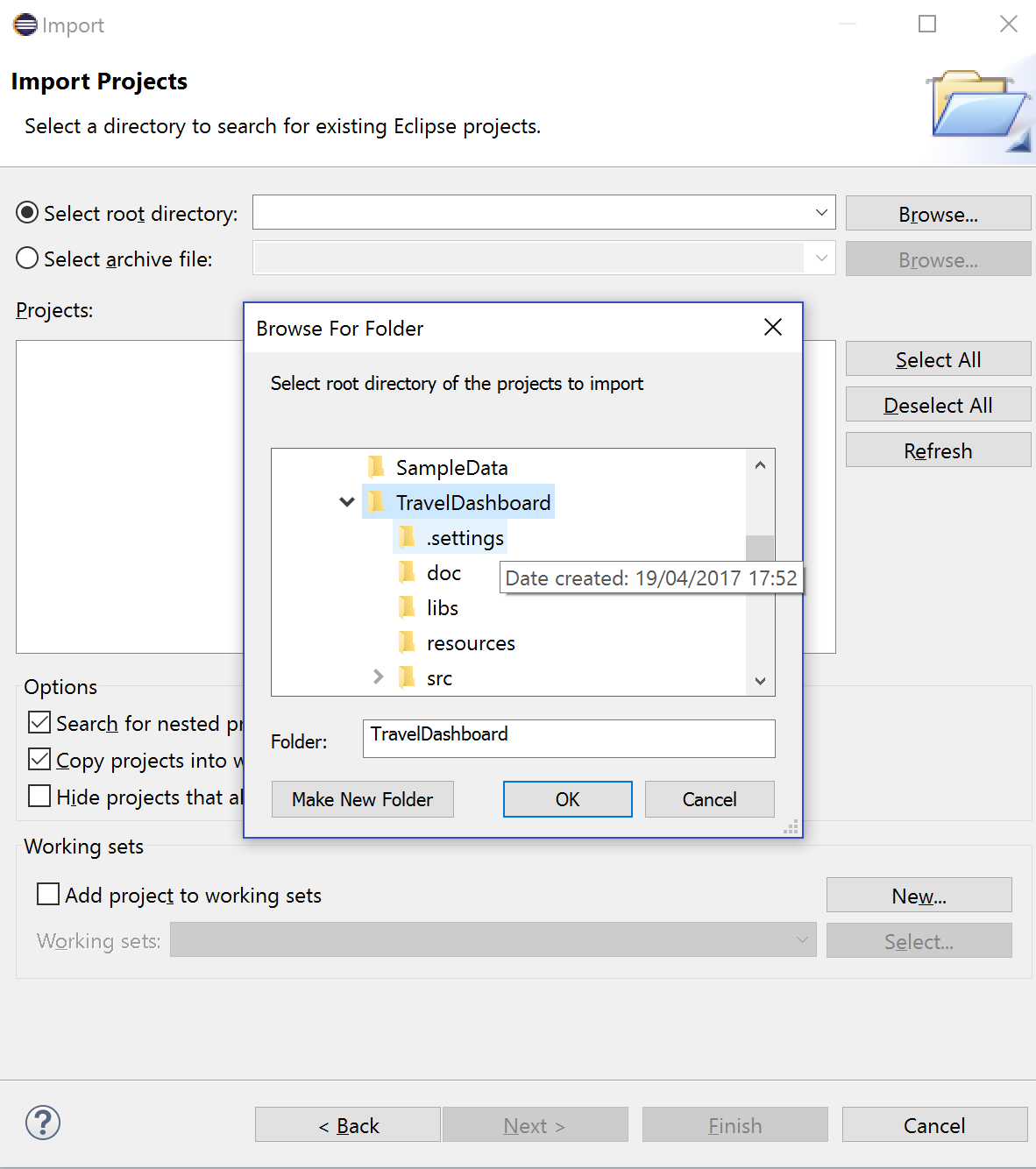


To import a java file click on the **File** option and then click **Import.**

This window will appear after choosing **Import option.**



Choose **this option** and press **Next.**



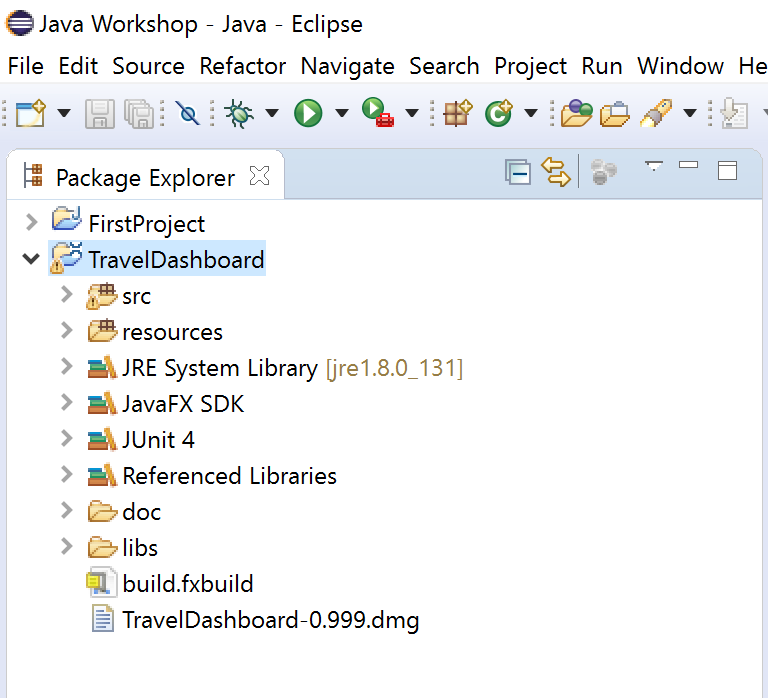
If importing a **zip file** choose **Select archive file**

If importing an **unzip file,** choose **Select root directory**

Choose **Browse** to import the file.

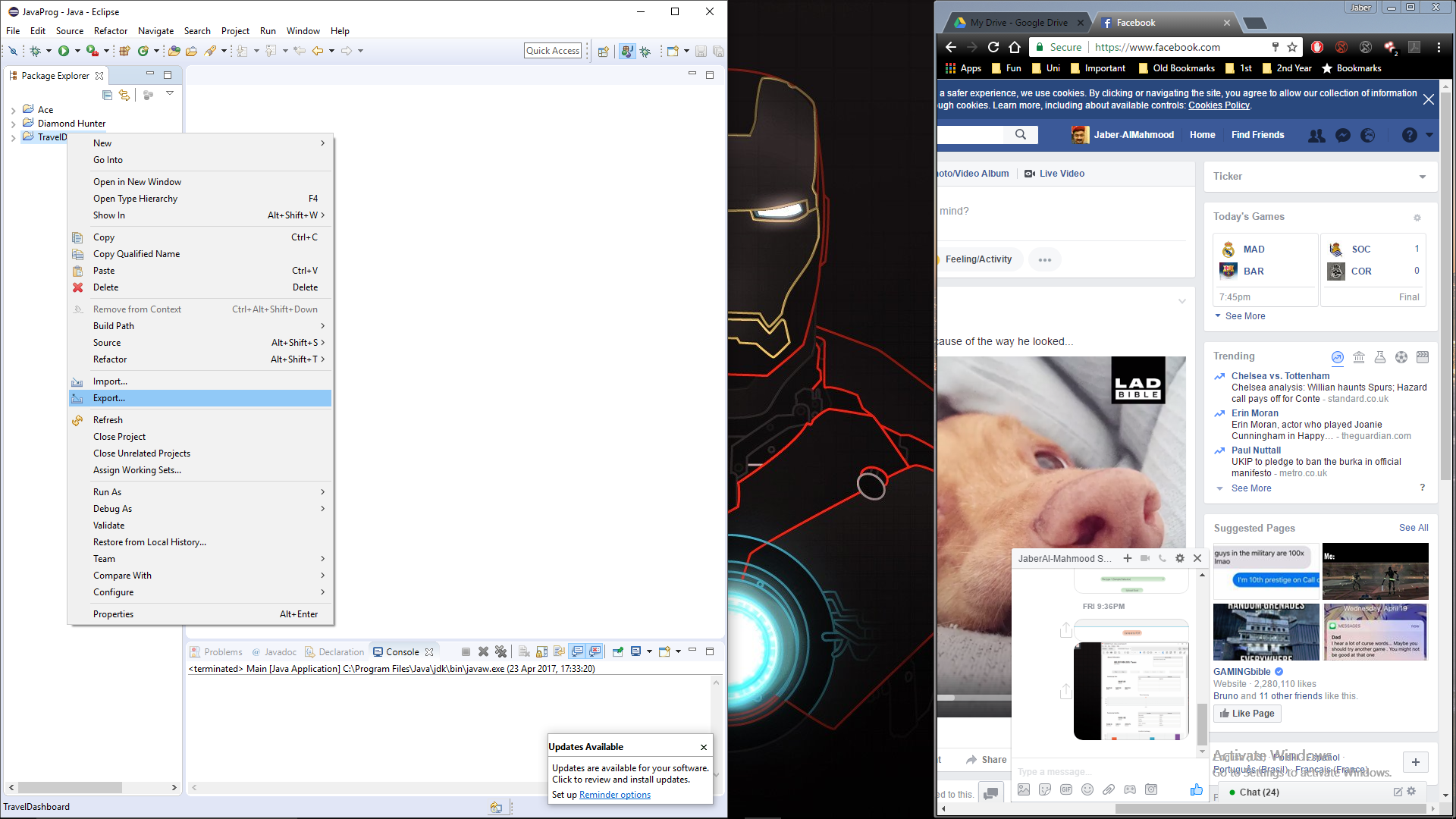
Choose **the correct** file and press **OK.**

|  |  |
| --- | --- |
|  | This shows that file can be imported  This shows that the file that should be imported  Press on Finish to import the file  This is the file that has been imported. |

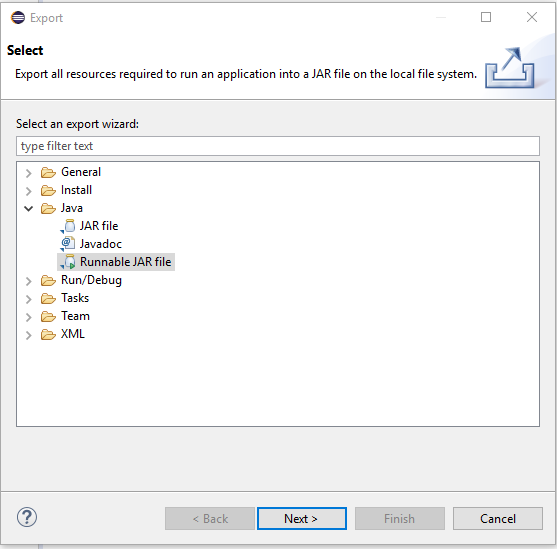


This is the file that has been imported.

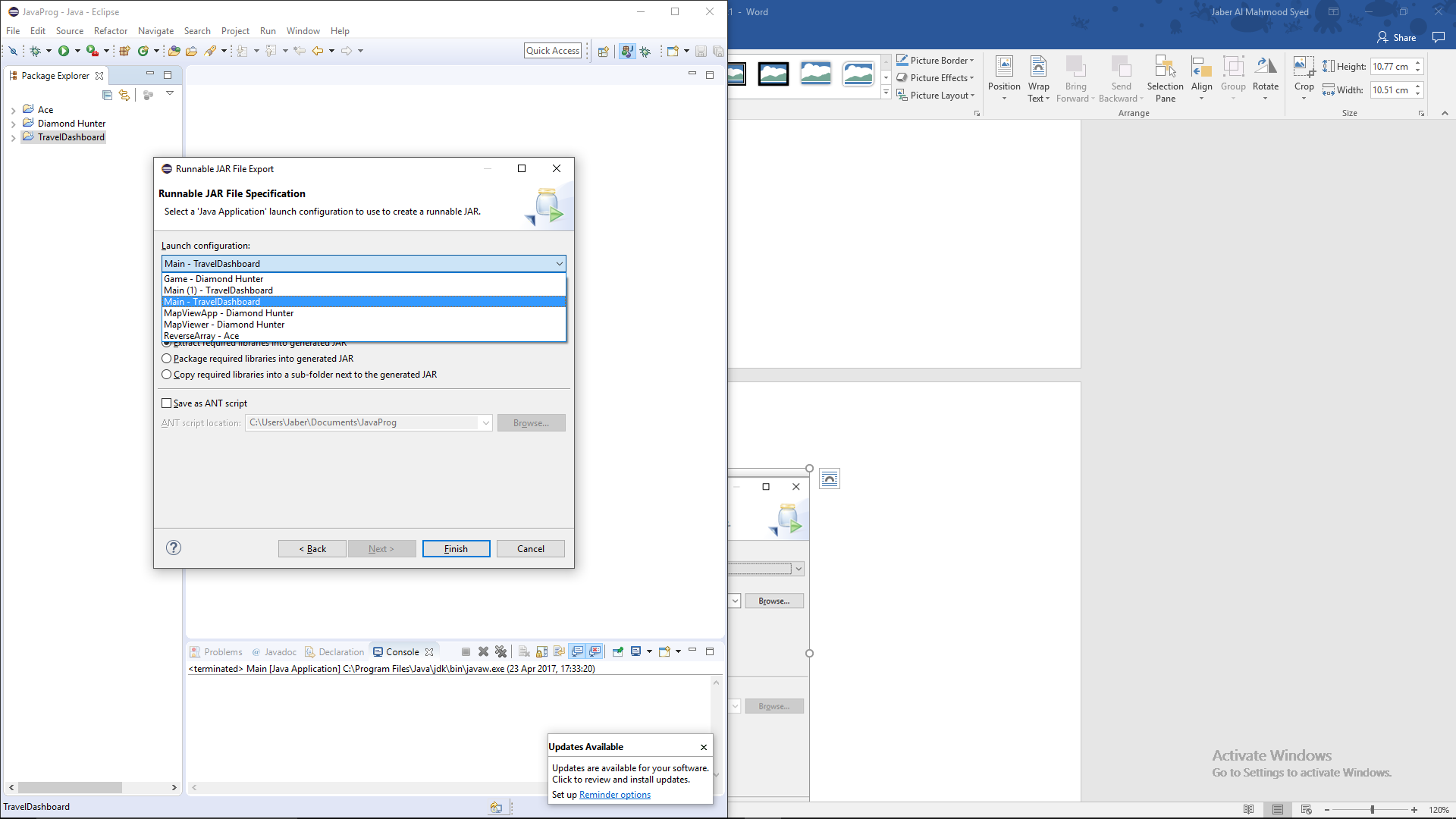
## How to Produce .JAR File



Right click on the Java project and choose the “Export” option.



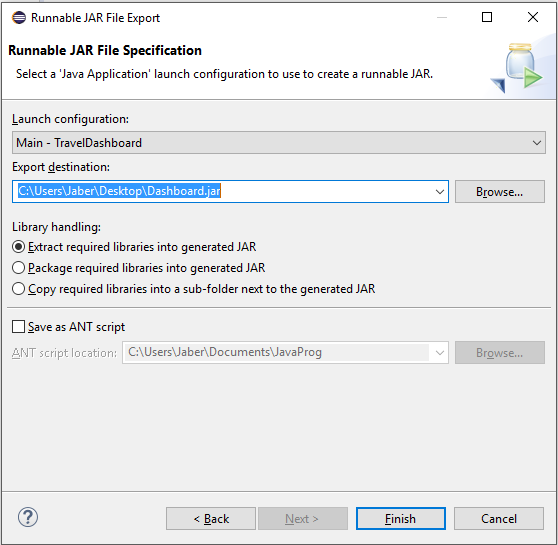
Right click on the Java project and choose the “Export” option.



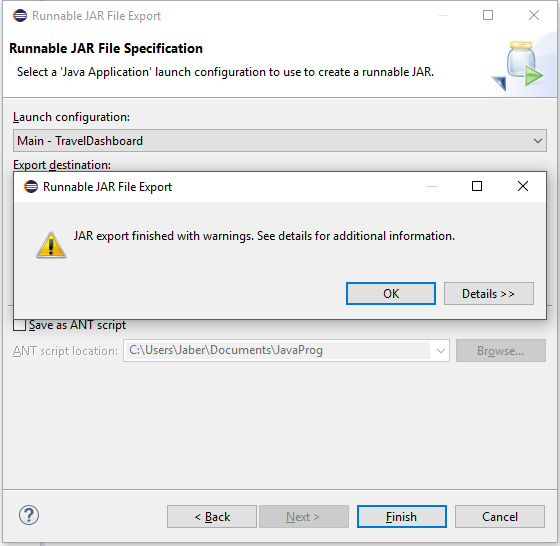
Choose the **Main class** from the drop-down list (in our case “Main-Travel Dashboard.”)

Once the “Export” option is chosen, this is the window that appears.

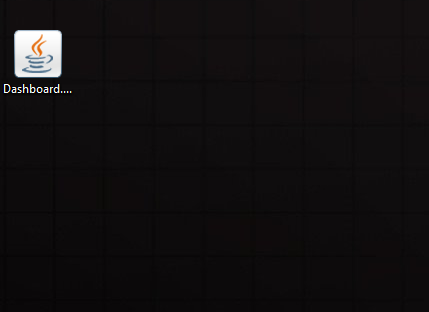
After pressing “Next” this is the window that will appear on the screen.



After choosing the **Main class of the project** choose the path where you want to save the .jar file and press **Finish.**



This is the window that will pop up once the **Finish button** is pressed. Press **OK** to continue.



Once pressed **OK,** this is the .jar file that will be produced. You can double click on this application to open the software.

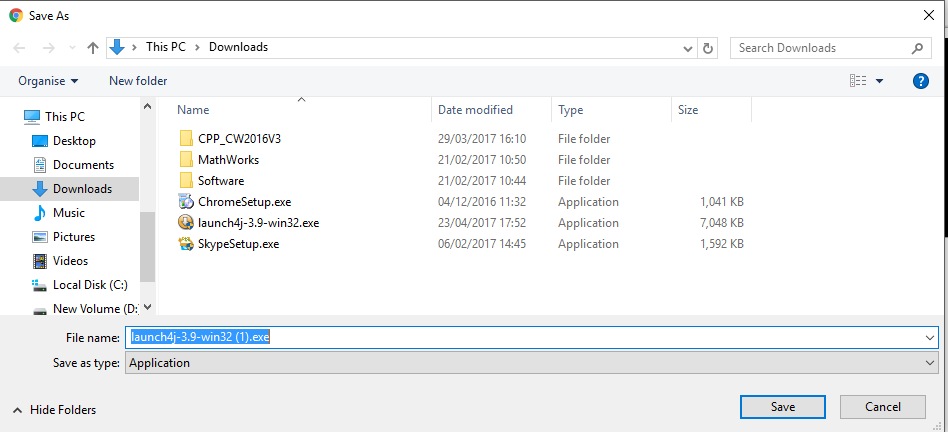
## Producing .exe File

To produce the .exe file for the software, we have used **“Launch4j” software.** To download this software please visit this link-

<https://sourceforge.net/projects/launch4j/files/latest/download>

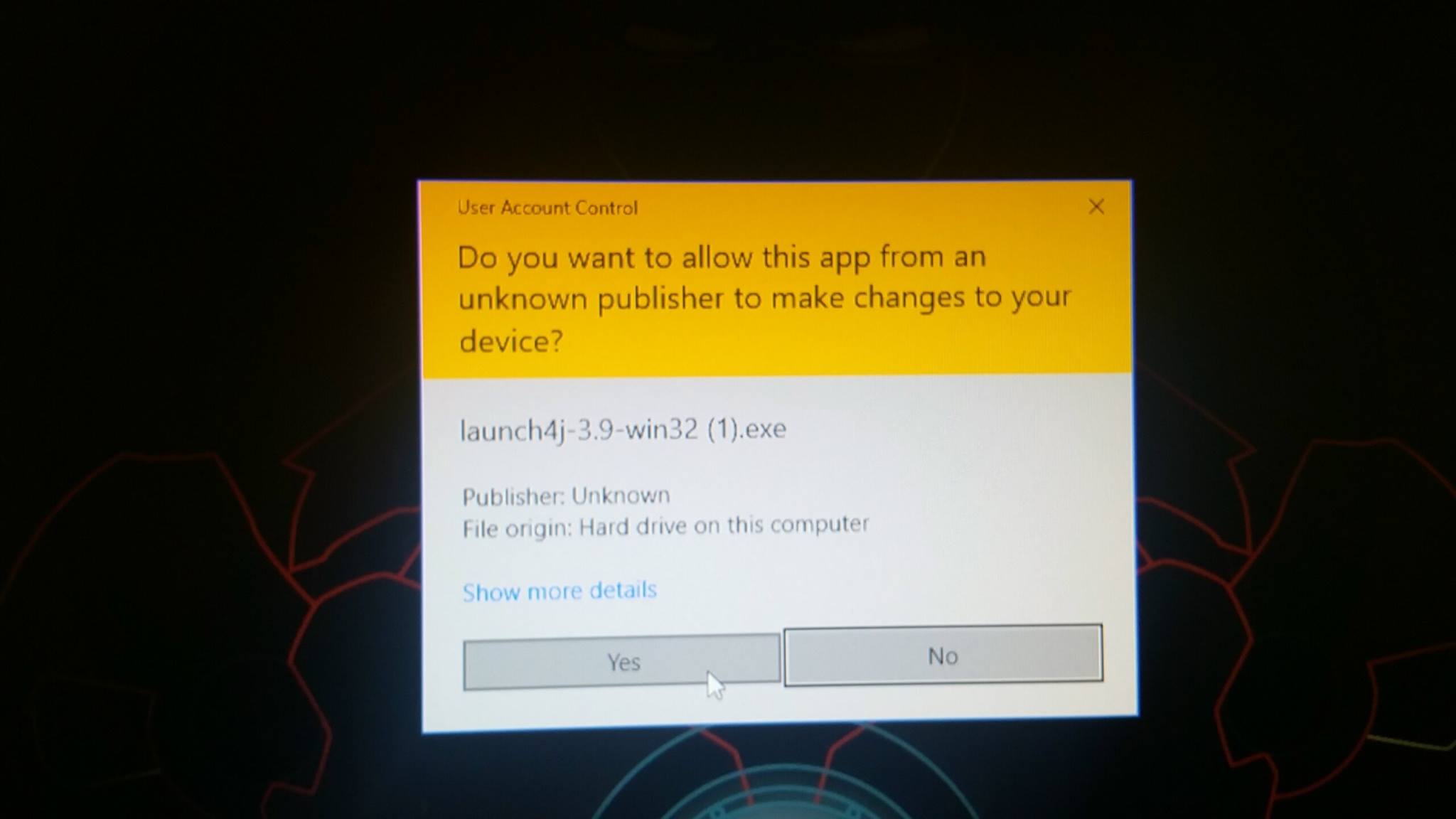
**Download Launch4j**

Once you visit the link above, this window will appear.

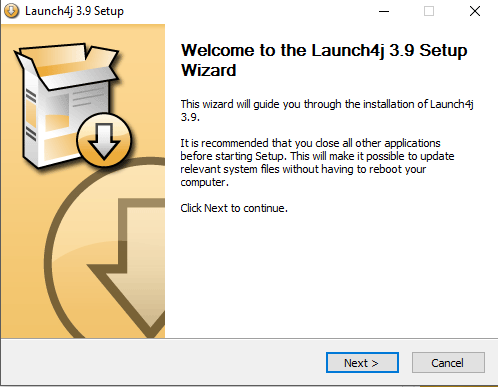


Choose the path where you want to download the software and press **save.**

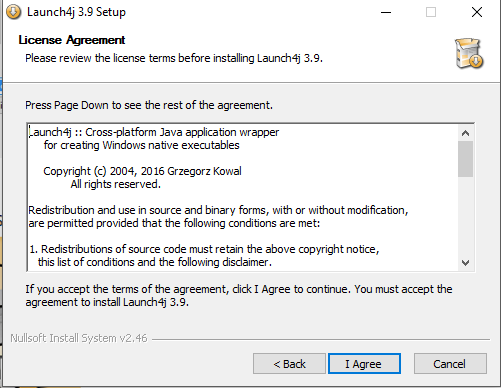
After saving the software, run it on the computer. The following window will pop-up once you run the software. Press **Yes** to continue.



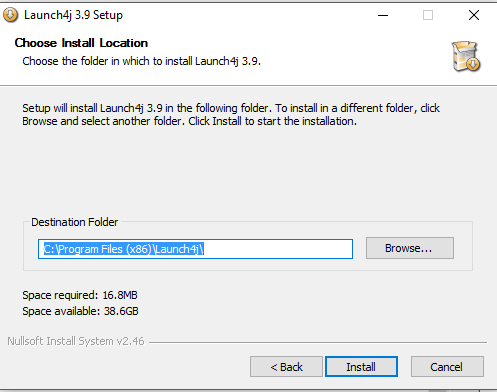
Once the **Yes** button is pressed then this window will be shown on the screen.



Press **Next** to continue.



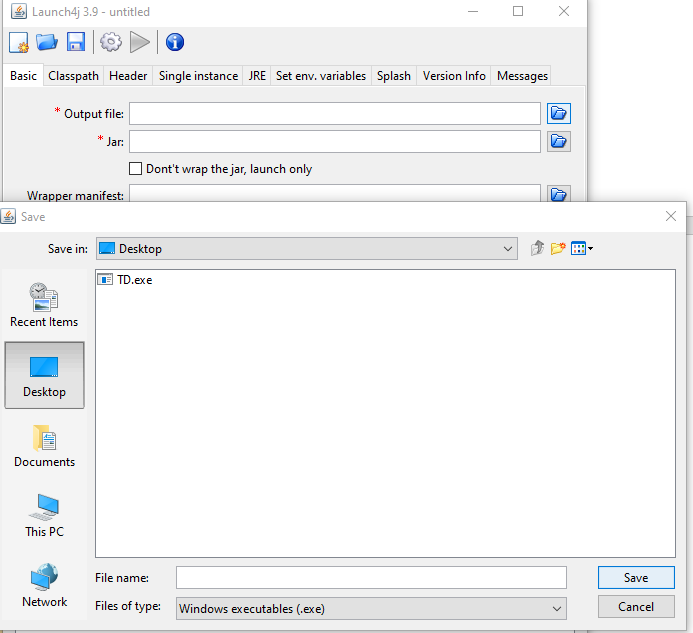
Click **I Agree** to continue.



Choose the path where you want to install the software and press **Install button.**

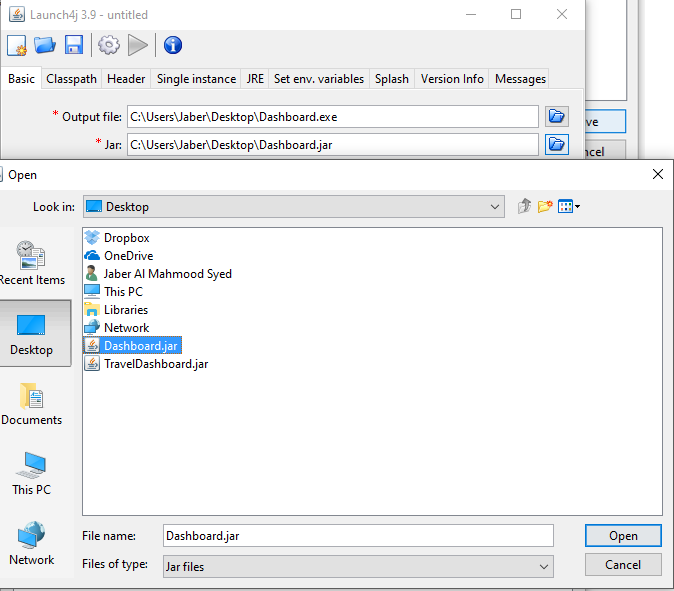


Press **Finish** to finish the installation process.



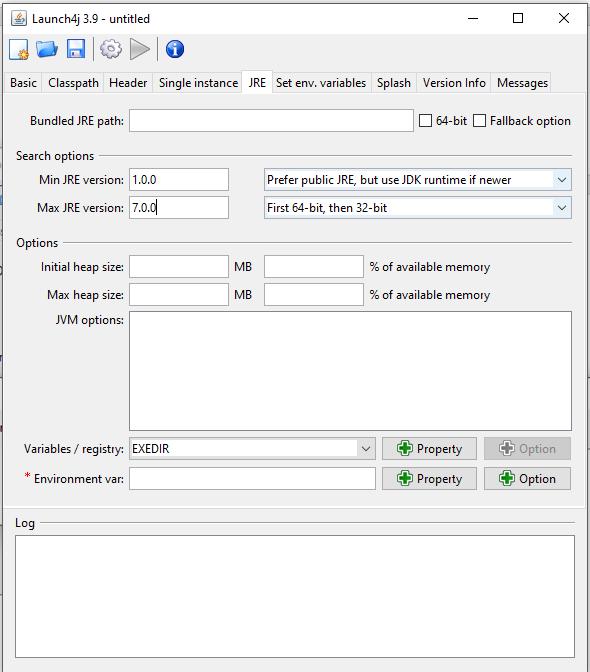
Give the .exe file a suitable name and click **Save.**

Choose the path where you want to save the .exe file.

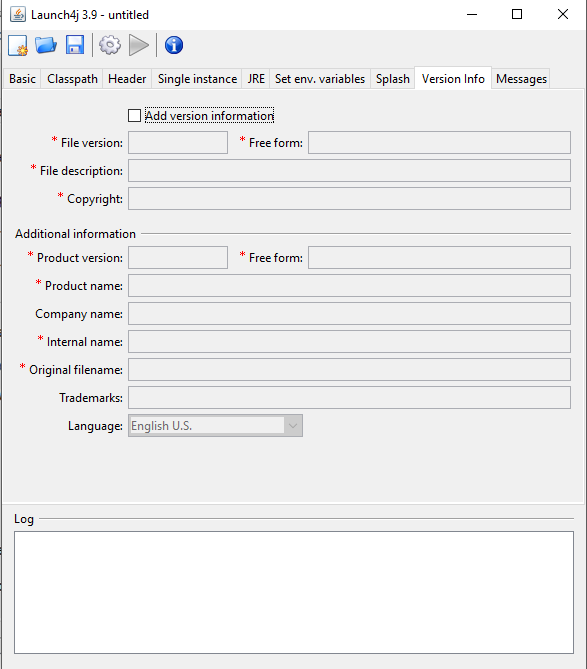


Choose the .jar file that you have produced by clicking this button.

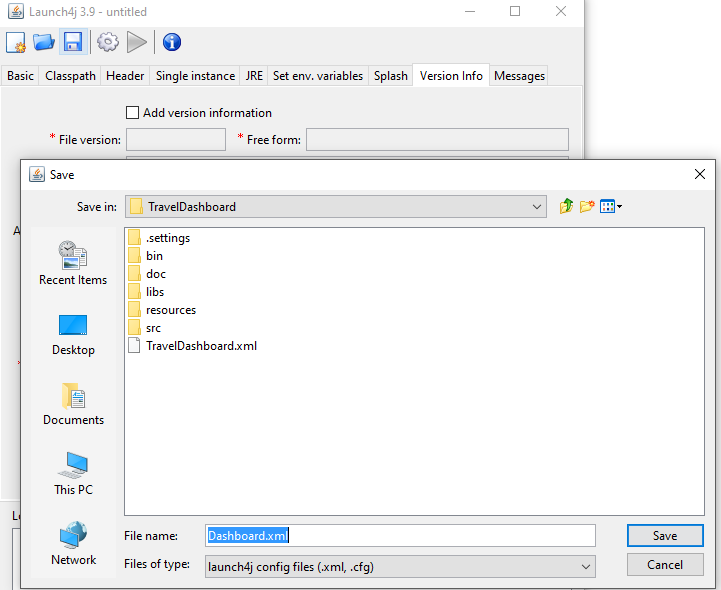
Choose the .jar file that you have produced and press **Open.**



Visit **JRE** from the menu to specify the min/max **JRE versions.**

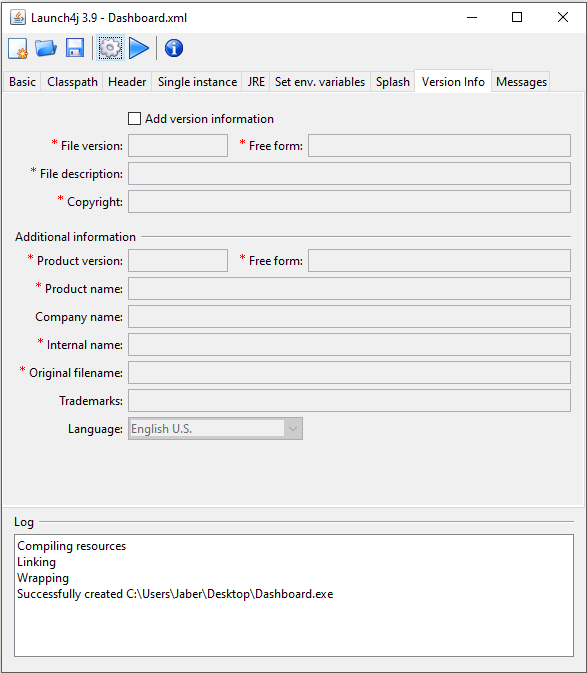


You can also visit the **Version Info** option to specify the version of the software and related information.



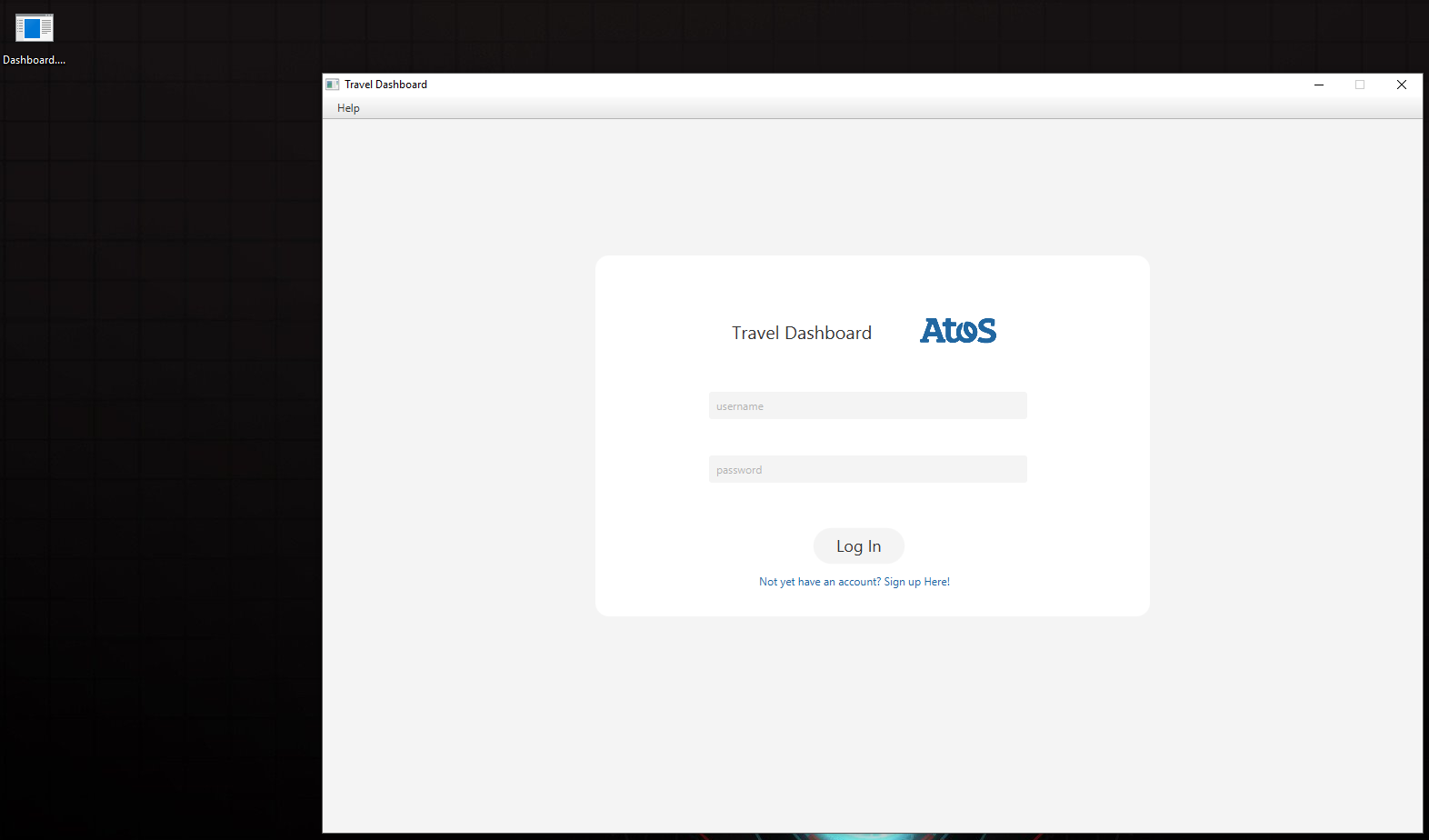
Choose the path (recommended to save it inside the project file), name the file with **.XML** type and click **Save.**

Choose **Save Option.**



Click **this button** to generate the .exe file successfully.

This shows the .exe was created successfully.



This is the software that opens when the .exe file is clicked.

This is the .exe file.

### Helpful Sources-

**Install Java**

<https://www.youtube.com/watch?v=Hl-zzrqQoSE>– JDK Installation

<http://www.oracle.com/technetwork/java/javase/downloads/index-jsp-138363.html> - JDK from Oracle.

<https://www.youtube.com/watch?v=CE8UIbb_4iM> – Download eclipse IDE

<http://www.eclipse.org/downloads/eclipse-packages/> - Eclipse Neon

**Note- Please download the latest versions**

**Generating .Jar and .exe Files**

<https://www.youtube.com/watch?v=mE3rbtKm-pk>– Generating .jar file

<https://www.youtube.com/watch?v=-n62XdODrK0> – Converting .jar to .exe using Launch4j.

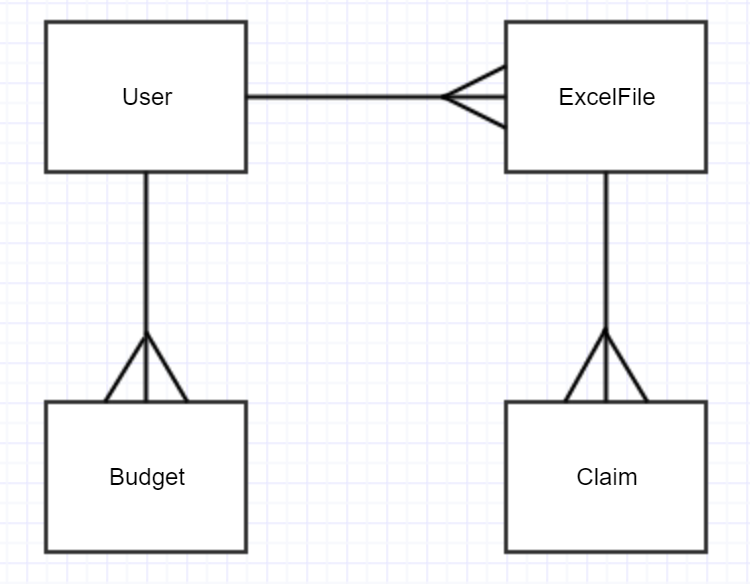
# Database

This section of the report illustrates how the database system is used in the system. The database diagram shows all the tables that are made for the system. It also displays the connection between each table. The “My SQL Table” section explains the queries which are requested from the system and how the data is examined by using these queries.

## Software Used

* MySQL
* PHPMyAdmin

## Database Diagram



This diagram shows the tables and their connections in the database. There are mainly four tables-

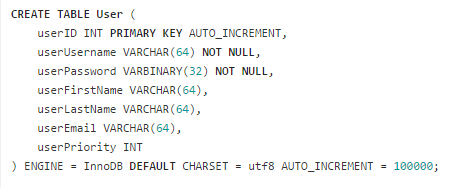
User Table- This table is used for storing user information when they log in.

ExcelFile Table- This table contains information about an excel file when they are uploaded by the user. As it can be seen from the diagram, one user can have multiple files uploaded in the system. However, one file cannot be shared between multiple users (although multiple users can upload the same file but it will be still saved with a different id in order to identify that specific excel file for that specific user.)

Budget Table- Budget table is used to store the budget that is set by the user. It has one to many relationship with “User” table. This means that one user can set budget more than one time.

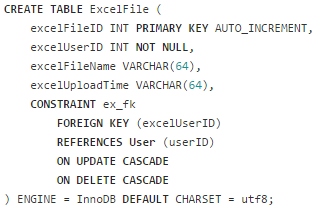
Claim- The claim table is used to store all the details about a single claim that can be found in a file. This is the reason why it has one to many relationship with ExcelFile table. This means that one excel file can have several claims, however, one claim cannot be in more than one excel file.

## User Table



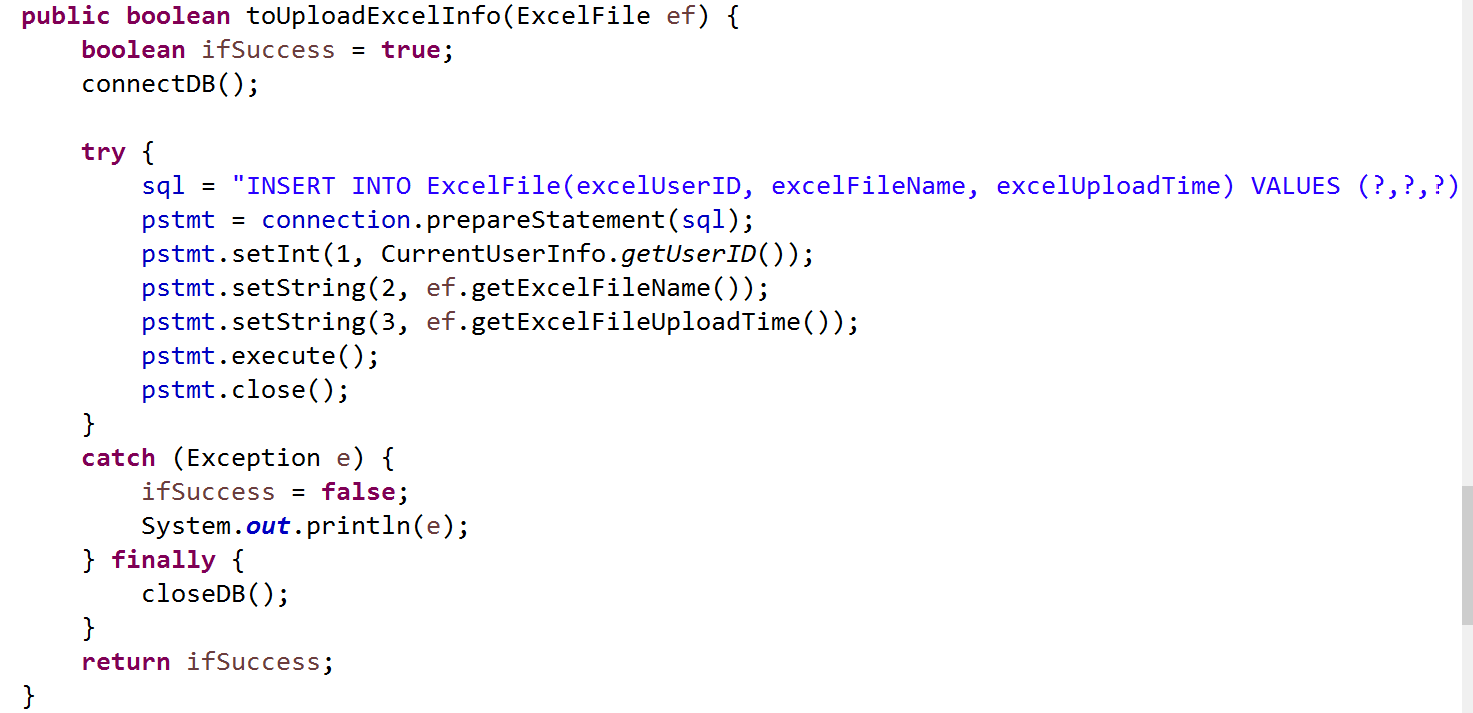
This is the user table that contains user-name, password, email address when the user signs up to the system. Each user gets a unique ID that starts from 10000 and the database uses Auto-increment to give unique ID to each user.

## Excel File Table

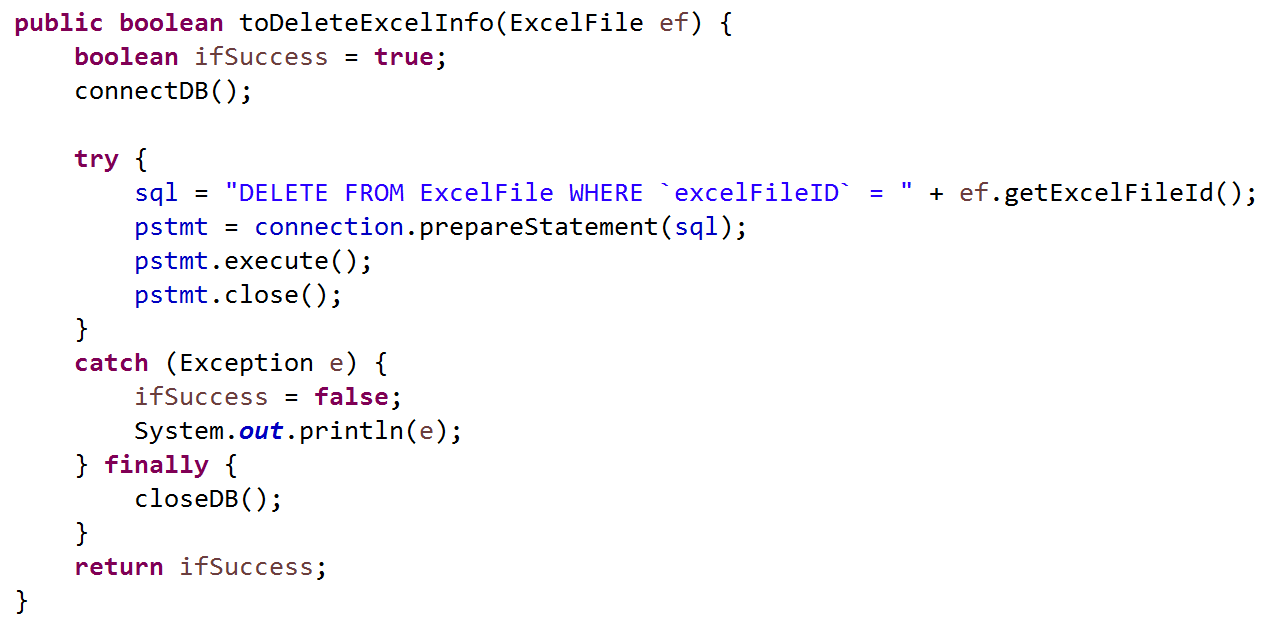


The “ExcelFile” table contains information about the files that are uploaded by each manager in the system. As each manager will have different account, so the excel files will vary based on the files the managers upload in the system.

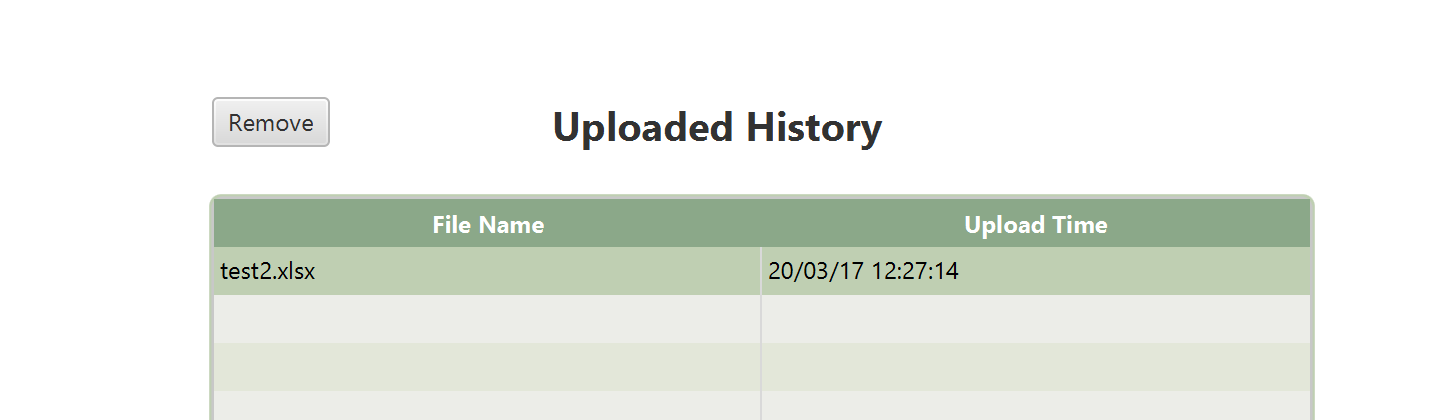
Queries from the Table



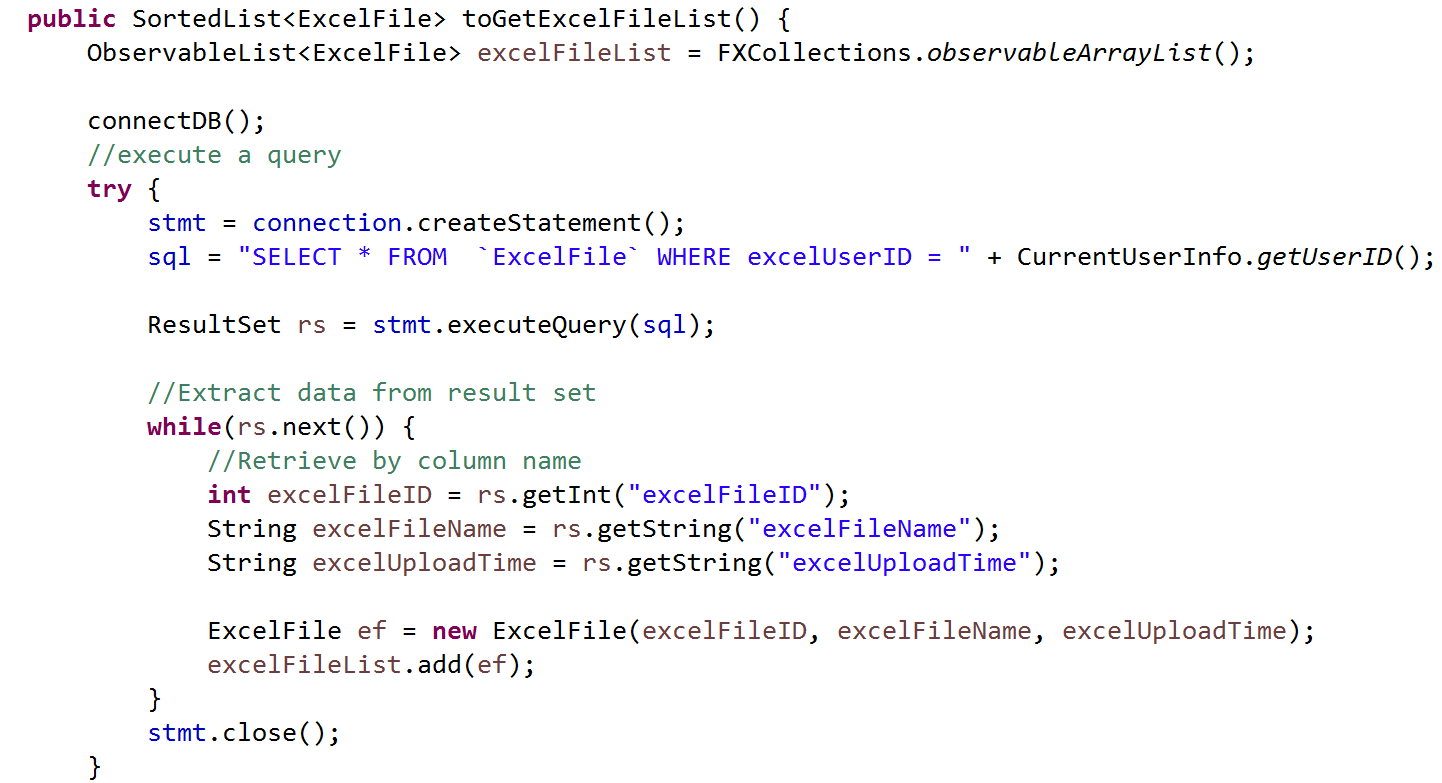
This is the method that keeps record of the user by inserting the file information inside the ExcelFile table. This is to help showing the status of the user in the system.



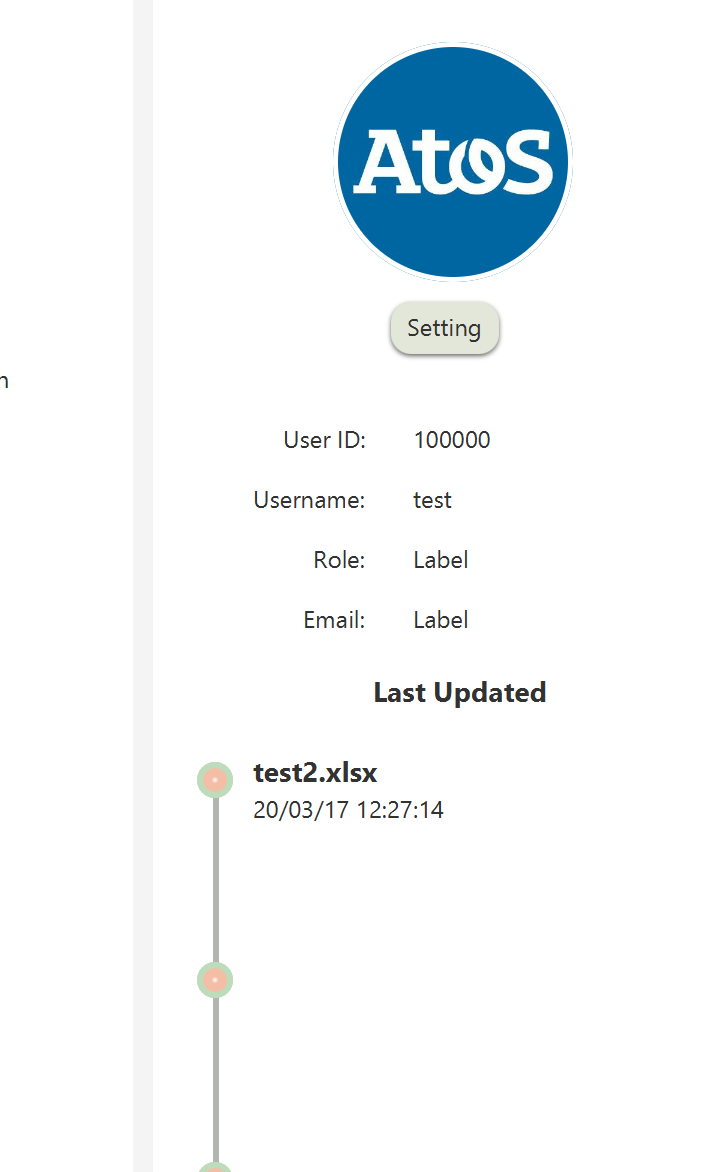
This method is used to delete the files from the database. The system allows the user (will allow the managers) to delete certain files if they want to. Each file is given a unique ID and the query finds the ID of the file that needs to be deleted and deletes all the data that is inside that file.



The image above illustrates a section of the “Upload” page in the system. The “Remove” button uses the above-mentioned method to delete all the data that belongs to a single file.

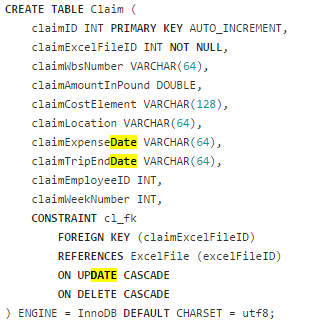


This method is used to show all the files uploaded by a user in the system.



This is the part where the above-mentioned method is used in the system. This image shows a part of the “Main” page in the system.

Claim Table



This table contains all the information that are inside a file. This references the “excelFileID” from the “ExcelFile” table. This is to make sure that all the data that is inside a file is shown. This also helps when it comes to deleting the files. For example- if a manager wants to delete a file, then they only need to delete the file instead of deleting each row in that file in the database.

### Queries from the Table

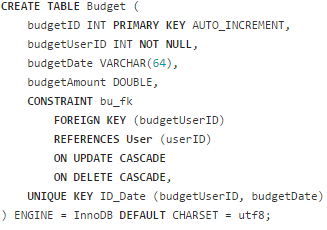


This is the method that finds out all the claims a single employee has made. The query finds all the claims of an employee from the different files (if there are more than one file that contains that specific employee’s information.)



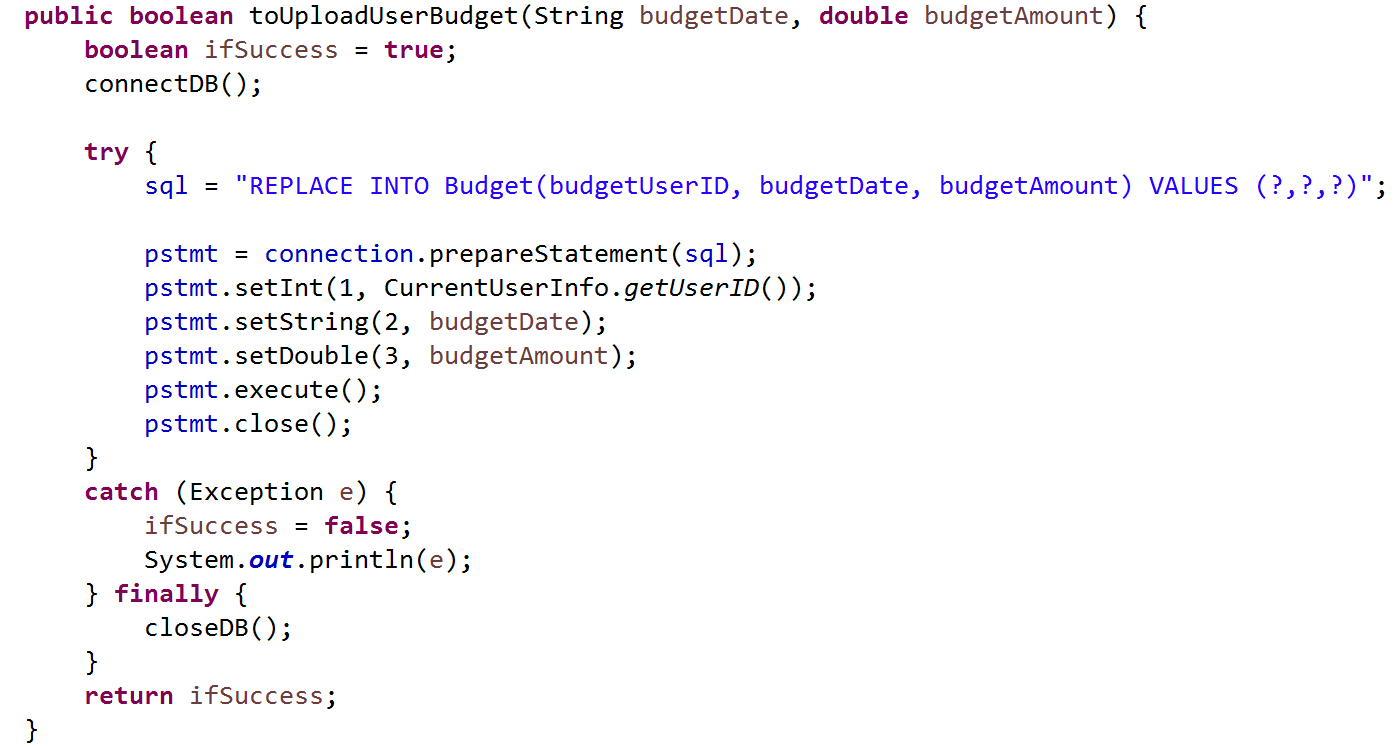
This method is used to insert all the claims into the database. This is mainly based on each spreadsheet and as it can be seen, only the essential columns are inserted into the database to analyse the data.

## Budget Table

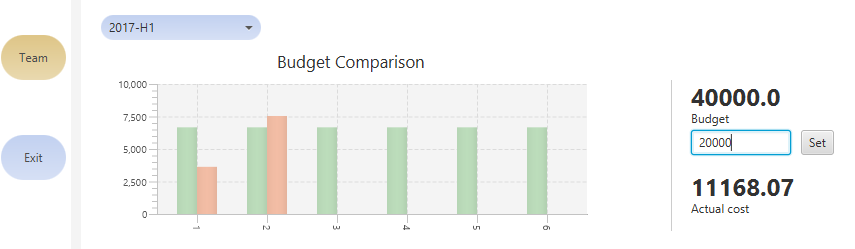


This is the budget table that holds information about the budget when it is specified by the manager in the “Main Page” of the system.

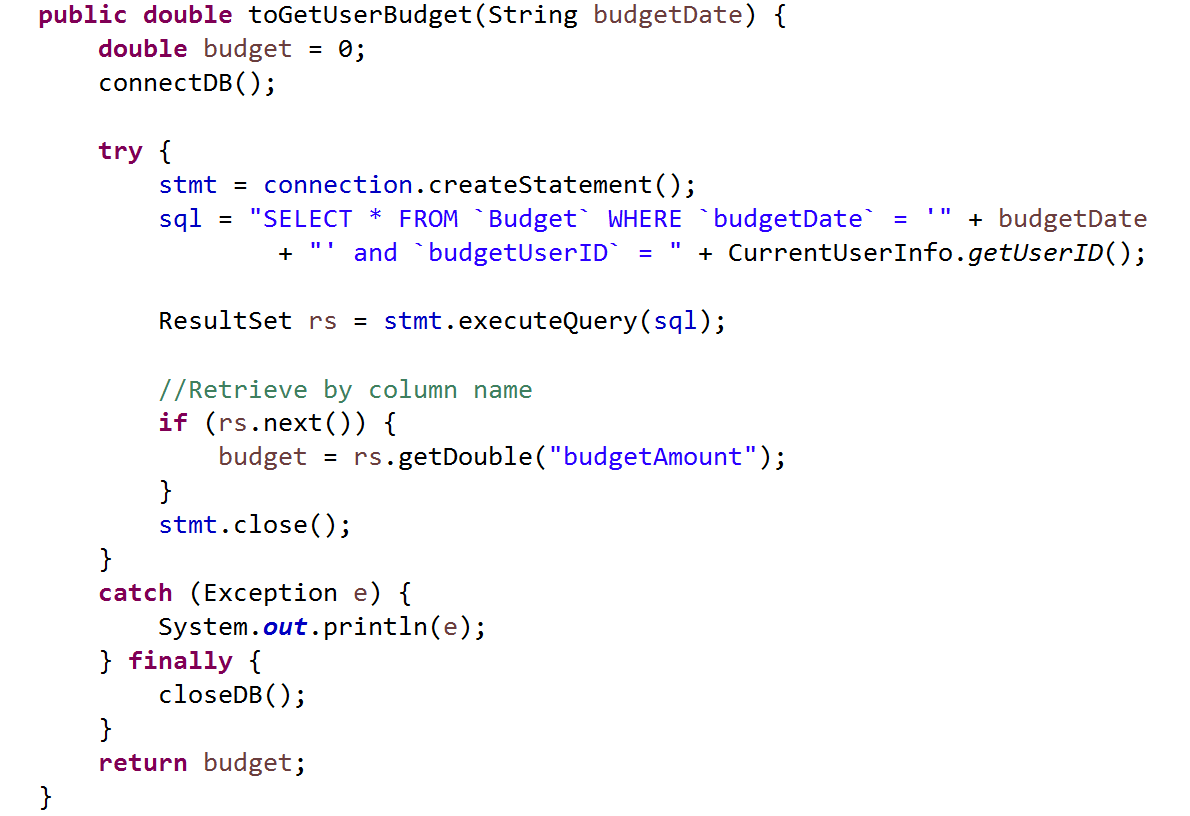
### Queries from the Table



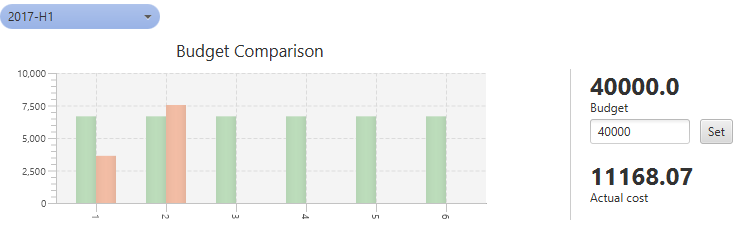
This is the method in the system that is used whenever the budget is set in the “Main Page.” The “REPLACE” keyword is used to replace all the budget details in the database with the once that are specified by the managers each time they change it in the “Main Page.”



The budget is now £40,000, however, once it will set to £20,000, the query above will replace the data and it will mainly show the graph based on £20,000 instead of £40,000.



This method is used to show how much a user’s/manager’s team(s) have spent in a year. The year is divided into first half and second half. So, this query allows the system to find the budget for a specific half (1st or 2nd half) of the year which is specified by the user in the system.



This is where this method is used. As it can be seen from the screenshot that the budget is divided into six months and it compares the actual expense of each month with the budget.

# Bibliography-

**Third-party libraries that were used in the project:**

* MySQL connector, licensed under the GPL license.

Access from: <https://dev.mysql.com/downloads/connector/j/>

* Apache POI, licensed under the Apache License, Version 2.0.

Access from: <https://poi.apache.org>

* Apache PDFBox, licensed under the Apache License, Version 2.0.

Access from: <https://pdfbox.apache.org>

* JUnit, licensed under the Eclipse Public License 1.0.

Access from: <http://junit.org/junit4/>

**Git link:**

<https://projects.cs.nott.ac.uk/psyzy5/G52GRP_TEAM19_2016_Beliebers>

**Google Drives link:**

<https://drive.google.com/drive/folders/0B09zzUIA23jnM05qREMwX0dQaWs?usp=sharing>

**Trello link:**

Backend Development- <https://trello.com/b/6nidMYtv/backend>

Frontend Development- <https://trello.com/b/gDCIeEvn/design>

Test- <https://trello.com/b/nelg3eGp/test>

Meetings- <https://trello.com/b/Bvt1NQ5E/meeting>

Documentations- <https://trello.com/b/tjUe2eIq/documentation>

**END**