**Instrument Management System**

Index

|  |  |  |
| --- | --- | --- |
| Sr. no. | Topic | Page no. |
| 1 | Overview | 1 |
| 2 | XML Schema Definition | 1 |
| 3 | Sequence Diagrams | 2 |
| 4 | Flowchart | 4 |
| 5 | Block Diagram | 5 |
| 6 | Screenshots | 6 |

1. **Overview :**

Instrument Management System is an application which can be used to store information of instruments such as name of instruments, user of the instrument and projects in which the instrument is being used. The information of these instruments is stored in an XML file which acts as database. You can also add new instruments to the database as well as you can update existing instruments in the database. If you do not want any instrument in the database, you can delete that instrument as well.

1. **XML Schema Definition:**

<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="Instruments">

<xs:complexType>

<xs:sequence>

<xs:element name="Instrument" maxOccurs="unbounded" minOccurs="0">

<xs:complexType>

<xs:sequence>

<xs:element type="xs:string" name="Name"/>

<xs:element type="xs:string" name="User"/>

<xs:element type="xs:string" name="Project"/>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

Above is the XML Schema Definition. This schema describes the structure of our XML file, where we shall store our data. The namespace of http://www.w3.org/2001/XMLSchema is used to validate the XML file.

1. **Sequence Diagrams:**

The following sequence diagram represents the Load Data functionality. When user wants to see all instruments data, user sends Load Data request to main window. Further main window calls the getInstruments method of Reader-Writer class and displays the Instruments data from the database file.

Diagram, box and whisker chart

Description automatically generated

The following sequence diagram represents the Add New Instrument functionality. When user wants to add a new instrument to database file, they pass the new instrument’s data to main window. Further the main window calls addInstrument method of Reader-Writer class and pass a parameter through it. With this method new instrument’s data is then stored in the database file.Chart, diagram, box and whisker chart

Description automatically generated

The following sequence diagram represents Update Instrument functionality. When user wants to update an instrument, they make specific changes in the instrument and sends the updated version of the instrument to main window. Further main window calls updateInstrument method of Reader-Writer class and pass some parameters. This method takes all data sent by main window and makes changes to that specific instrument in the database file.Diagram, box and whisker chart

Description automatically generated

Following sequence diagram represents the Delete Instrument functionality. When user wants to delete an instrument, they should select and delete an instrument on the main window. Further the main window calls deleteInstrument method of Reader-Writer class and passes a parameter through it. This method deletes the instrument from the database file as well.

Diagram

Description automatically generated

1. **Flowchart:**

Following is a flowchart which represents the flow of the application.

Diagram

Description automatically generated

1. **Block Diagram:**

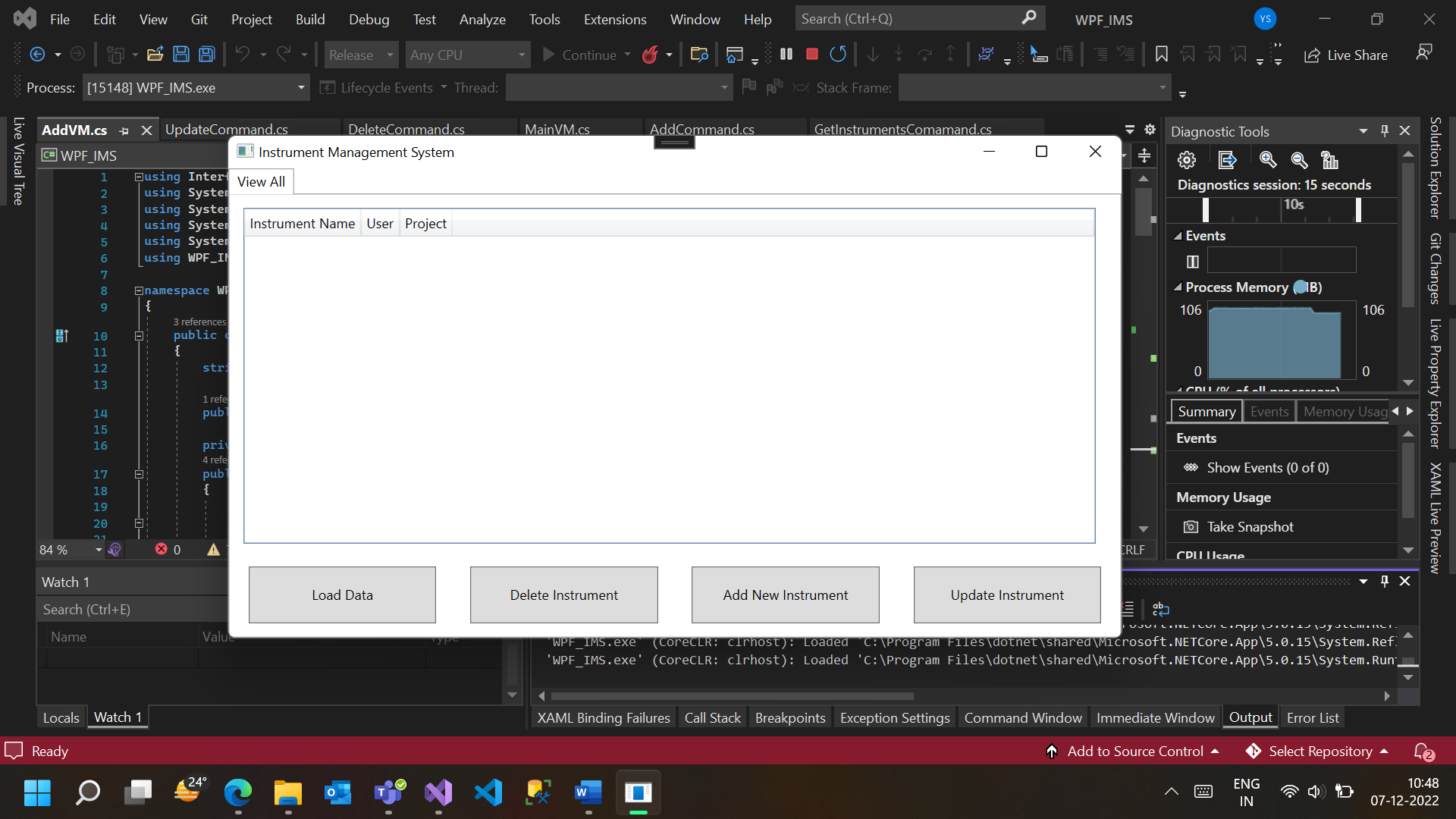
Following is the block diagram for Instrument Management System. There are three components here MainWindow, Reader-Writer library and XML file. MainWindow calls the methods getInstruments, deleteInstrument, addInstrument, updateInstrument from Reader-Writer library via an interface object. As soon as these methods are executed, respective functionalities are performed on the XML file.

Diagram

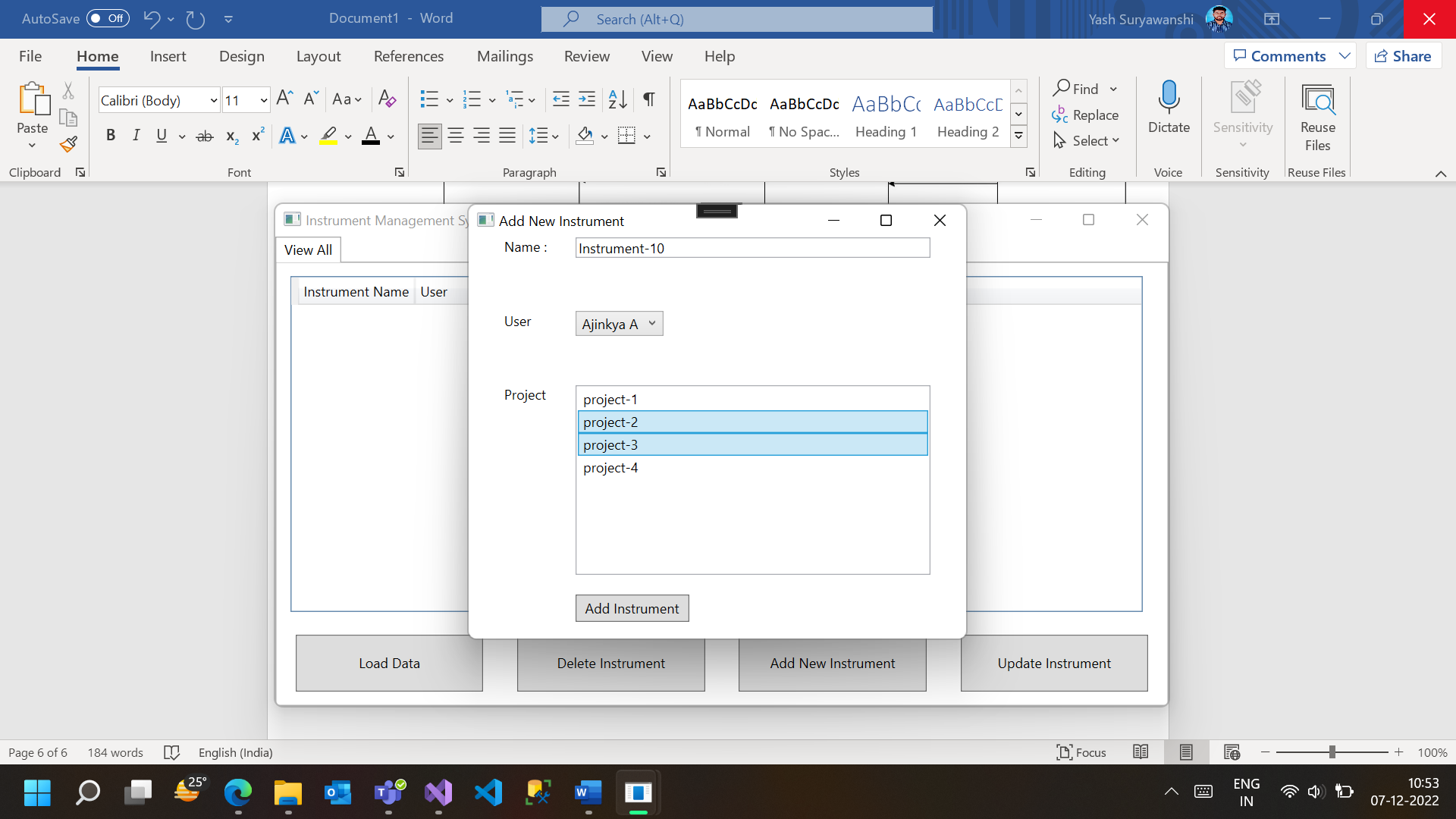
Description automatically generated

1. **Screenshots:**

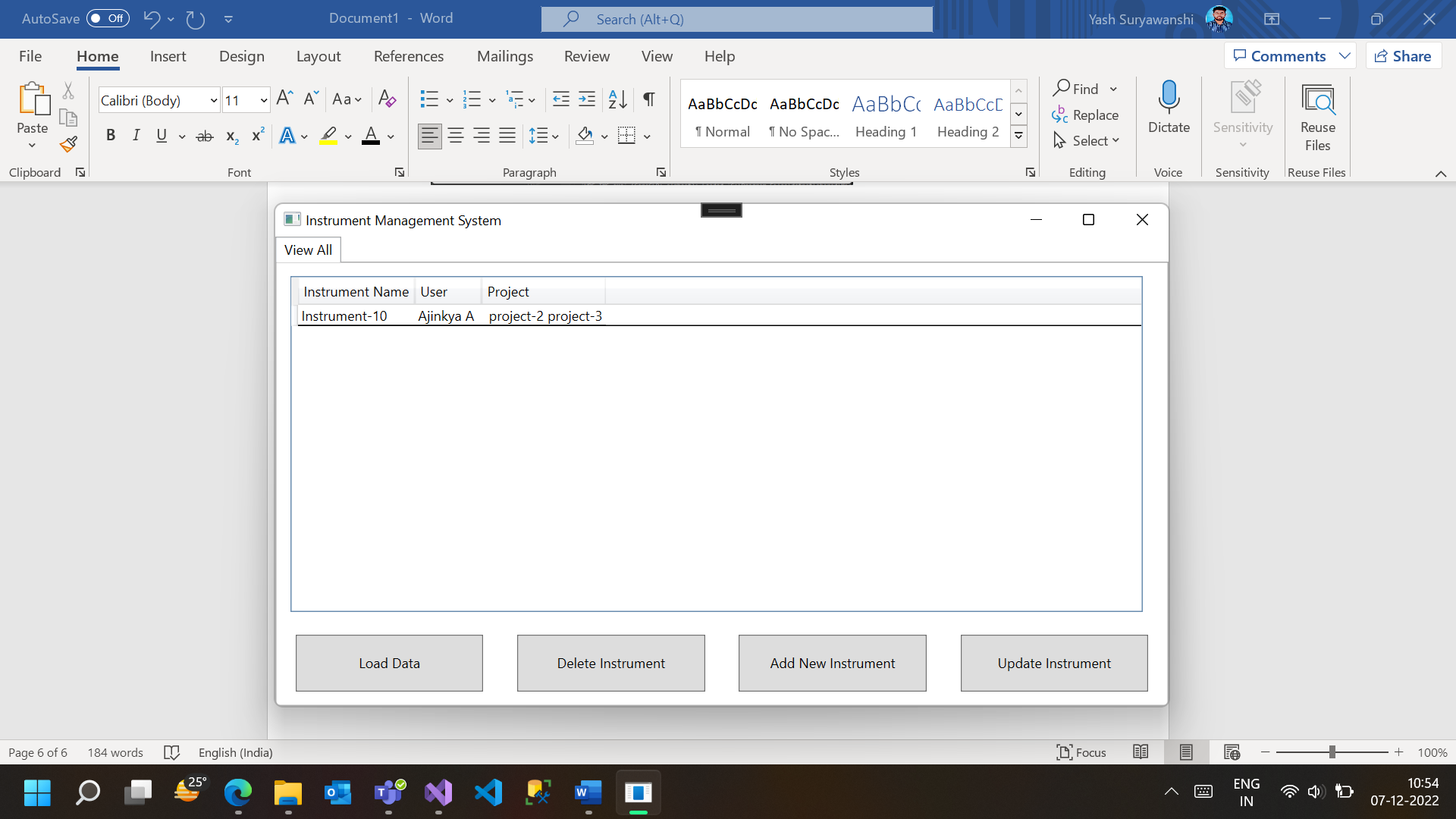
Following is the screenshot of the application when it is run initially. The name of the window is Instrument Management System. If you are running the application for the first time, you will not have xml file on the machine. In that case if you click Load data button, it will inform you that there is no data present, add data to the file.



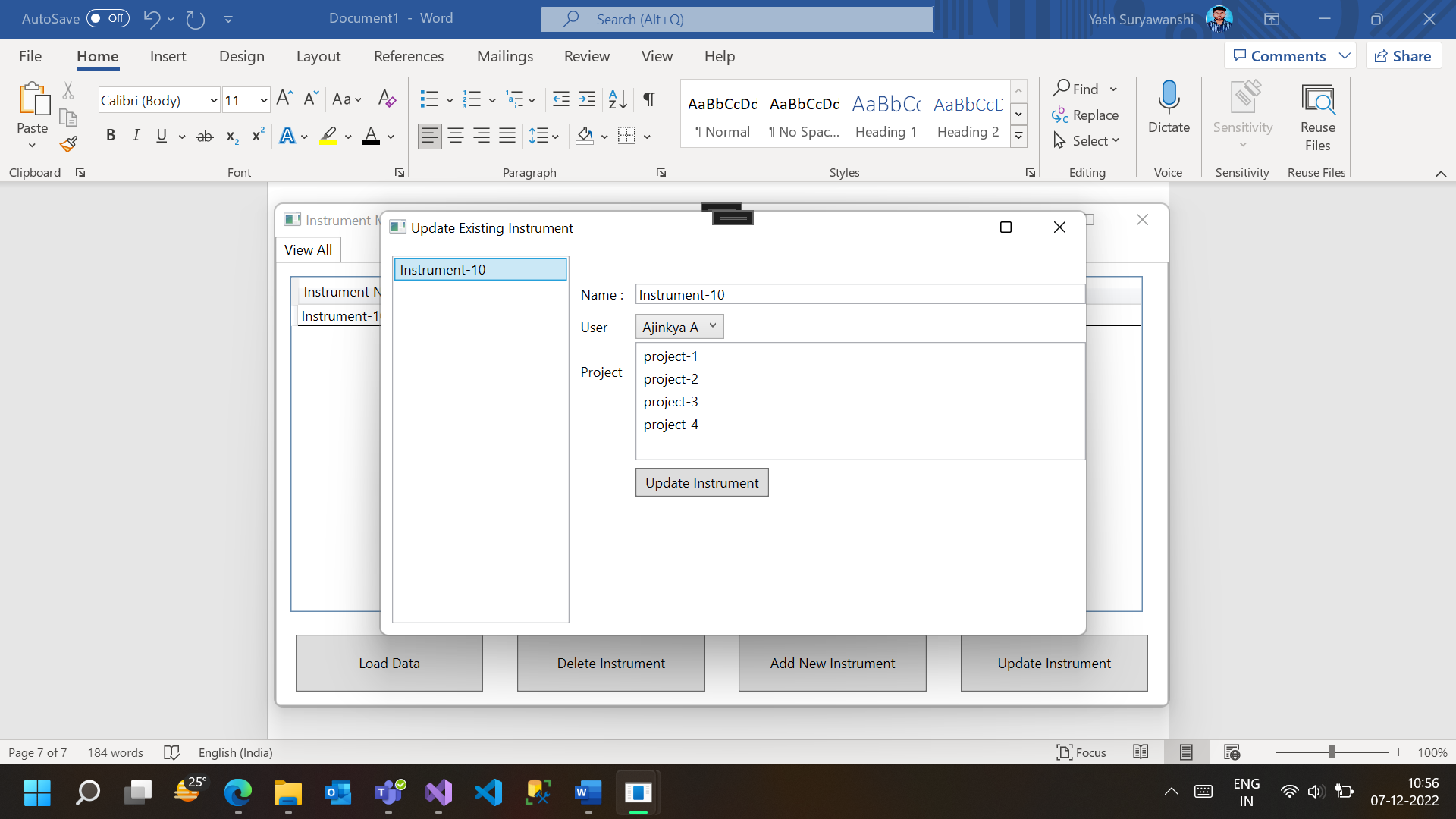
So now when you click Add New Instrument button, a new dialog box pops up. There you can add name of the instrument, select a user of the instrument and select one or more projects from the project list then click Add Instrument. It will add the instrument to the database file.



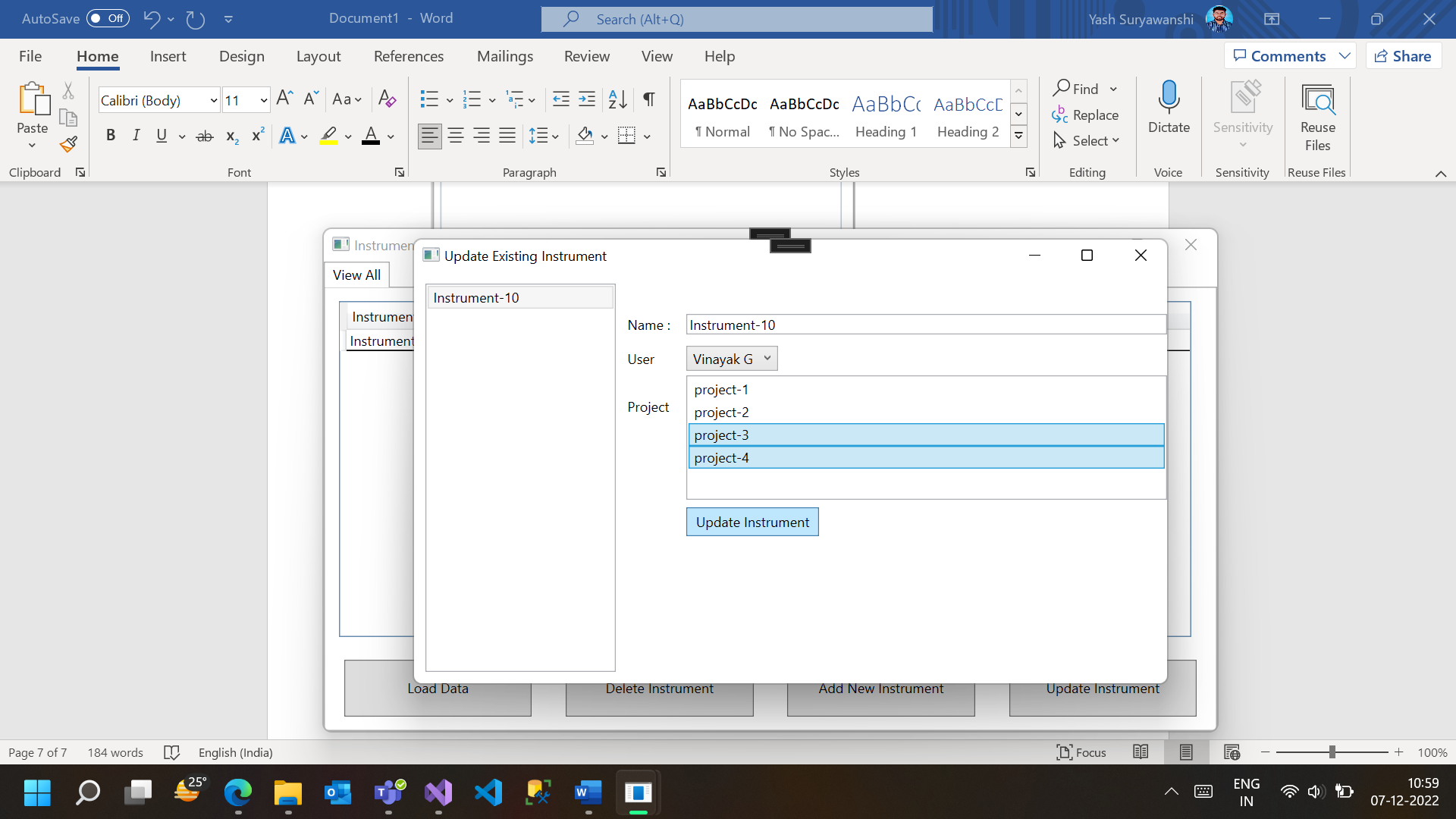
Later when you click Load Data button then you will see the Instrument data that you entered earlier.



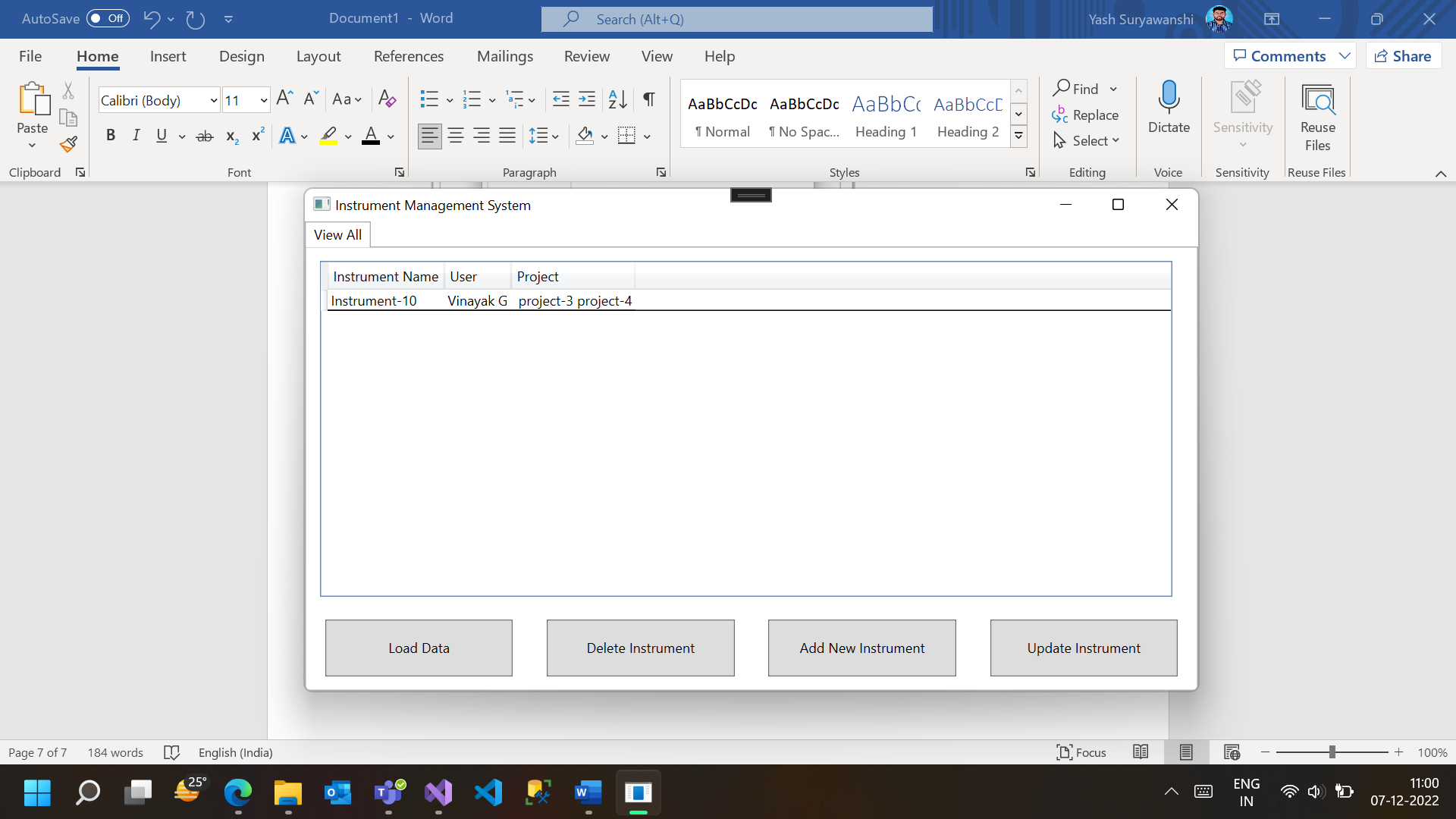
Suppose now you want to update the user and projects of instrument-10, then click on Update Instrument button of Instrument Management System window. A new dialog box will pop up by the name of Update Existing Instrument. There, on the left side, you can select which instrument you want to update.



When you click on instrument-10 on left side, content on right side gets auto-filled with existing data. Now you can update the desired user and desired projects from project list. After making changes click on Update Instrument. It will save the updated changes of instrument-10 in the database file.



Click the Load Data button to check if the updated data is showing on this window.



Suppose you do not want the instrument-10 in your database anymore, you can click on the instrument-10 and then click Delete Instrument. It will delete the instrument from the database file as well.

