Reminder Project / Test Cases

**Use case Register**

**Primary actor:** Student/User

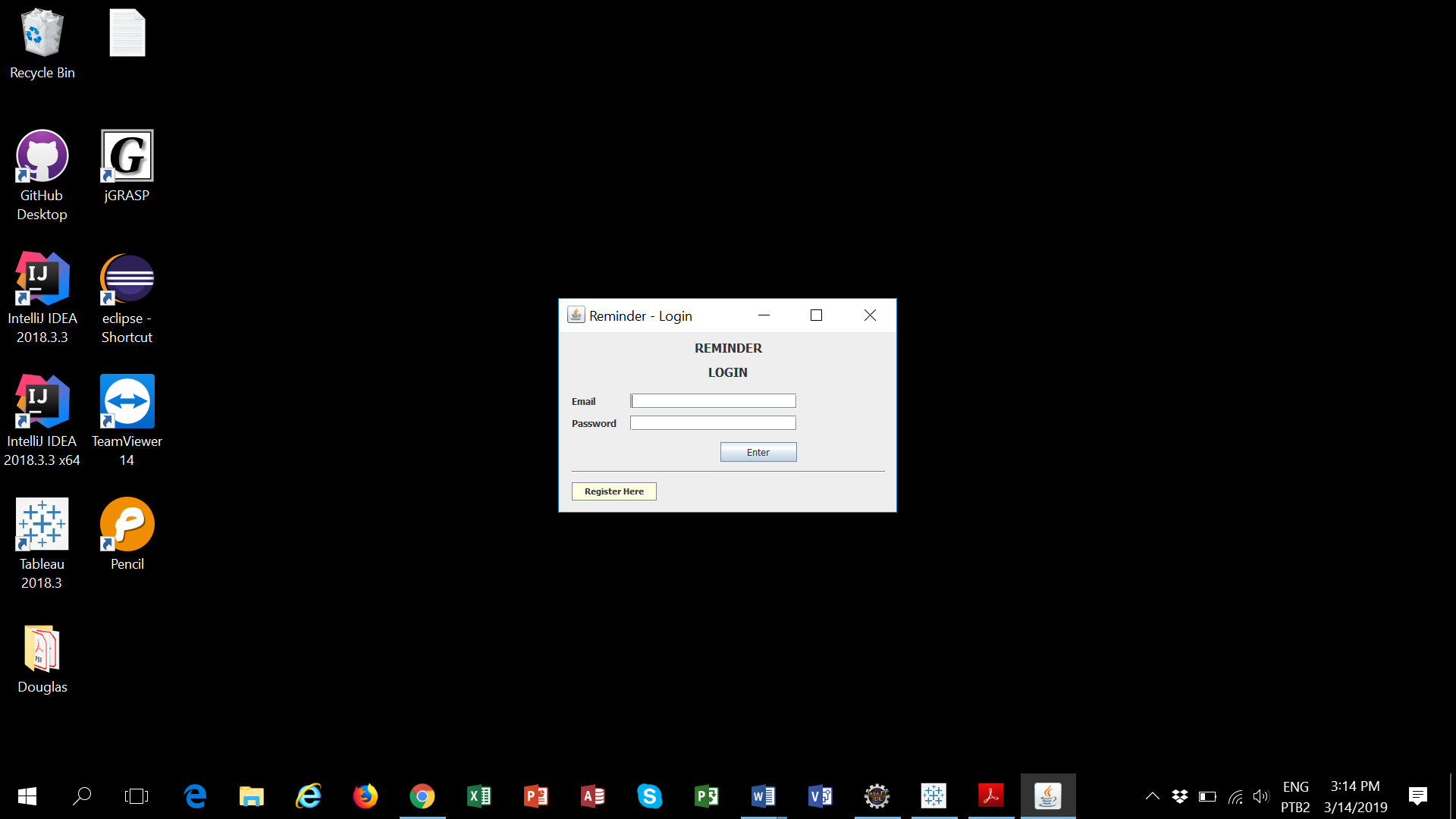
**The goal in context:** The student/user is able to register in the reminder application

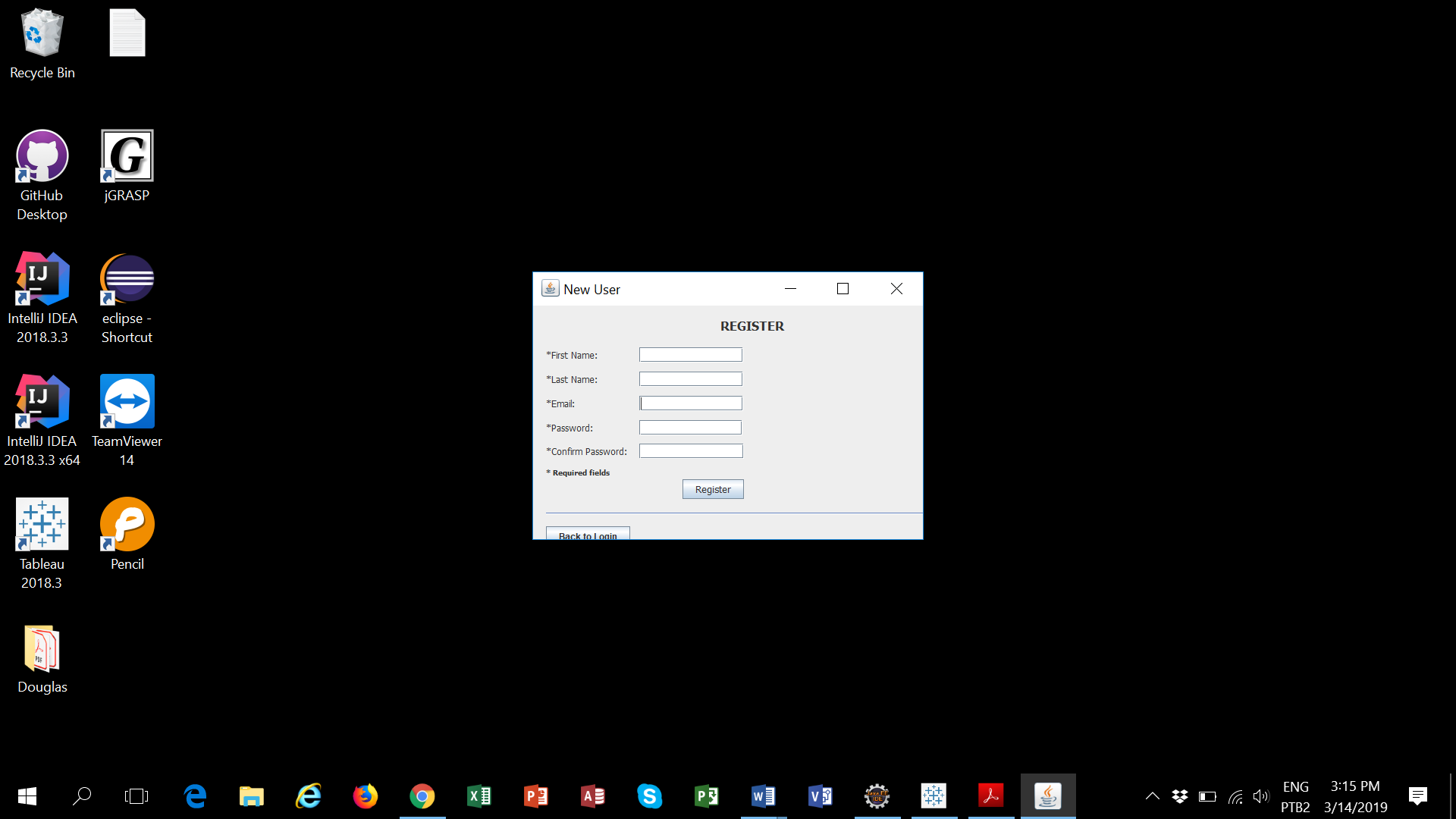
**Preconditions:** The architecture of the system must be available such as basic interfaces, database.

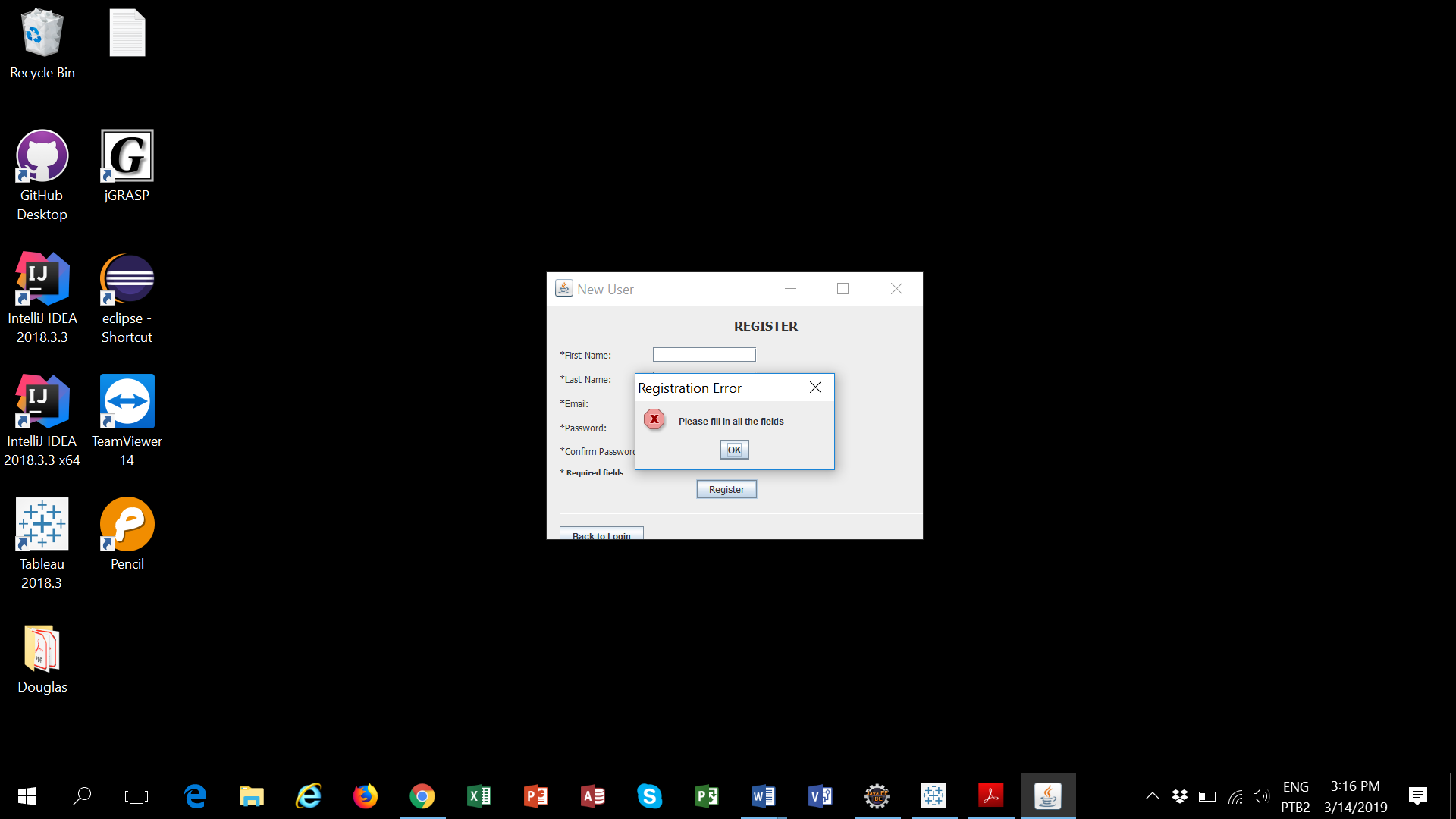
**Scenario:**

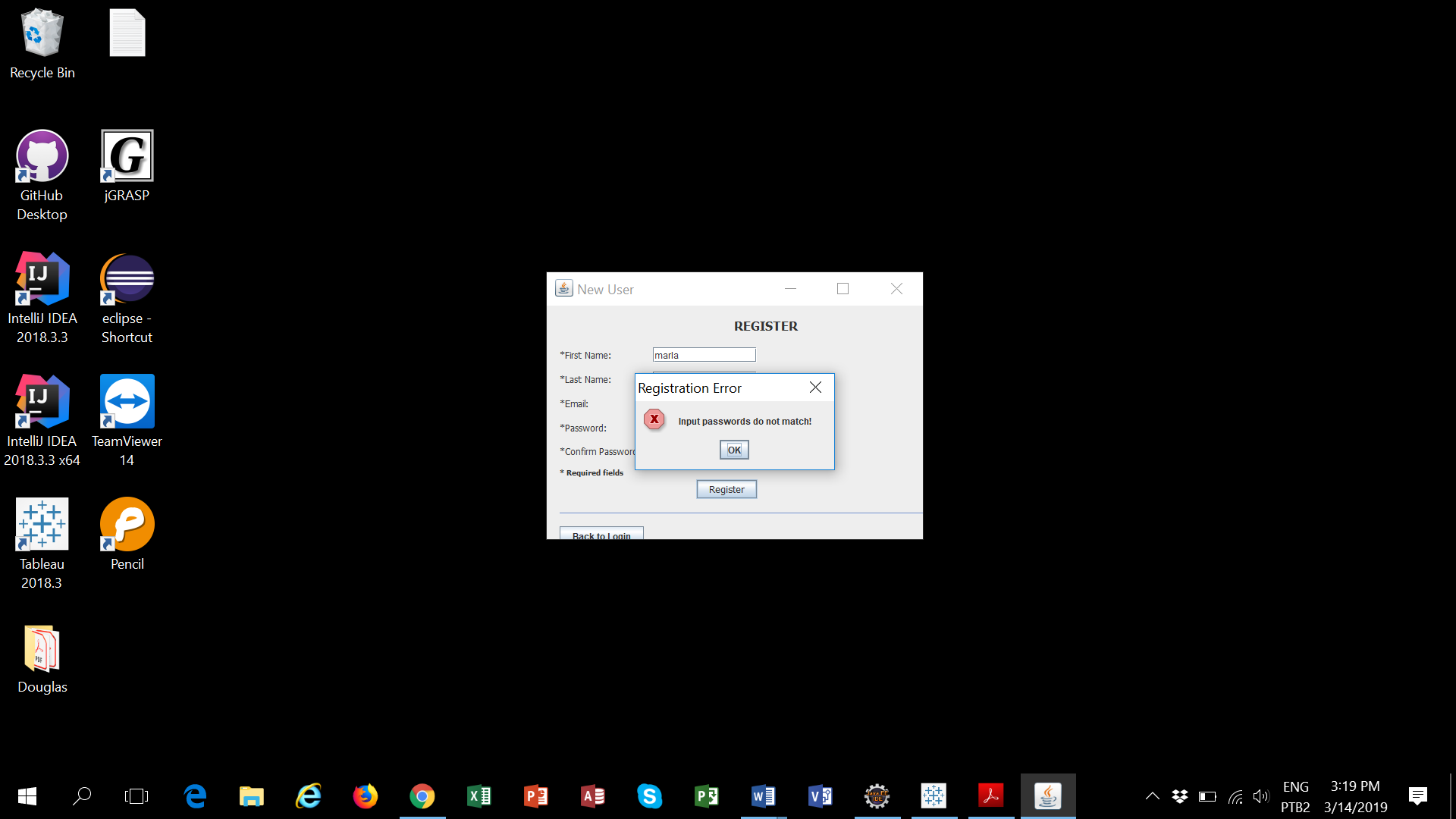
1. the student/user accesses the reminder application on the internet.
2. the student/user enters his first name, last name, email, password and confirm password and clicks the register button.
3. the application requires the first name, last name, email, and password to be filled if any of the fields are not filled the application displays an error message.
4. The application validate the email address into the database if the email address already exists the application displays an error message.
5. the application validates the password policy (it requires eight characters in length, and at least one number). if the password does not follow the policy the application displays an error message.
6. the application validates password and confirm password if they do not match the application displays an error message.
7. The application automatically adds the user-type during the registration process.
8. the application complete the registration process after doing all the validations and then allows the Student logs onto the application.

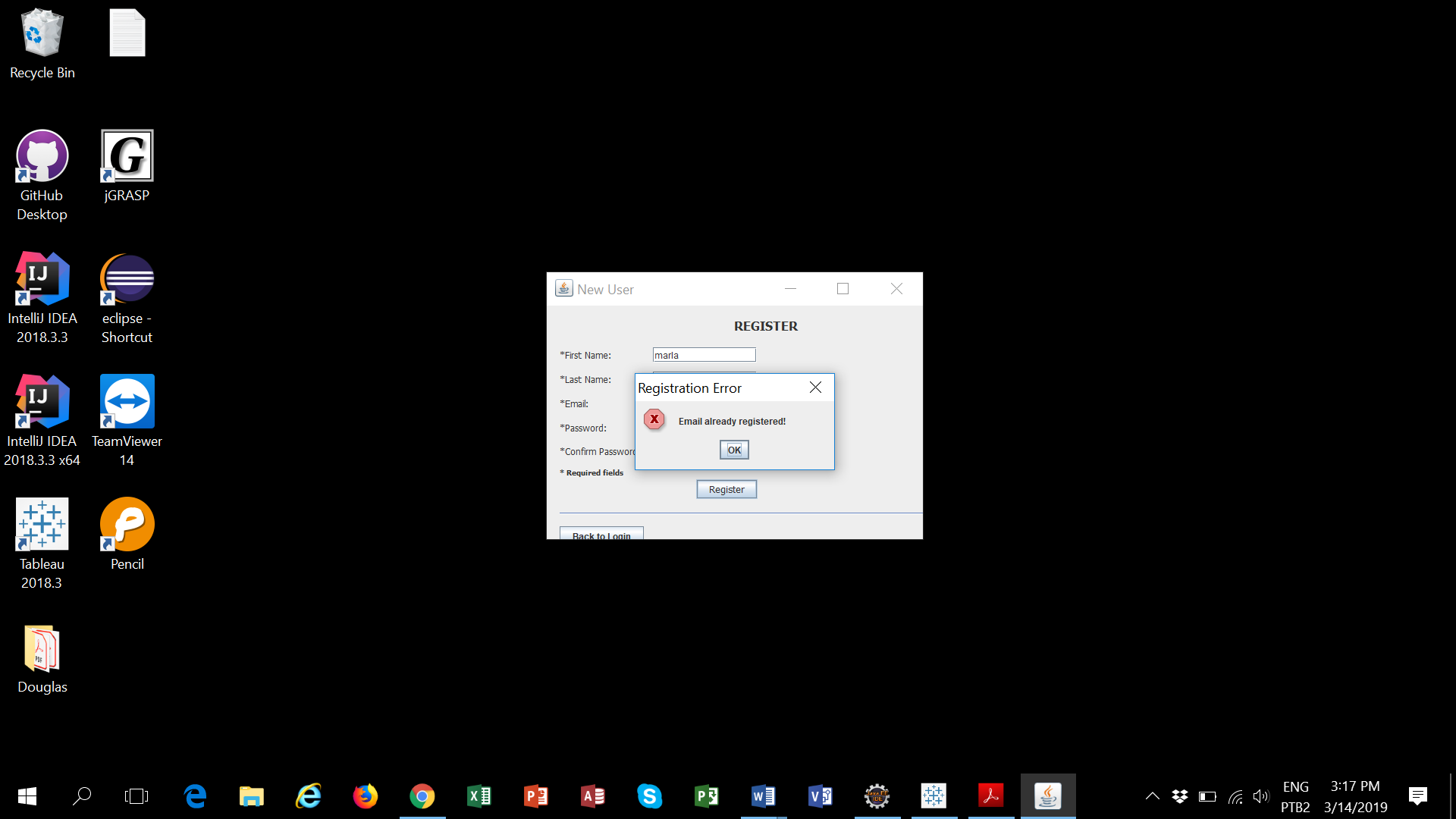
**Results:**











**Use case Authenticate/Login**

**Primary actor:** Student/User

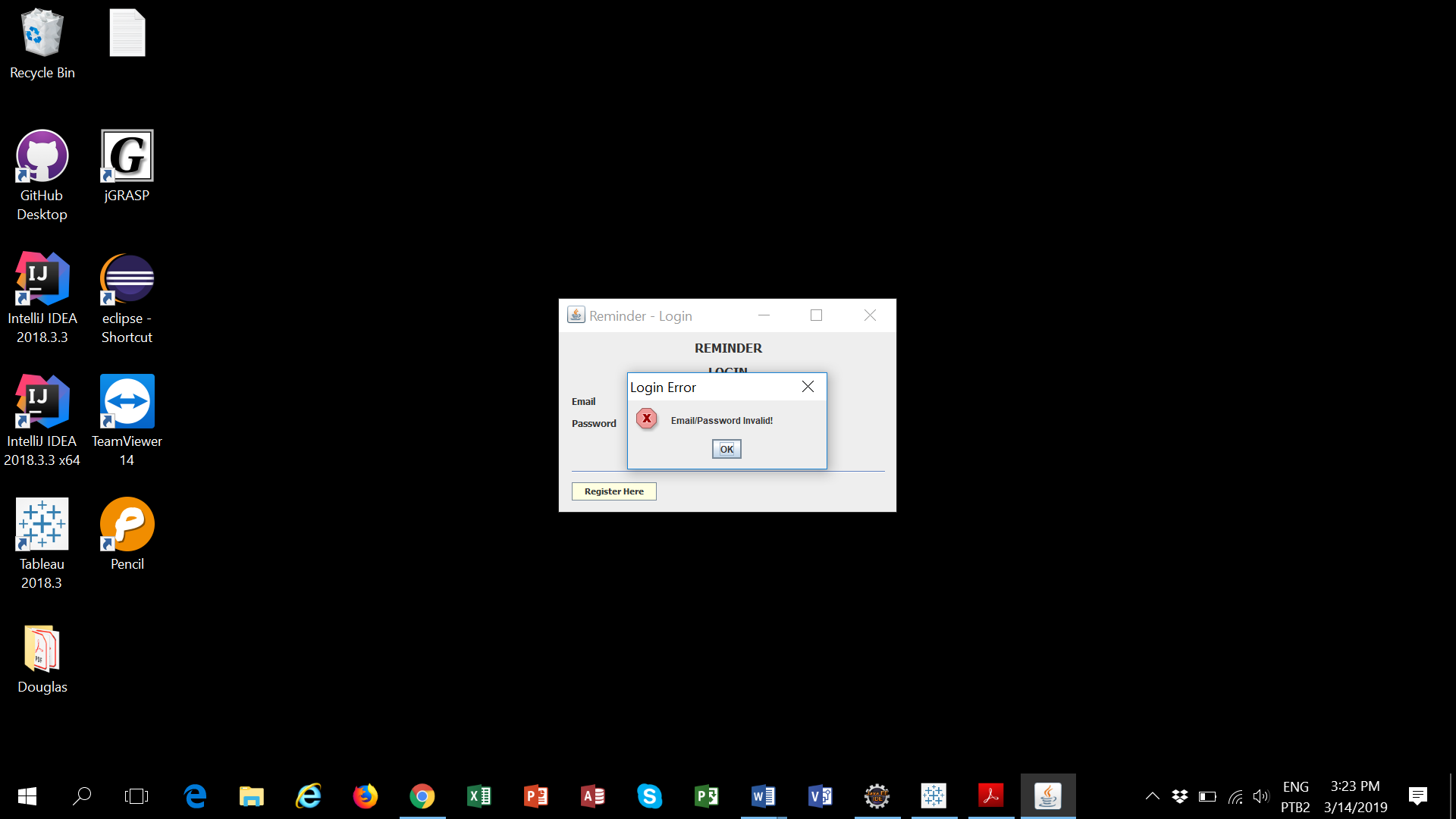
**The goal in context:** The student/user is able to logs onto the reminder application

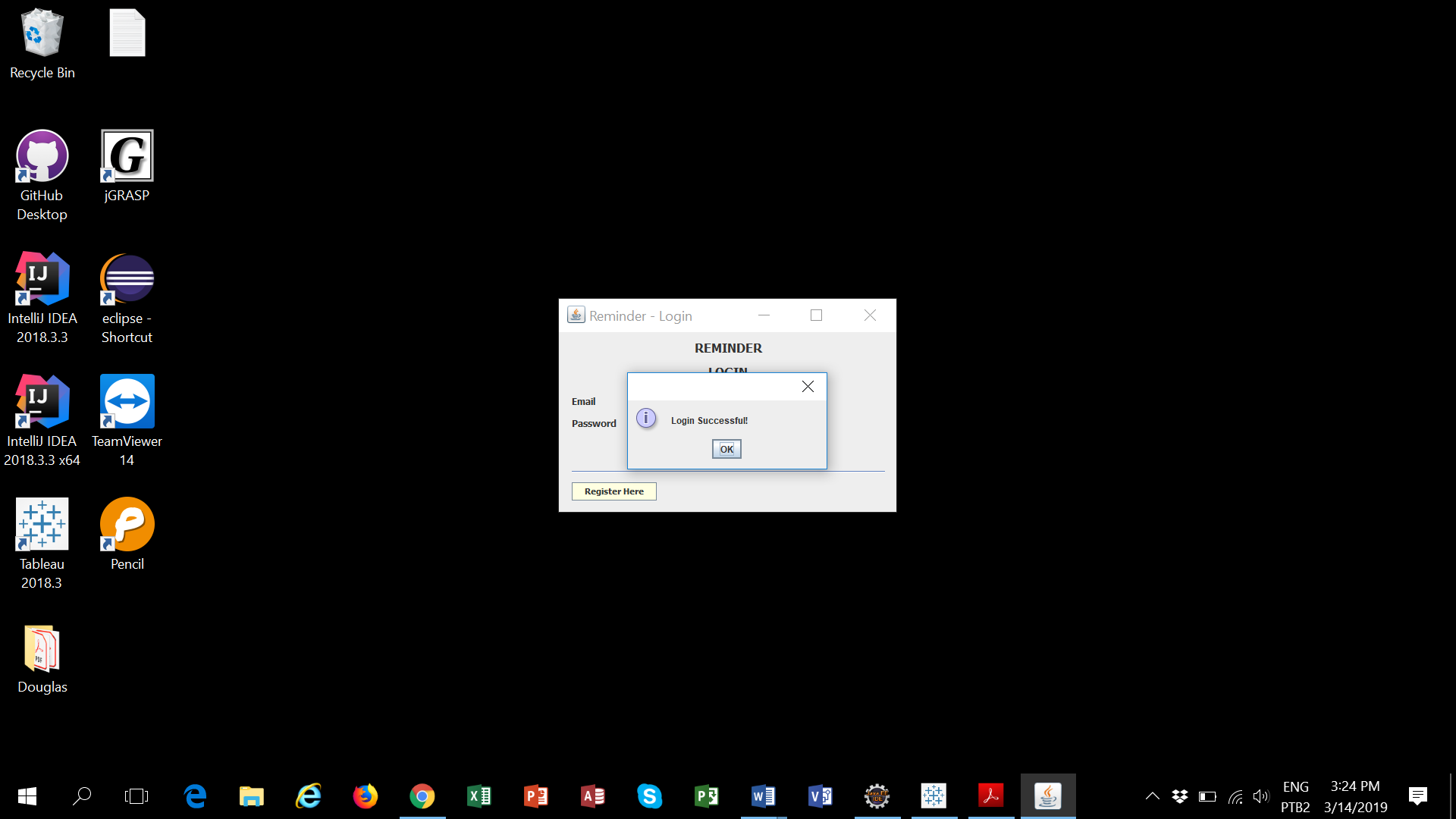
**Preconditions:** The register component must be available.

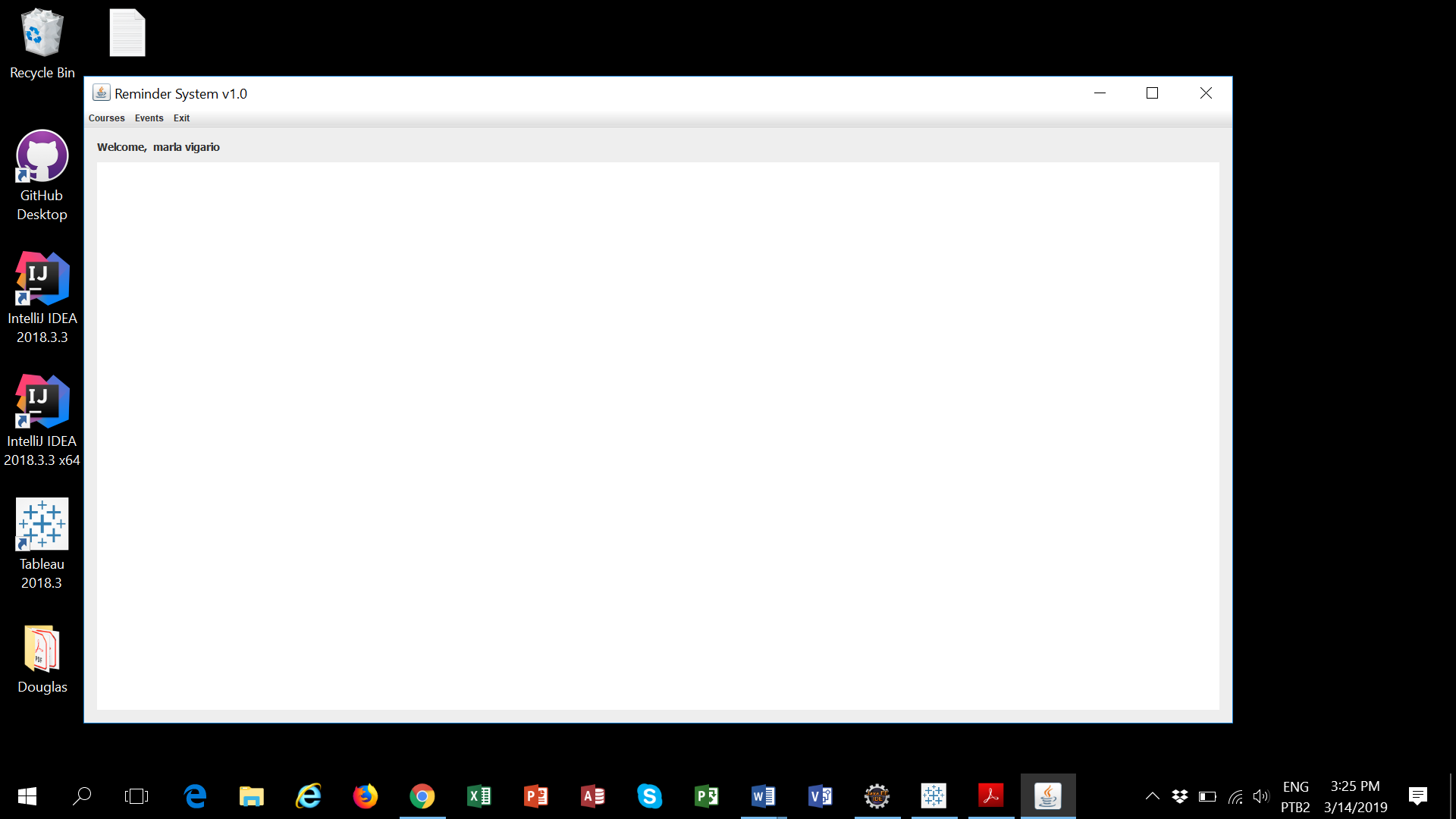
**Scenario:**

1. the student/user enters his email and password and clicks the login button.
2. the application requires email and password to be filled if any of the fields are not filled the application displays an error message.
3. The application validates the email address into the database if the email does not exists the application displays an error message.
4. the application validates the password into the database if the password does not match the application displays an error message.
5. the application complete the authentication process after doing all the validations and then allows the Student enters into the application.

**Results:**







**Use case Course**

**Primary actor:** Student/User

**The goal in context:** The student/user is able to Add, Edit or Delete the course information in the reminder application

**Preconditions:** The architecture of the system must be available such as basic interfaces, database. The User component must be available

**Scenario:**

1. the student/user accesses the reminder application on the internet.
2. the student/user executes all the steps to Authenticate/Login into the reminder application.
3. the application shows automatically the Username that is logged in
4. the student/user chooses the option hovering the mouse over the option wanted and then click on it.
5. the student/user accesses the primary menu and can choose the Course option, Event option or choose the Exit option to leave the application.
6. the student/user chooses the course option and then accesses the second menu which can choose the List Course option or Add course option.

**List Course option:**

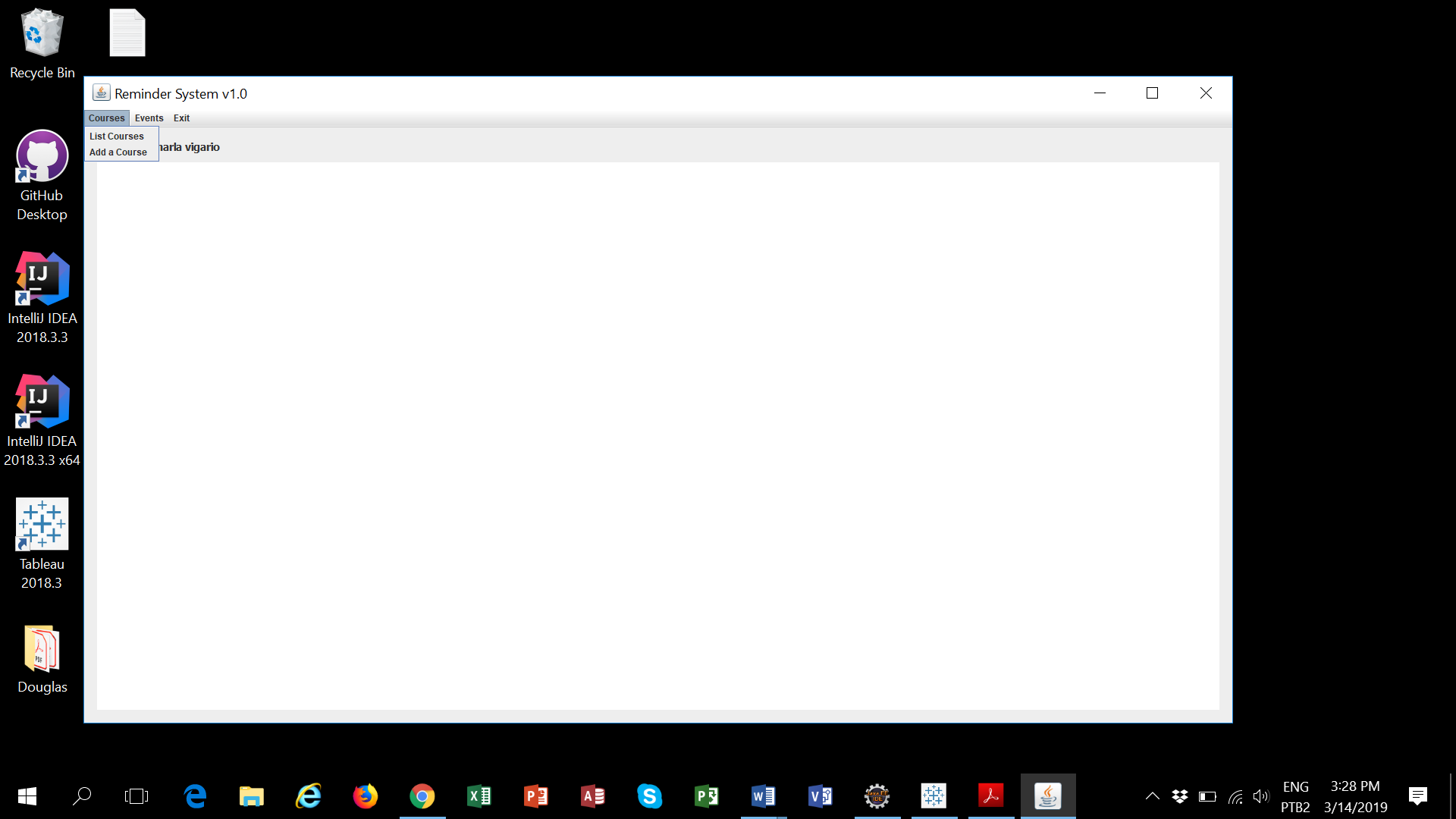
1. the application displays a table showing all course information that exists recorded in the system owned by the user logged in. If the user double click in one row the application allows edit or delete the course information.  
   It is also available in this table the option to Add new course information.
2. the application displays an error message if the user does not choose one row to edit or delete the course information.
3. the student/user selects the course information to be deleted and then clicks on the delete button.
4. the application requires a confirmation to delete the course information. It will be displayed a combo-box to the user confirms.

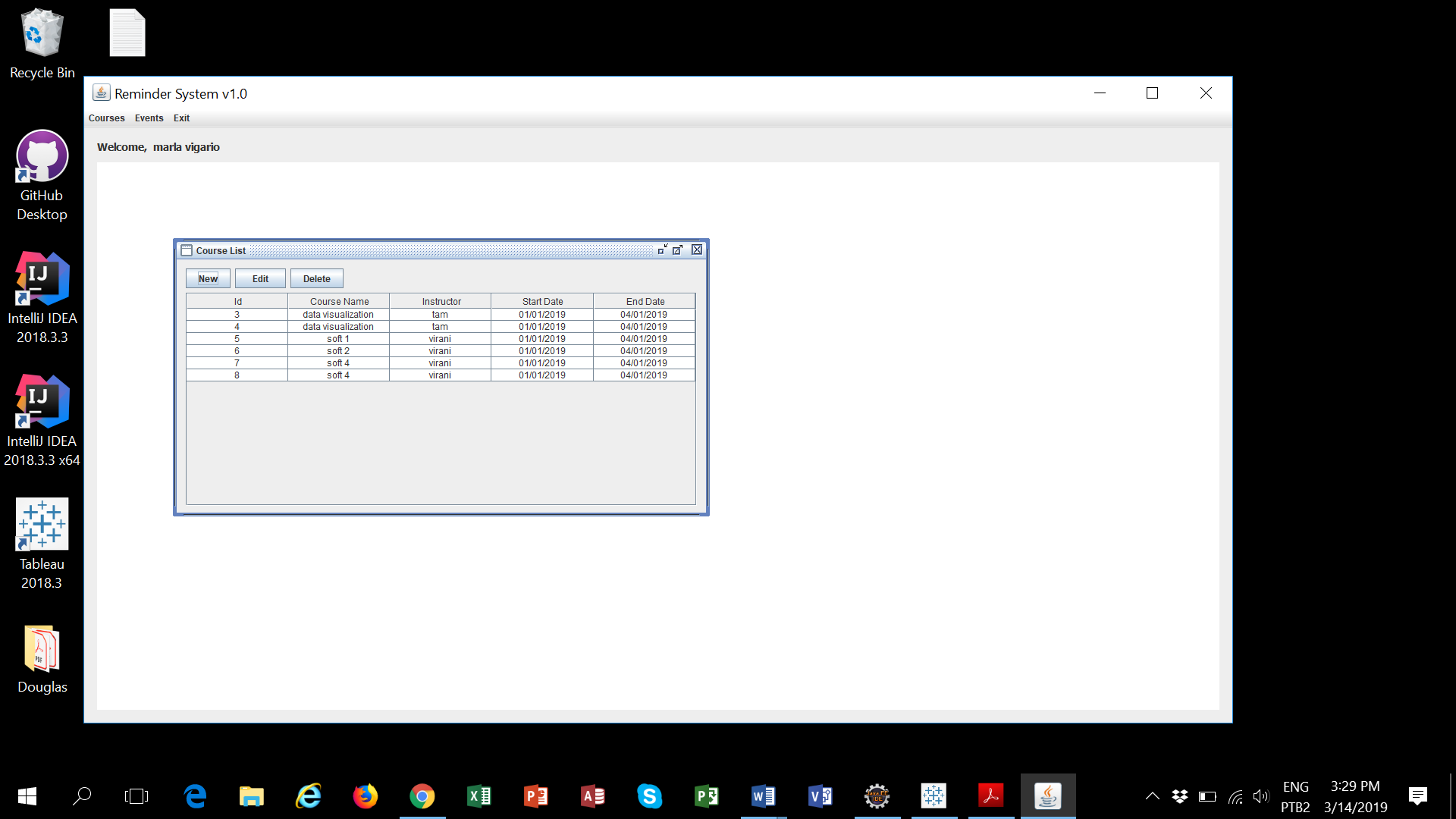
**Add course option:**

1. the application requires the Course Name field to be filled and the application displays an error message if the course name field is blank.
2. the application optionally allows informing the course instructor name.
3. the application optionally allows informing the course start and end date.
4. the application validates the start date and end date if the date is not valid the application displays an error message.
5. the student/user clicks on Add button to save the new course information.
6. The application creates automatically one CourseID for the new course information.

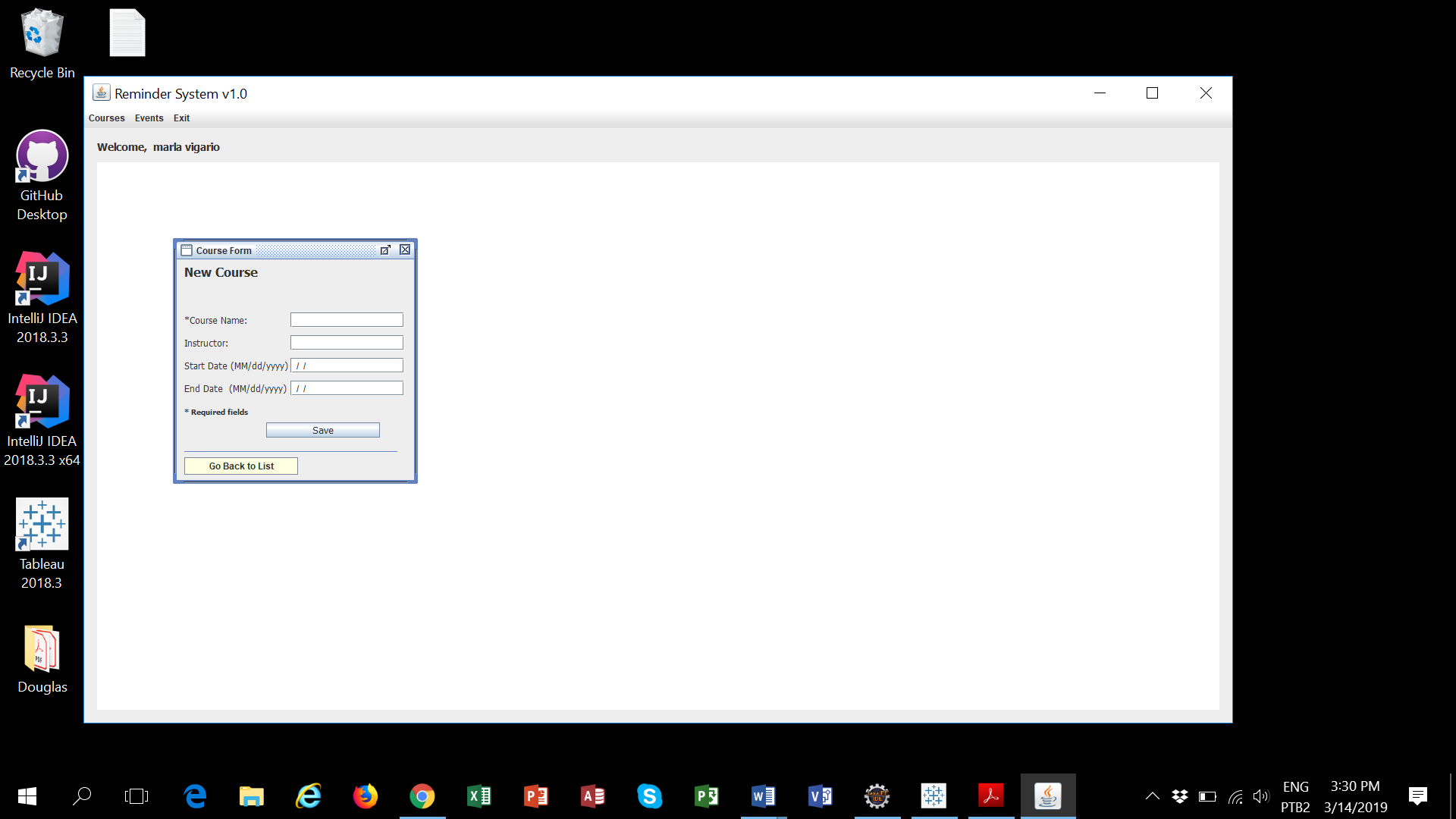
**Results:**

**List Course option:**

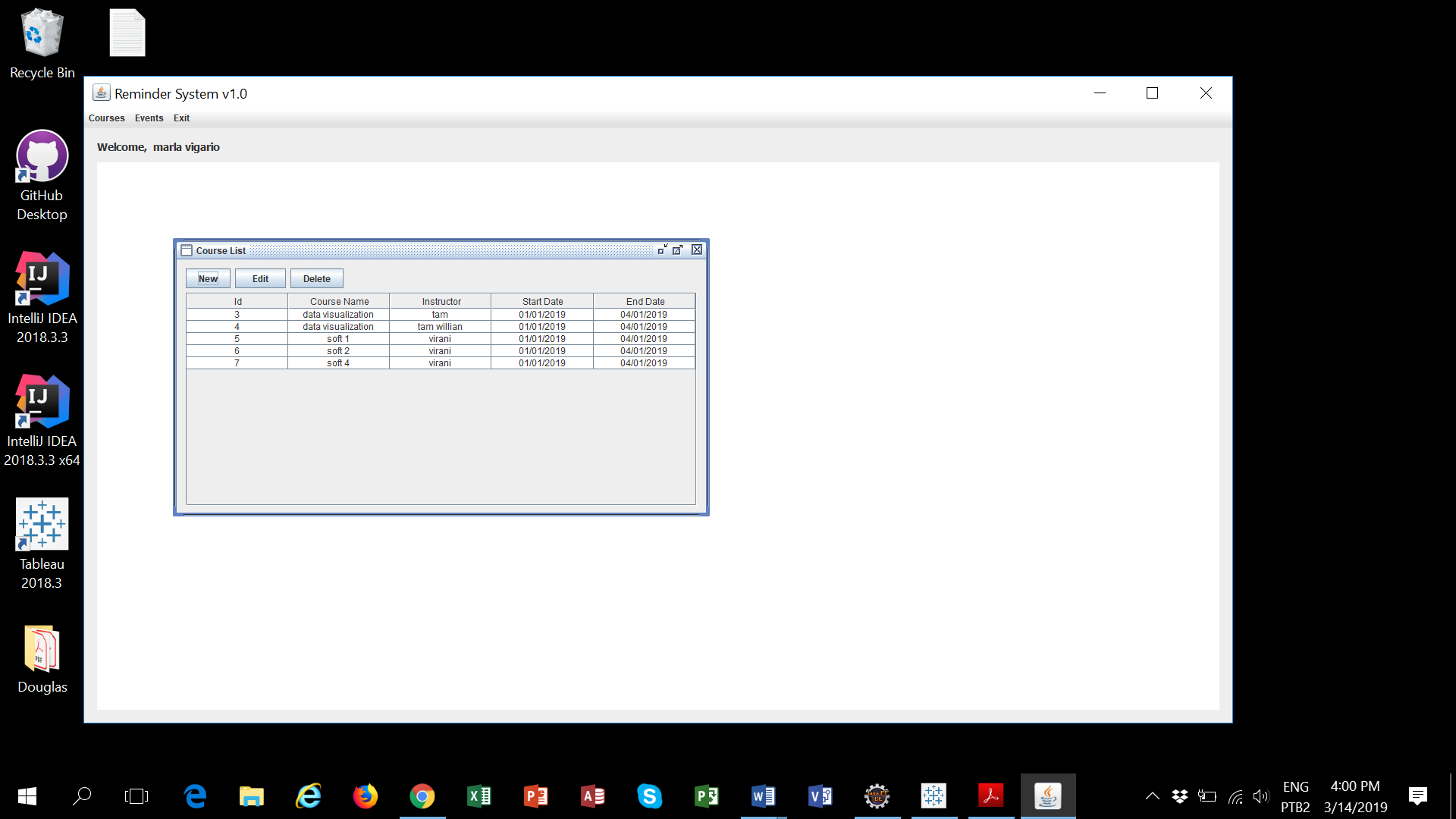


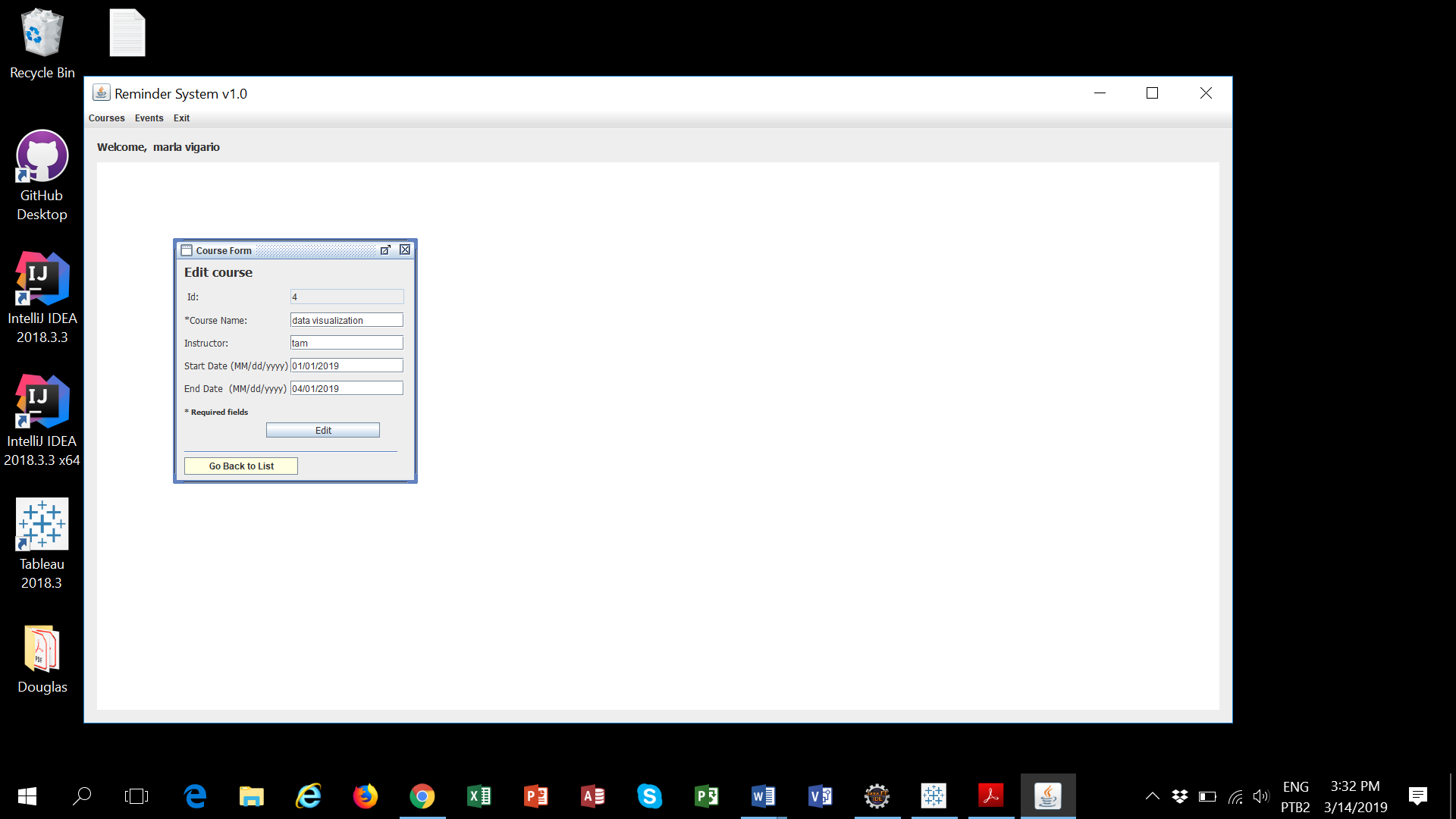


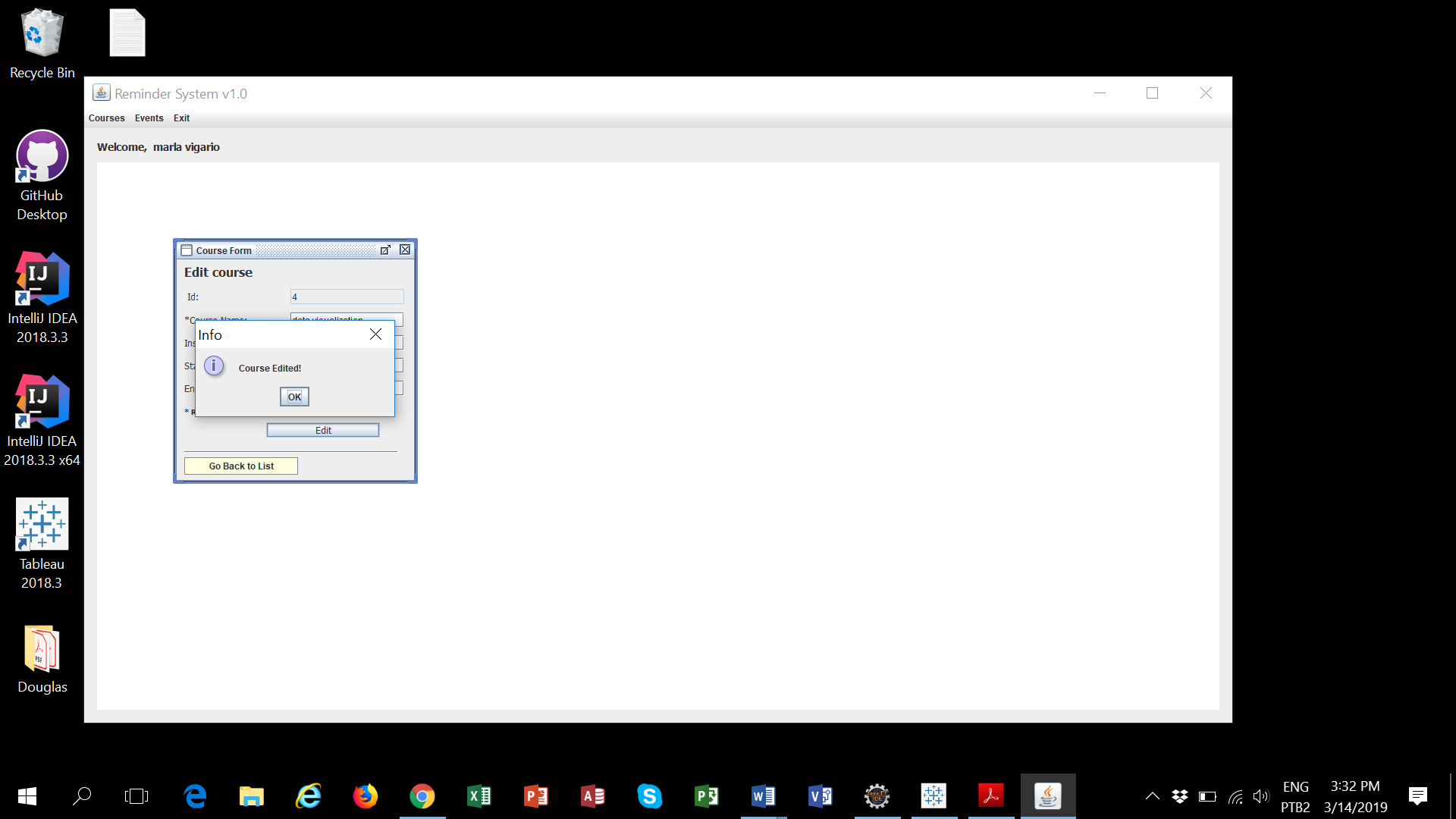
New Option:



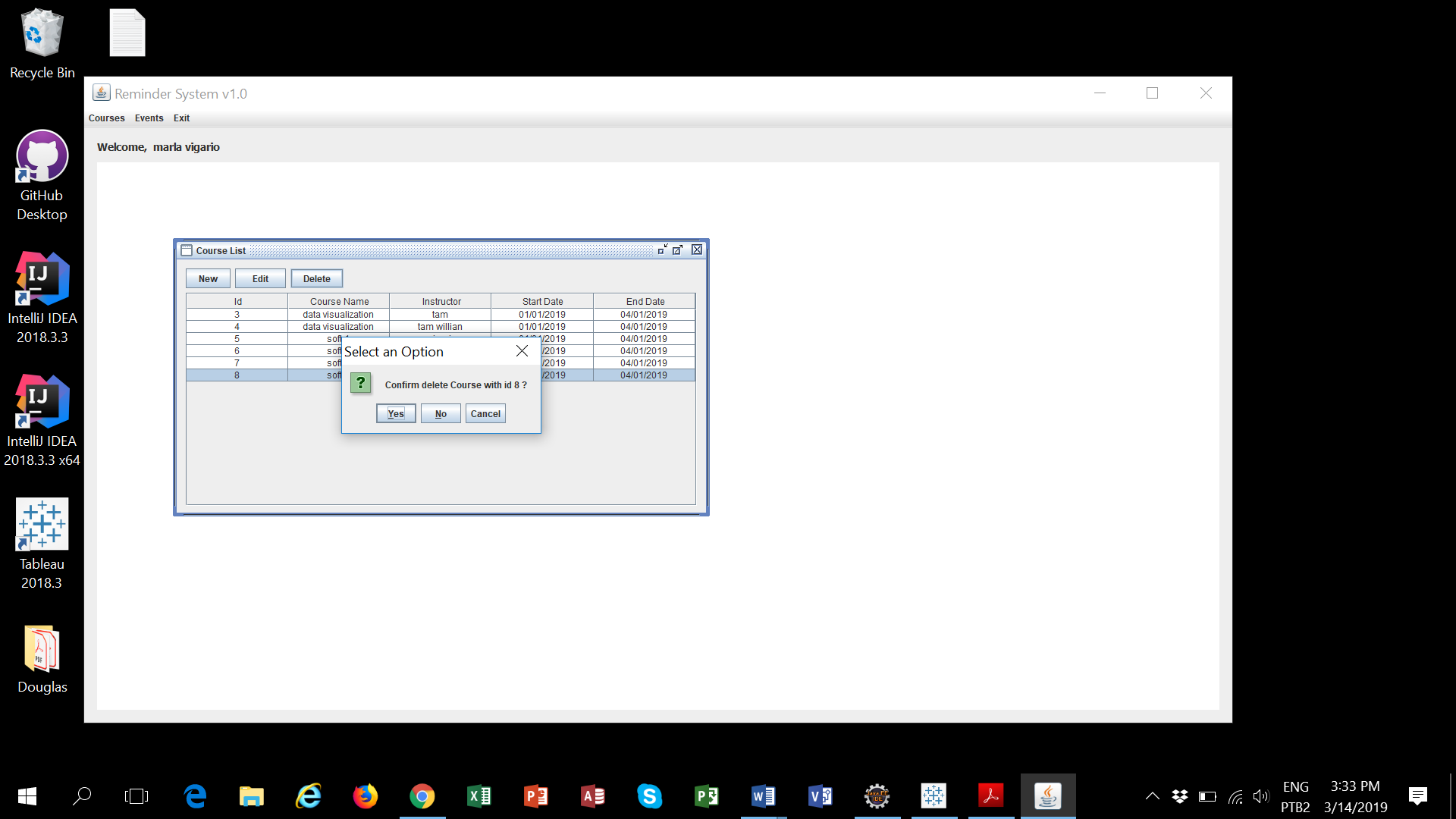
Edit Option:



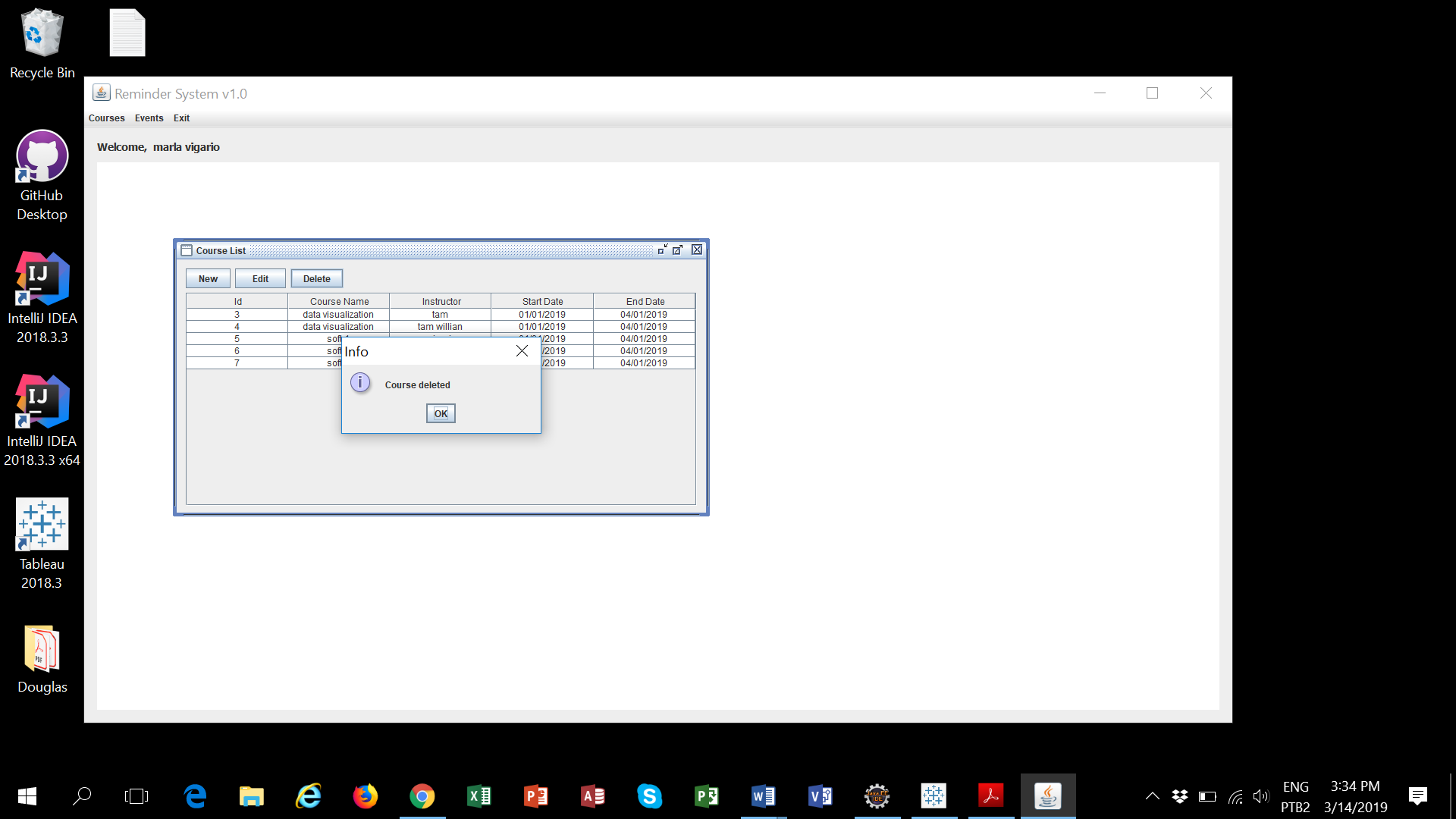


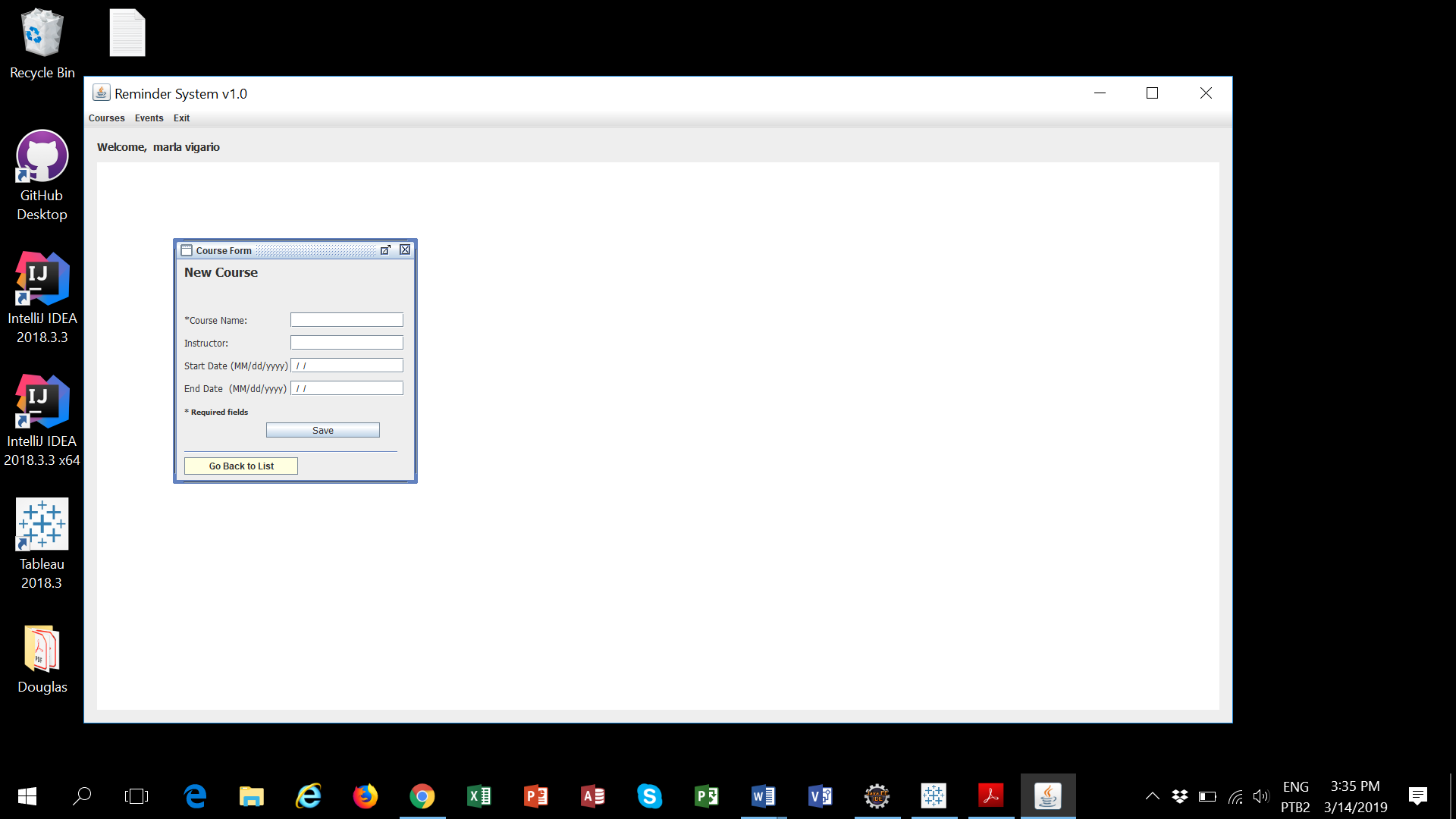


Delete Option:



**Add course option:**





**Use case Event**

**Primary actor:** Student/User

**The goal in context:** The student/user is able to Add, Edit or Delete the event information in the reminder application

**Preconditions:** The architecture of the system must be available such as basic interfaces, database. The User and course components must be available

**Scenario:**

1. the student/user accesses the reminder application on the internet.
2. the student/user executes all the steps to Authenticate/Login into the reminder application.
3. the application shows automatically the Username that is logged in.
4. the student/user chooses the option hovering the mouse over the option wanted and click on it.
5. the student/user accesses the primary menu and then can choose the Course option, Event option or choose the Exit option to leave the application.
6. the student/user chooses the event option and then accesses the second menu which can choose the List event option or Add event option.

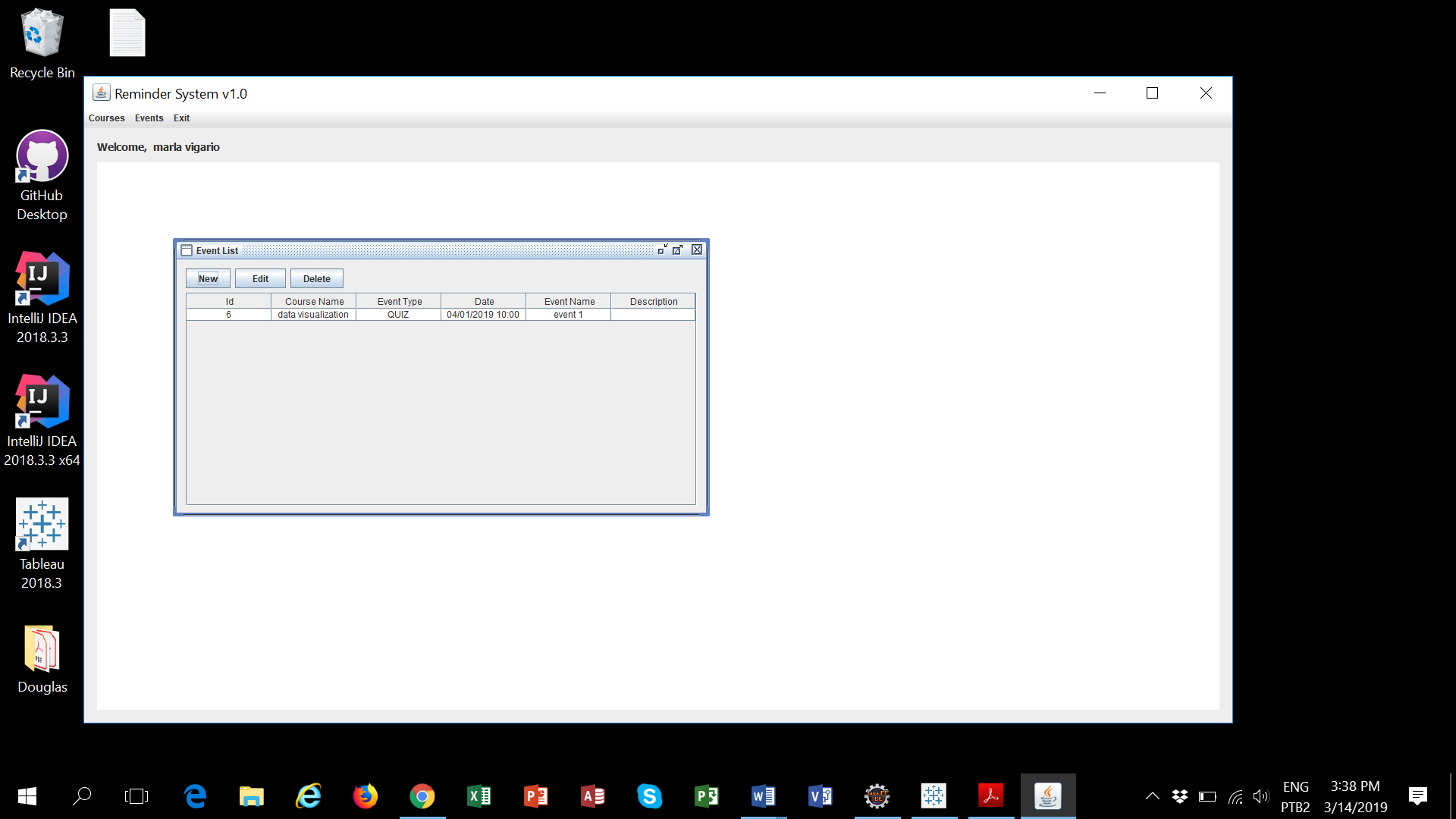
**List event option:**

1. the application displays a table showing all event information associated with course that exists recorded in the system owned by the user logged in. If the user double click in one row the application allows edit or delete the event information. It is also available in this table the option to Add new event information.
2. the application displays a error message if the user does not choose one row to edit or delete the event information.
3. the student/user selects the event information to be deleted and then clicks on the delete button.
4. the application requires a confirmation to delete the event information. It will be displayed a combo-box to the user.

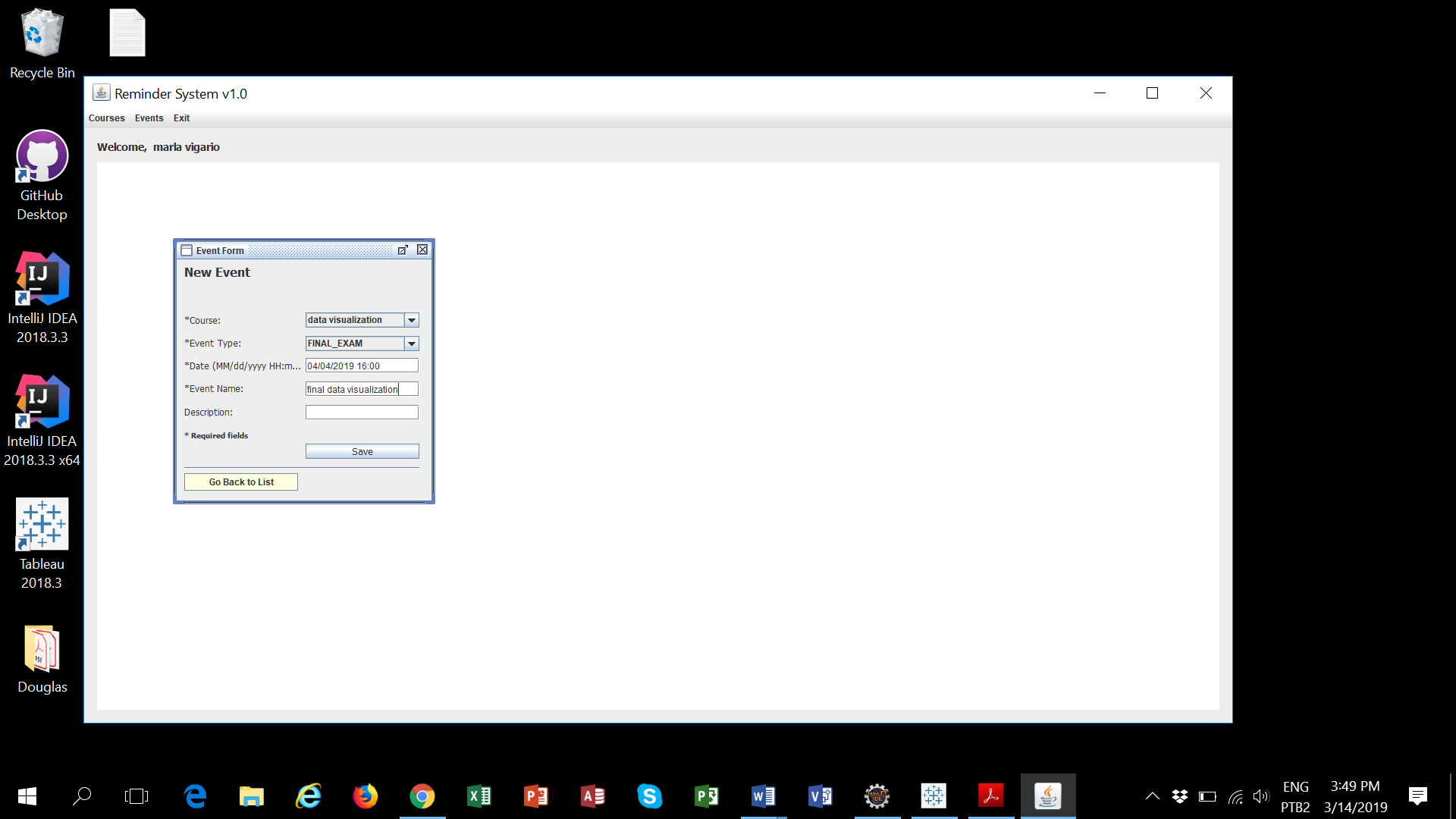
**Add event option:**

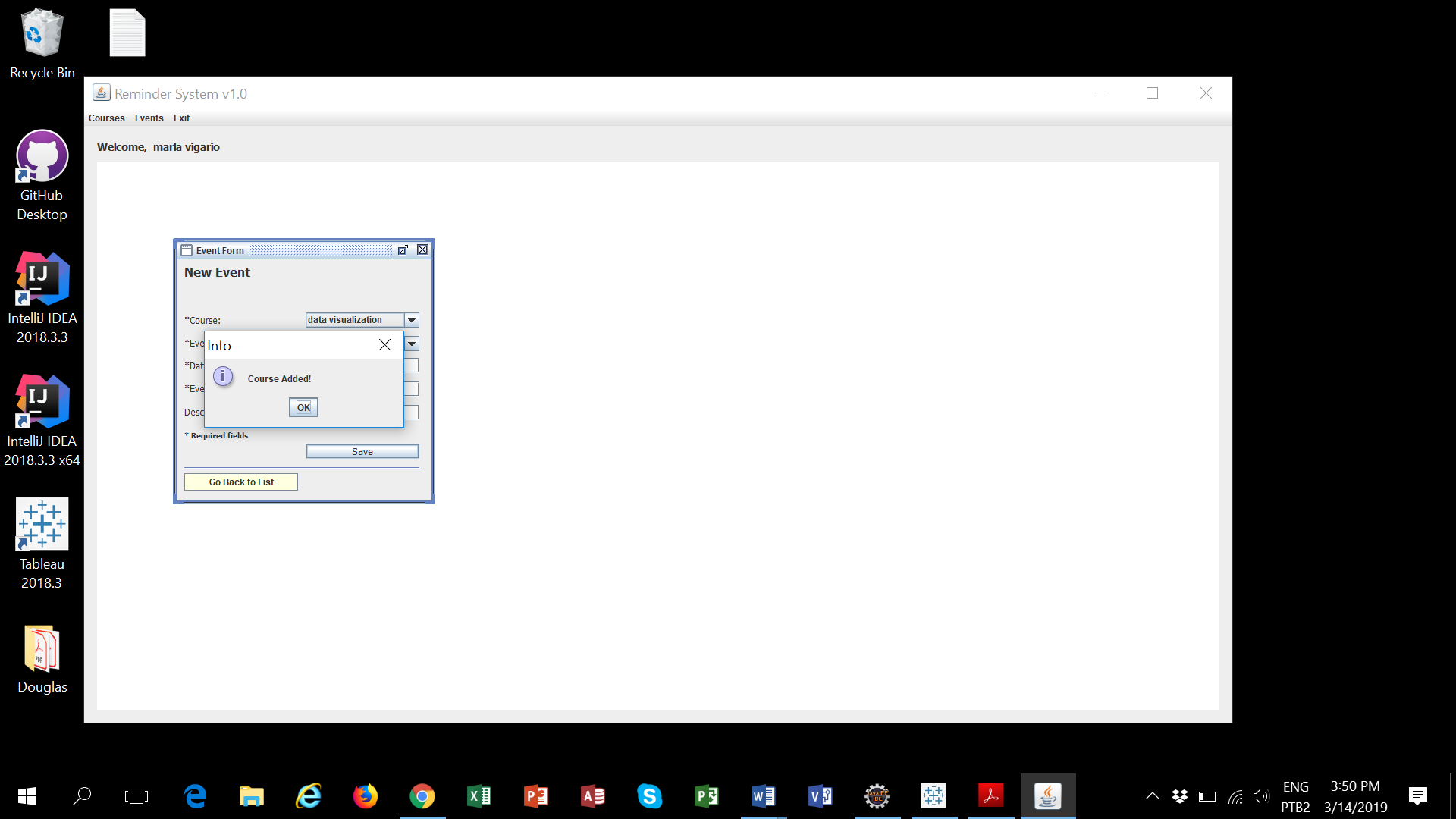
1. the application requires the event name, event type, date, and time fields to be filled and the application displays an error message if one of the fields are blank.
2. the application validates the date if it is not valid the application displays an error message.
3. the application allows the Student/user to choose 4 types of event Assignment, Quiz, midterm exam, or final exam.
4. the student/user clicks on Add button to save the new event information.
5. The application creates automatically one EventID for the new event information associated with a course.

**List event option:**

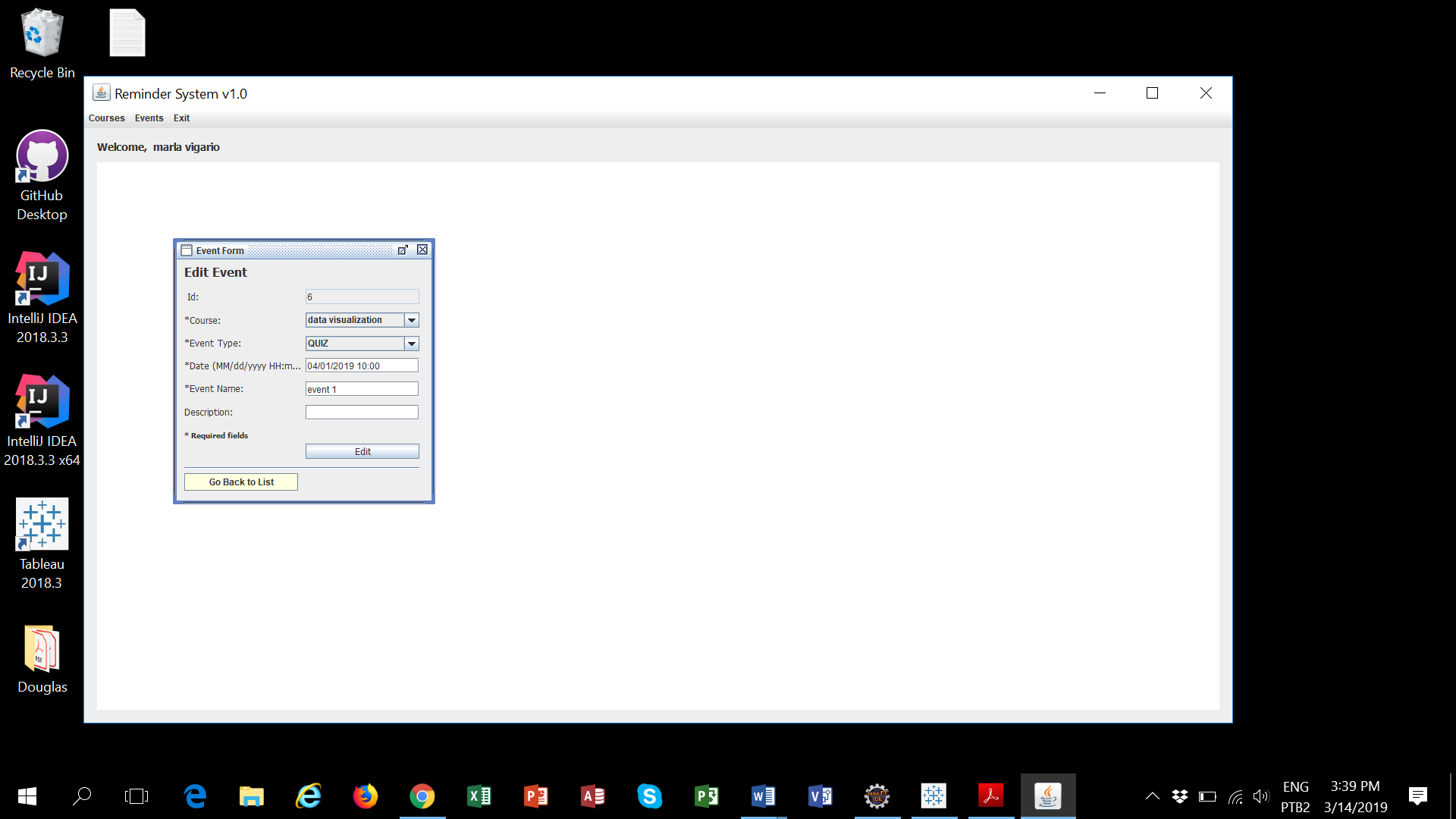


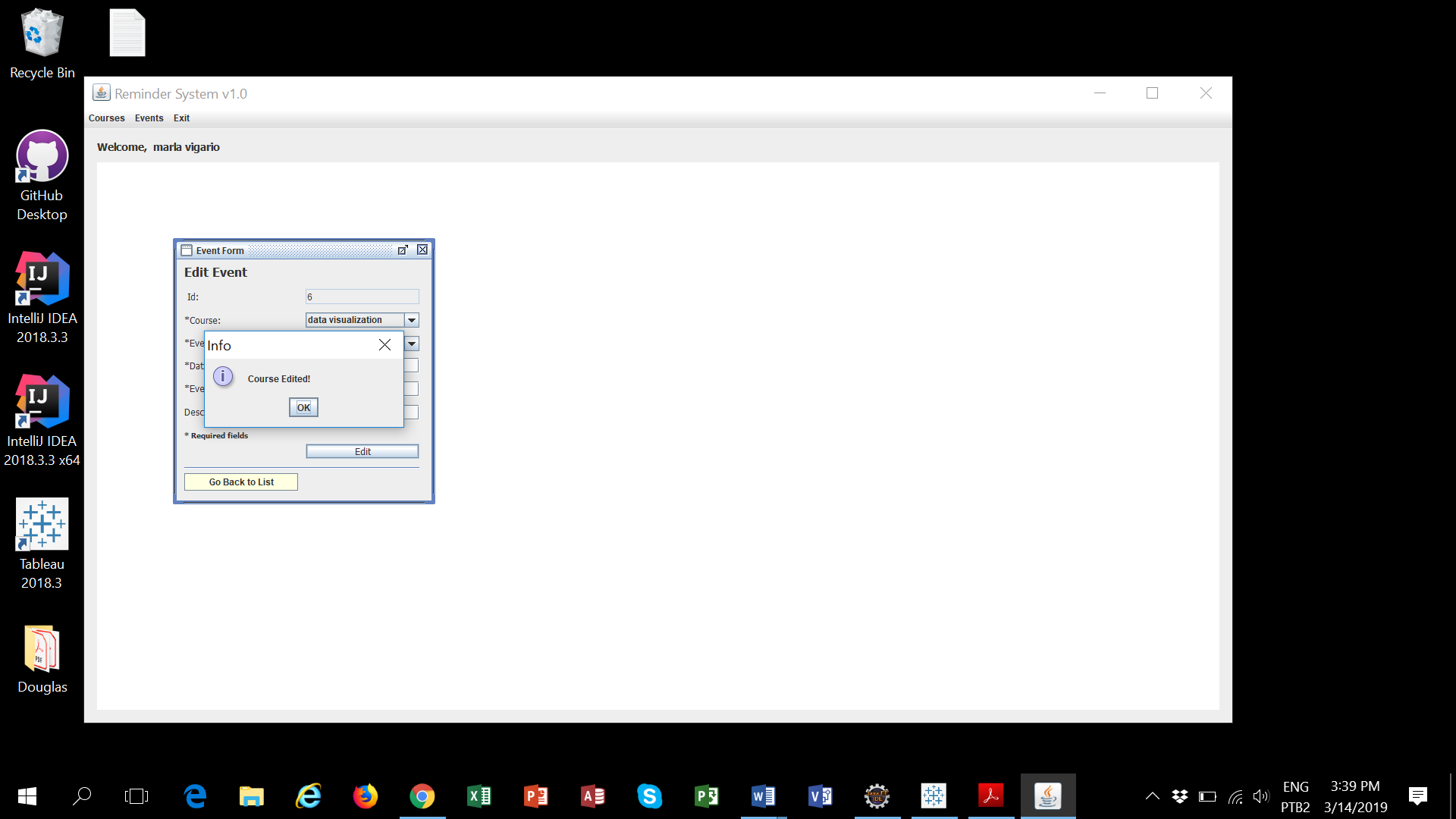
New Option:



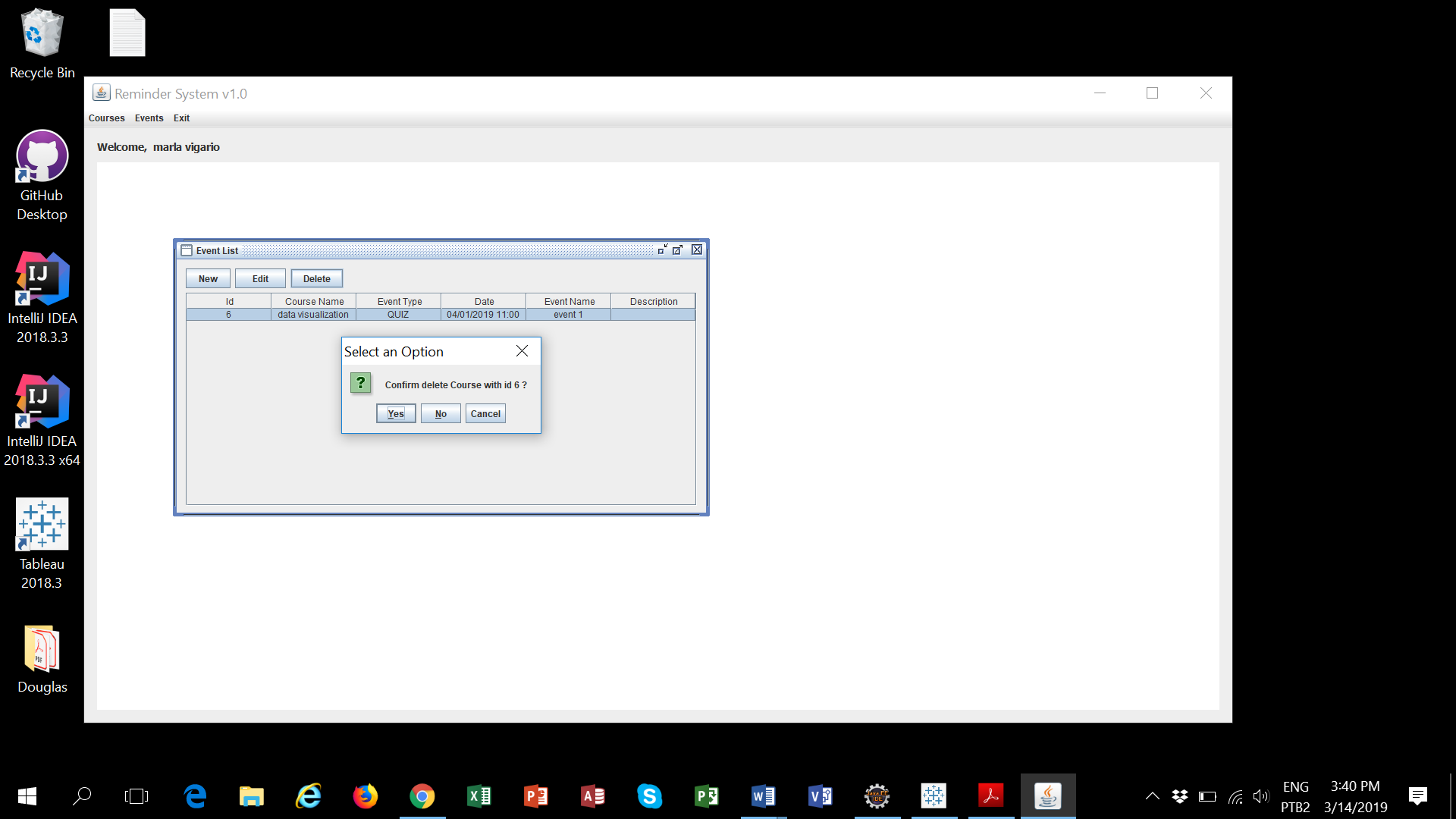


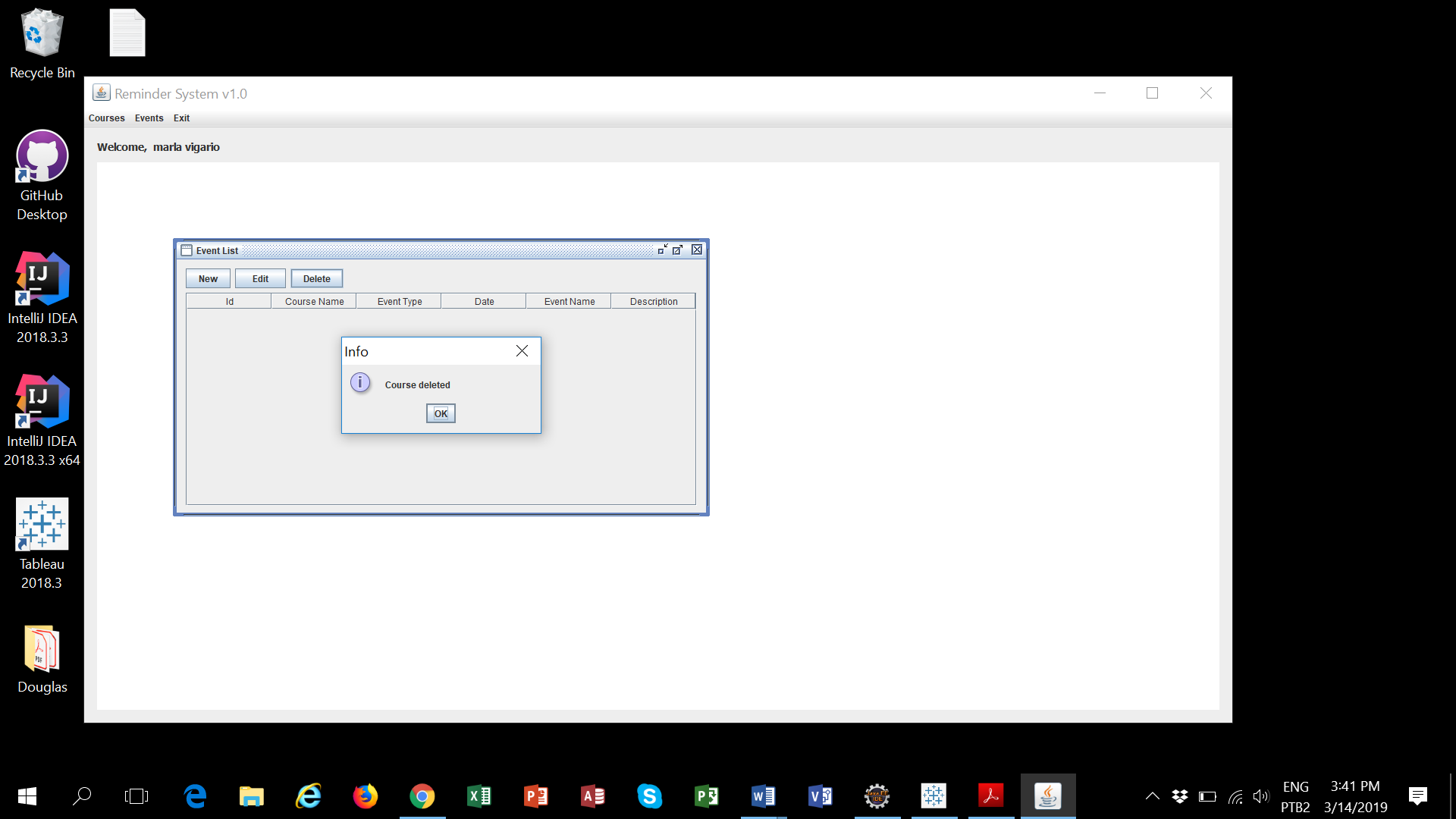
Edit Option:





Delete Option:





**Add event option:**

