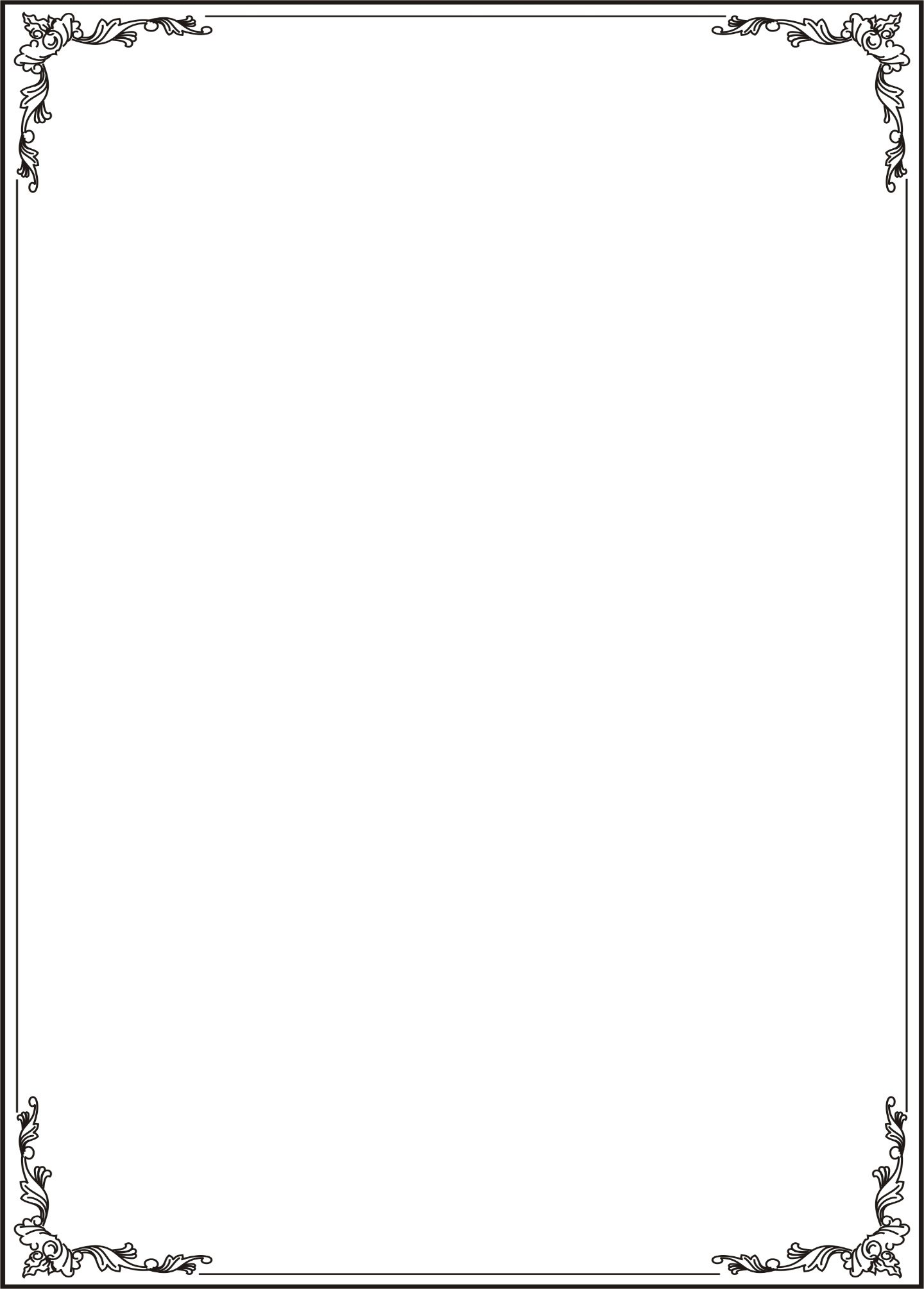
****

**HCMC UNIVERSITY OF TECHNOLOGY AND EDUCATION**

***Faculty For High Quality Training***

-----🙞🙜🕮🙞🙜-----

**FINAL PROJECT REPORT**

**SIMPLE NOTE PROGRAM**

**LECTURER: Mr. Huỳnh Xuân Phụng**

**STUDENT INVOLVEMENT:**

**Thiều Quang Khánh 18110022**

**Trần Đăng Khoa 18110024**

**Đinh Văn Trường 18110060**

**Ho Chi Minh City**

SCORE

|  |  |  |  |
| --- | --- | --- | --- |
| CRITERIA | CONTENT | PRESENTATION | TOTAL |
| POINT |  |  |  |

LECTURER’S COMMENT

**EVALUATION:**

* Thiều Quang Khánh:
* Trần Đăng Khoa:
* Đinh Văn Trường:

LECTURER

Contents

[**Chapter 1: Program Overview** 4](#_Toc43200418)

[**1.** **General Introduction** 4](#_Toc43200419)

[**1.1.** **Simple Note task note software** 4](#_Toc43200420)

[**1.1.1.** **Project request** 4](#_Toc43200421)

[**1.1.2.** **Project analysis** 4](#_Toc43200422)

[**2.** **Simple Note software specification** 4](#_Toc43200423)

[**2.1.** **Simple Note** 4](#_Toc43200424)

[**2.1.1.** **Simple Note introduction** 4](#_Toc43200425)

[**2.1.2.** **Use case diagram** 5](#_Toc43200426)

[**2.1.3.** **Data Input - Output** 5](#_Toc43200427)

[**2.1.4.** **Main feature** 5](#_Toc43200428)

[**2.1.5.** **Expected interface** 6](#_Toc43200429)

[**Chapter 2: Implementation plan** 6](#_Toc43200430)

[**1.** **Plan** 6](#_Toc43200431)

[**2.** **Assignment of work** 7](#_Toc43200432)

[**Chapter 3: Design Software** 8](#_Toc43200433)

[**1.** **Interface** 8](#_Toc43200434)

[**2.** **Algorithm** 14](#_Toc43200435)

[**3.** **Diagram** 22](#_Toc43200436)

[**4.** **Database** 23](#_Toc43200437)

[**Chapter 4: Testing Software** 24](#_Toc43200438)

[**Chapter 5: Group’s conclusion** 24](#_Toc43200439)

[**1.** **Conclusion** 24](#_Toc43200440)

[**2.** **Development** 24](#_Toc43200441)

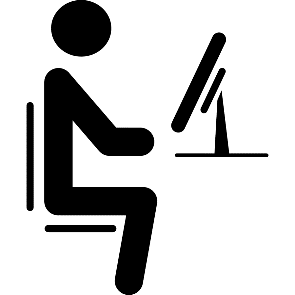
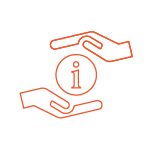
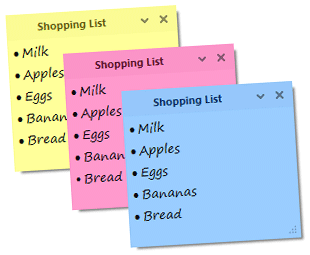
# **Chapter 1: Program Overview**

1. **General Introduction**
   1. **Simple Note task note software**
      1. **Project request**

* Design and build object-oriented software to solve task notes with a simple and easy to use interface.
  + 1. **Project analysis**
* Building object-oriented software.
* String analysis to get input data
* Store and display as text when the user requests.
  + 1. **Direction of implementation**
* Building object-oriented software with C #.

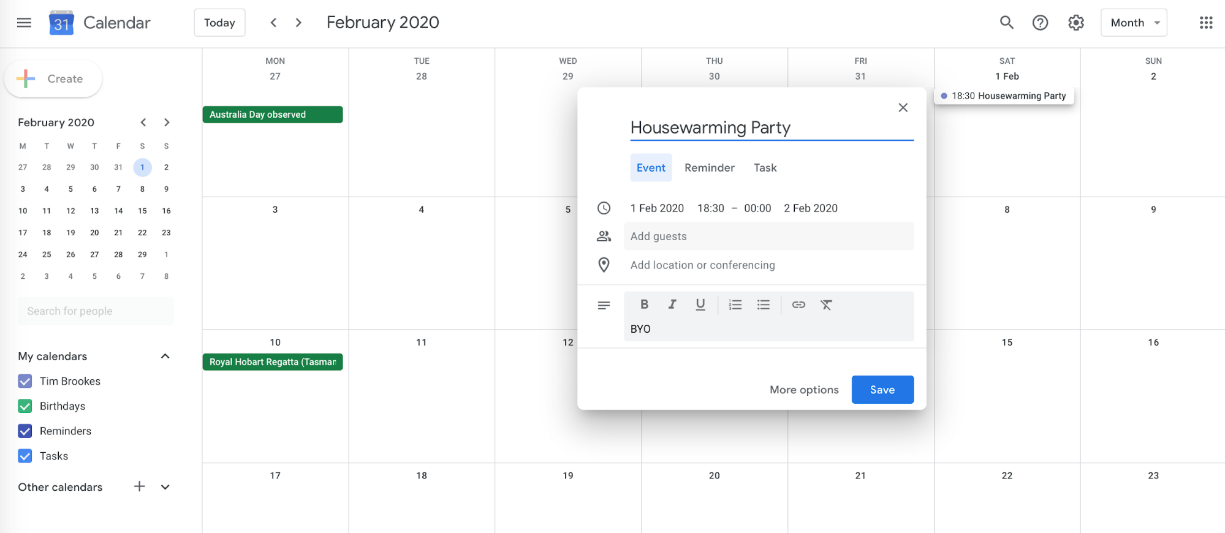
1. **Simple Note software specification** 
   1. **Simple Note**
      1. **Simple Note introduction**

* Simple Note, like its title, is a small note-taking software that can be run without installation. Simple Note provides users with an easy reminder solution to not forget important notes. Now, you can completely throw away all the paper in the room and neatly organize notes right on the desktop.
  + 1. **Use case diagram**

User Simple Note Note

* + 1. **Data Input - Output**
* **Input:** Text entered from keyboard
* **Output:** Display the saved user text under a note interface (Day, Month, Year, Content).
  + 1. **Main feature**
* Write and store notes in the database, allowing users to edit note formats.
* Display detailed note information (content, status (DONE, DOING, MISS, COMING), Time).
* Search for notes by note title and content by note.
* Display calendar and note management interface
* Display notification if there are notes in the day.
  + 1. **Expected interface**
* The interface displays in an elegant square style, using a light background color that does not cause eye strain for the user.
* Simple click and type operations.



# 

# **Chapter 2: Implementation plan**

1. **Plan**

**Work plan**

1- Learn about Simple Note software, outline ideas about interfaces and functions.

2- Analyze input and software requirements. Start designing the interface.

3- Build the main interface layers and main functions for the software

* Create new note
* Edit note (Copy, Cut, Paste, Select All)
* Save modified note
* Search note

4- Building extended functions for the software.

* Edit note (Change Font, Color, Size, Undo, Redo)
* Export note into .txt files
* Import note from .txt files
* Set alarm
* Create Recycle bin

5- Building a basic calendar interface for the software.

6- Create the database.

7- Connecting database and software, fixing errors and completing the software.

8- Write the report.

1. **Assignment of work**

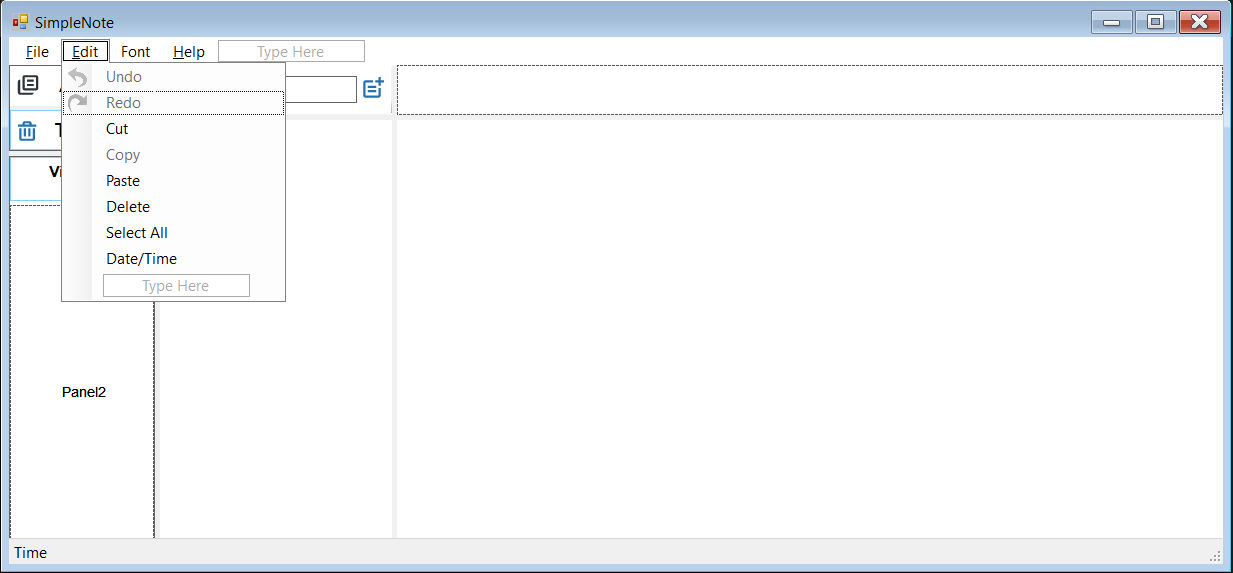
|  |  |  |  |
| --- | --- | --- | --- |
|  | Name | Task description |  |
| 1 | Thiều Quang Khánh | * Divide the work for member * Search note (Name, content) * Create database * Connect to SQL * Check and fix bugs | 34% |
| 2 | Trần Đăng Khoa | * Calendar * Alarm function * Design interface * Write Report | 33% |
| 3 | Đinh Văn Trường | * Write function: Change font, undo, redo * Edit note: Copy, Cut, Paste. * Delete note. * Trash bin * Write report | 33% |

# **Chapter 3: Design Software**

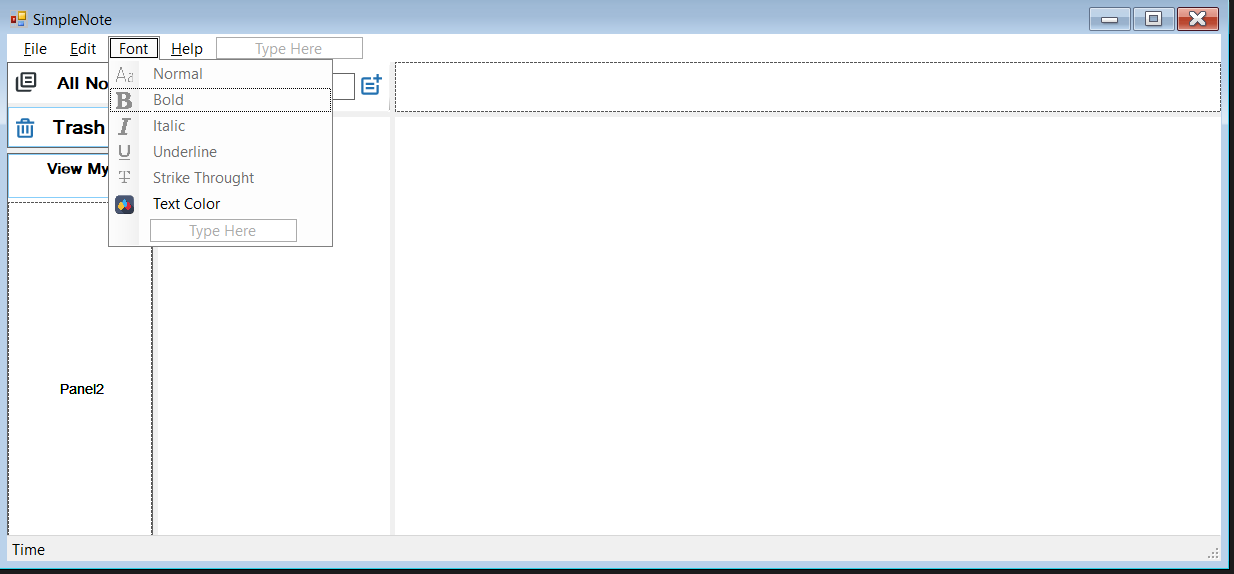
1. **Interface**

|  |
| --- |
| Main view( Note)  This design’s program was created by Winform. This program includes three main interfaces: Note, TrashNote, Calendar. |
| Note: Delete, Remind, Information, Add button, . |
| Trash Note: Delete forever, Restore.    Calendar: Notify.  *Menu strip*  File: New Note, Open, Save, Save As, Exit. |
|  |

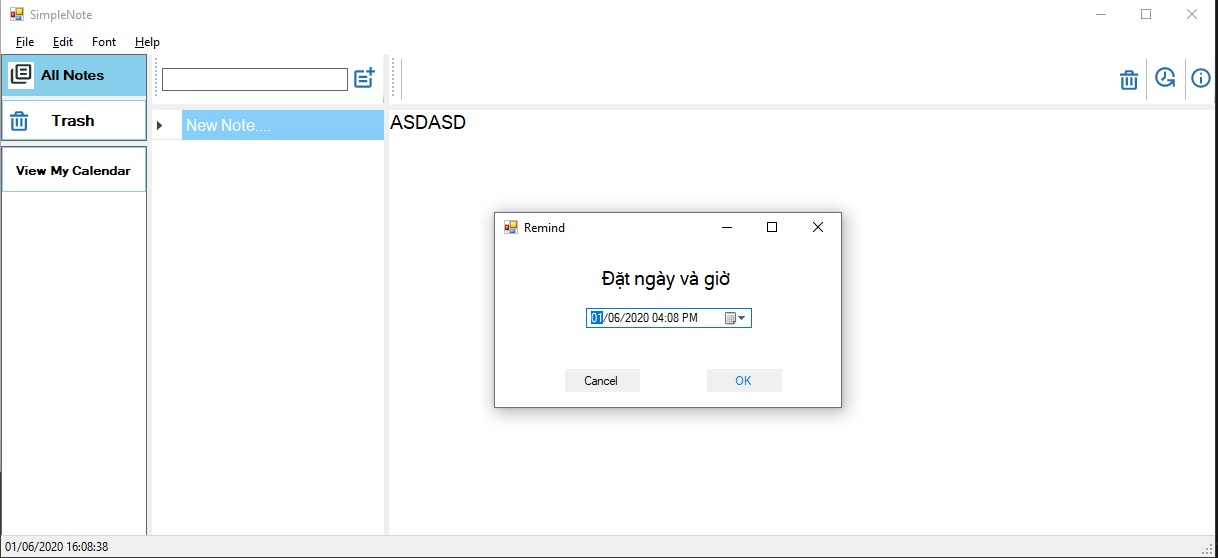
Edit: Undo, Redo, Cut, Copy, Paste, Delete, Select All, Date/Time.



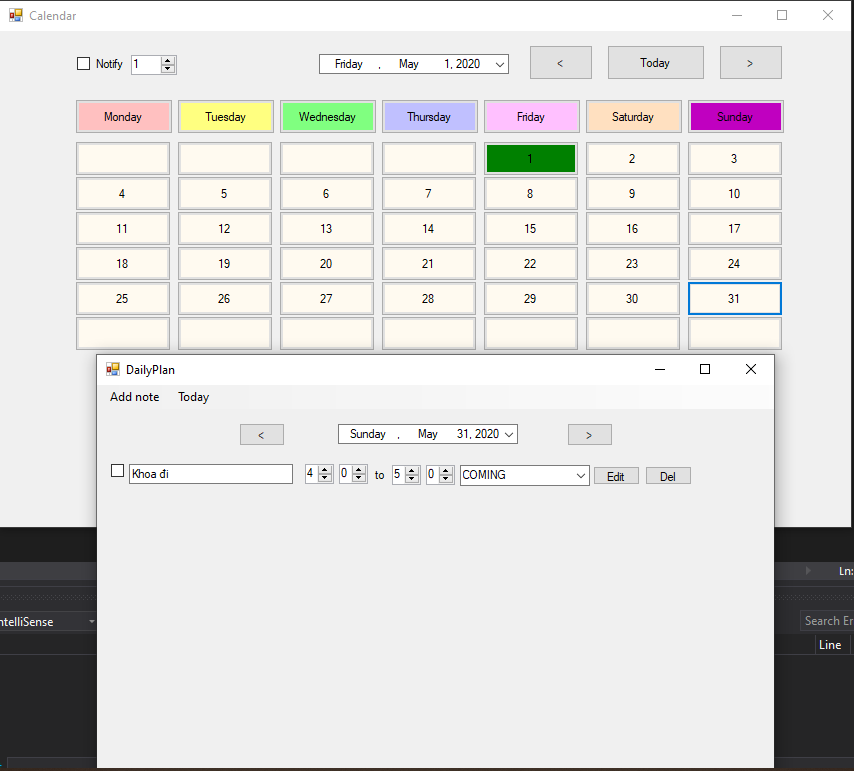
Font: Normal, Bold, Italic, Underline, Strike Through, Text Color.



*Additional development features*



Set an alarm to remind important note or event.



Calendar Work to manage all event that was reminded by user in the next few days.

1. **Algorithm**

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Class | Purpose | Student name |
| 1 Controller | NoteController | Execute from data including updating(content,name),getting notes,insert and delete note. | Thieu Quang Khanh |
| 2 | RemindNoteController  Inherits from NoteController | Execute all information about Note that was reminded | Thieu Quang Khanh |
| 3 | TextController  Inherits from NoteController | Update a font style and content color to database | Thieu Quang Khanh |
| 4 | NoteTagController  Inherits from NoteController | Save a tag note to database when user add a tag | Thieu Quang Khanh |
| 5 | TrashNoteController  Inherits from NoteController | Store which note is deleted and user can restore or delete it forever. | Thieu Quang Khanh |
| 6 View | Form1 | Is the main form | Thieu Quang Khanh |
| 7 | Form2 | Is a calendar form to | Thieu Quang Khanh |
| 8 | DailyPlan | Show all the jobs on day | Thieu Quang Khanh |
| 9 | FrmRemind | Remind user about the status of notes in paticular date and time | Thieu Quang Khanh |
| 10 Models | clasData | Connect,add,delete,update data | Thieu Quang Khanh |
| 11 | PlanItem | Set types of status including done,doing,coming,missed | Thieu Quang Khanh |
|  |  |  |  |

B.Methods in class

B.1. Methods in class form1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Method | Purpose | File name,line cotain | Student name |
| 1 | ShowNoteContent()  Output:none | Store content and color of note and adding a tag | Views/Form1.cs  (198) | Dinh Van Truong |
| 2 | tooldStripbtnDelete()  output:none | Delete selected note. | Views/Form1.cs (231) | Dinh Van Truong |
| 3 | DVGNoteName\_CellEndEdit()  Output:none | Allow user to change note name in each row of data grid view | Views/Form1.cs  (281) | Dinh Van Truong |
| 4 | btnAllNote()  output:none | Method is used to display all note cotent that is stored in data table | Views/Form1.cs  (332) | Dinh Van Truong |
| 5 | btnTrashNote()  output:none | Call data table from trashNoteController and get the data source and display all deleted notes | Views/Form1.cs  (290) | Dinh Van Truong |
| 6 | toolstripBtnDeleteForever()  toolstripBtnRestore()  output:none | Get all notes from trashNoteController  then allow user delete or restore note. | Views/Form1.cs | Dinh Van Truong |
| 7 | saveAstoolStripMenuItem()  exitToolStripMenuItem()  ouput:none | Call a save of open dialog then open or save by a txt file and read or write note content | Views/Form1.cs  (665) | Dinh Van Truong |

B.2.Methods in form2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Method | Purpose | File name,line cotain | Student name |
| 1 | DayofMonth(DateTime date)  Input:date  Output:the day of month(28,29,30,31) | Count a numbers of days in particular month | Views/Form2.cs  (80) | Tran Dang Khoa |
| 2 | Bool HavenoteOnDates(Datetime date)  Input:date  Output:true/false | Remind user have how many notes in day | Views/Form2.cs  (134) | Tran Dang Khoa |

B.3.Method in DailyPlan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Method | Purpose | File name,line cotain | Student name |
| 1 | ShowJobByDate(date)  Input:date  Output:none | show how many jobs in selected date | Views/DailyPlan.cs | Thieu Quang Khanh |

B.4.Method in FrmRemind

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TT | Method | Purpose | File name,line cotain | Student name | |
| 1 | btnOK\_click()  input:none Thieu Quang Khanh  output:none | Insert and update remind note including content and status from remindNoteControler | Views/Remind.cs | | Thieu Quang Khanh |

B.5.Methods in claData

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Method | Purpose | File name,line cotain | Student name |
| 1 | Boolean exedata(string cmd)  Input string cmd  Ouput:true/false | Add ,delete,update date from database | Models/claData  (33) | Thieu Quang Khanh |
| 2 | DataTable readdata(string cmd)  Input:string cmd  Output:data table | Read data and return data from data table | Models/claData  (51) | Thieu Quang Khanh |

B.6.Methods in PlanItem

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Method | Purpose | File name,line cotain | Student name |
| 1 | List<string> ListStatus  Input:none  Output:List<string> | Static method to output all status of note(done,doing,coming,missed) | Models/PlanItem(31) | Thieu Quang Khanh |

B.7.Methods in NoteController

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Method | Purpose | File name,line cotain | Student name |
| 1 | DataTable getNotes()  Input:none  Output:data table | Virtual method to read data and then get note | Controllers/NoteController/cs  (15) | Tran Dang Khoa |
| 2 | UpdateNoteContent(string NoteContent,int IDNote)  Input:NoteCotent,IDNote  Output:true/false | A bolean method that check to update note content and id note | Controllers/NoteController.cs(20) | Tran Dang Khoa |
| 3 | DataTable getNotes(int Idnote) | A method that get note by id note | Controllers/NoteController.cs(30) | Tran Dang Khoa |
| 4 | getNoteDateCreated(int IDNote)  input:IDNote  output:data table | A methed that get a date which is note created | Controllers/NoteController.cs(45) | Tran Dang Khoa |
| 5 | Boolean InsertNote(DateTime NoteDateCreated)  Input :NoteDateCreated  Output:true/false | A bolean method check to add a new note | Controllers/NoteController.cs(50) | Tran Dang Khoa |
| 6 | Virtual Boolean DeleteNote(int IDNote) | A virtual Bolean method that delete note from data table | Controllers/NoteController.cs(55) | Tran Dang Khoa |
| 7 | UpdateNoteName(sting NoteName, int NoteId)  Input: NoteName, NoteId  Output: true/false | Method to update Note name for Note that has this NoteId. Return true or false to check if successful or not | Controllers/NoteController.cs(25) | Tran Dang Khoa |
| 8 | DataTable getNotes(string)  Input: SearchText  Output: DataTable Note | Method to get list Note that Note name containing SearchText.Return Data Table of Notes. | Controllers/NoteController.cs(35) | Tran Dang Khoa |
| 9 | DataTable getNoteName(int IDNote)  Input: Note Id  Output: DataTable NoteName | Method to get list NoteName of Note that has ID = NoteId.Return Data Table of Notes. | Controllers/NoteController.cs(40) | Tran Dang Khoa |
| 10 | DataTable getNotesbyNoteContent(string NoteContent)  Input: Note’s Content search Text  Output: DataTable Note that Note’s content contains Note’s Content Search Text | Method to help user search Note by Note’s content containing Note’s Content Search Text. Return DataTable of Notes. | Controllers/NoteController.cs(60) | Tran Dang Khoa |
| 11 | DataTable getNotesbyNoteTag(string NoteTag)  Input: Note’s Tag search text  Output: Data Table of Notes | Method to help user search Note by Note’s Tag containing Note’s Tag Search Text. Return DataTable of Notes. | Controllers/NoteController.cs(65) | Tran Dang Khoa |

B.8.Methods in NoteTagController:NoteController

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Method | Purpose | File name,line cotain | Student name |
| 1 | Boolean UpdateNoteTag(int NodeId,string NoteTag)  Input:NodeId,NoteTag  Output:true/false | Check to update a tag note to data table | Controllers/NoteTagController.cs  (14) | Dinh Van Truong |
| 2 | DataTable GetNoteTag()  Input:  Output: DataTable of Note | Get a tag note from data table | Controllers/NoteTagController.cs  (19) | Dinh Van Truong |
| 3 | DataTable GetNoteTag(int IDNote)  Input: Note’s Id  Output: DataTable of Tag | Method help user to get Tag of Note that contains IDNote | Controllers/NoteTagController.cs  (24) | Dinh Van Truong |

B.9.Method in RemindNoteController

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Method | Purpose | File name,line cotain | Student name |
| 1 | Boolean InserRemindNote(int NoteID,Datetime NoteDate,string NoteStatus)  Input:NoteID,NoteDate,NoteStatus  Ouput:true/false | Boolean method to check to insert a remind note | Controllers/RemindNotecontroller.cs(15) | Thieu Quang Khanh |
| 2 | Boolean UpdateRemindNote int NoteID,Datetime NoteDate,string NoteStatus)  Input:NoteID,NoteDate,NoteStatus  Ouput:true/false | Boolean method to check to get a remind note | Controllers/RemindNotecontroller.cs(20) | Thieu Quang Khanh |
| 3 | DataTable getRemindNotes(DateTime date)  Input: Date Time’s Note  Output: DataTable of Note | This method help user to get Note’s information on that day = DateTime. | Controllers/RemindNotecontroller.cs(25) | Dinh Van Truong |
| 4 | int CountRemindNotes(DateTime date)  Input: Datetime  Output: quantity of Note on DateTime | This method help user to count how many notes on that day | Controllers/RemindNotecontroller.cs(30) | Dinh Van Truong |
| 5 | DataTable getNotes()  Input:  Output: Note Name, Note Date Reminded, Note status, Note Id | This method help user to get information of Note that was reminded | Controllers/RemindNotecontroller.cs(35) | Dinh Van Truong |
| 6 | bool DeleteNote(int IDNote)  Input: NoteId  Output: true/false | This method help user to delete Note that was reminded. Return true or false if successful or not | Controllers/RemindNotecontroller.cs(40) | Dinh Van Truong |

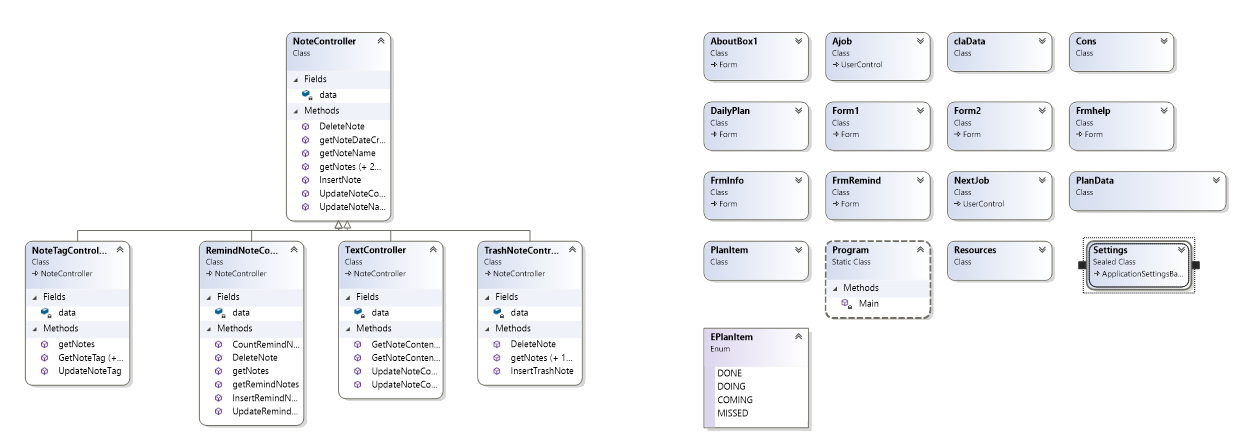
B.10.Methods in TextController

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Method | Purpose | File name,line cotain | Student name |
| 1 | Boolean check UpdateNoteContentFontStyle(fontStyle style,int IDNote)  Input:style,IDNote  Output :true/false | Update a current font style of note | Controllers/Textcontroller.cs  (15) | Thieu Quang Khanh |
| 2 | Boolean check UpdateNoteConTeentColor(Color color,int IDNote)  Input:style,IDNote  Output :true/false | Update a current content color | Controllers/Textcontroller.cs  (15) | Thieu Quang Khanh |
| 3 | FontStyle GetNoteContentFontStyle(int IDNote)  Input:IDNote  Output:style | Allow to choose multiple font style of note | Controllers/Textcontroller.cs  (25) | Thieu Quang Khanh |

B.11.Methods in TrashNoteController

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Method | Purpose | File name,line cotain | Student name |
| 1 | Boolean InsertTrashNote(int IDNote) | Boolean method to check to add a deleted note to trash note | Controllers/TrashNoteController.cs  (14) | Thieu Quang Khanh |
| 2 | Override bool DeleteNote(IDNote)  Input:IDNote  Output:true/false | Boolean method to check to delete note from trash note | Controllers/TrashNoteController.cs  (29) | Thieu Quang Khanh |

1. **Diagram**



1. **Database**

3.Database design

|  |  |  |
| --- | --- | --- |
| TT | Table name | Purpose |
| 1 | TrashNote | Strore deleted note |
| 2 | Note | store note information such as cotain ,id.. |
| 3 | RemindNote | Store to remind a status of notes,date reminded |

4.Field in table

4.1.Field in TrashNote

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Field name | Data type | Purpose |
| 1 | NoteId | Int | Store Id of notes |
| 2 | NoteDateDeleted | Datetime | Store date of deleted notes |

4.2.Fields in Note

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Field name | Data type | Purpose |
| 1 | NoteId | Int | Store id of notes |
| 2 | NoteName | Nvarchar | Store name of notes |
| 3 | NoteContent | Nvarchar | Store content of notes |
| 4 | NoteContentStyle | Nvarchar | Store font style of notes |
| 5 | NoteContentColor | Nvarchar | Store font color of notes |
| 6 | NoteDateCreated | datetimr | Store date of created note |
| 7 | Note tag | nvarchar | Store tags of notes |

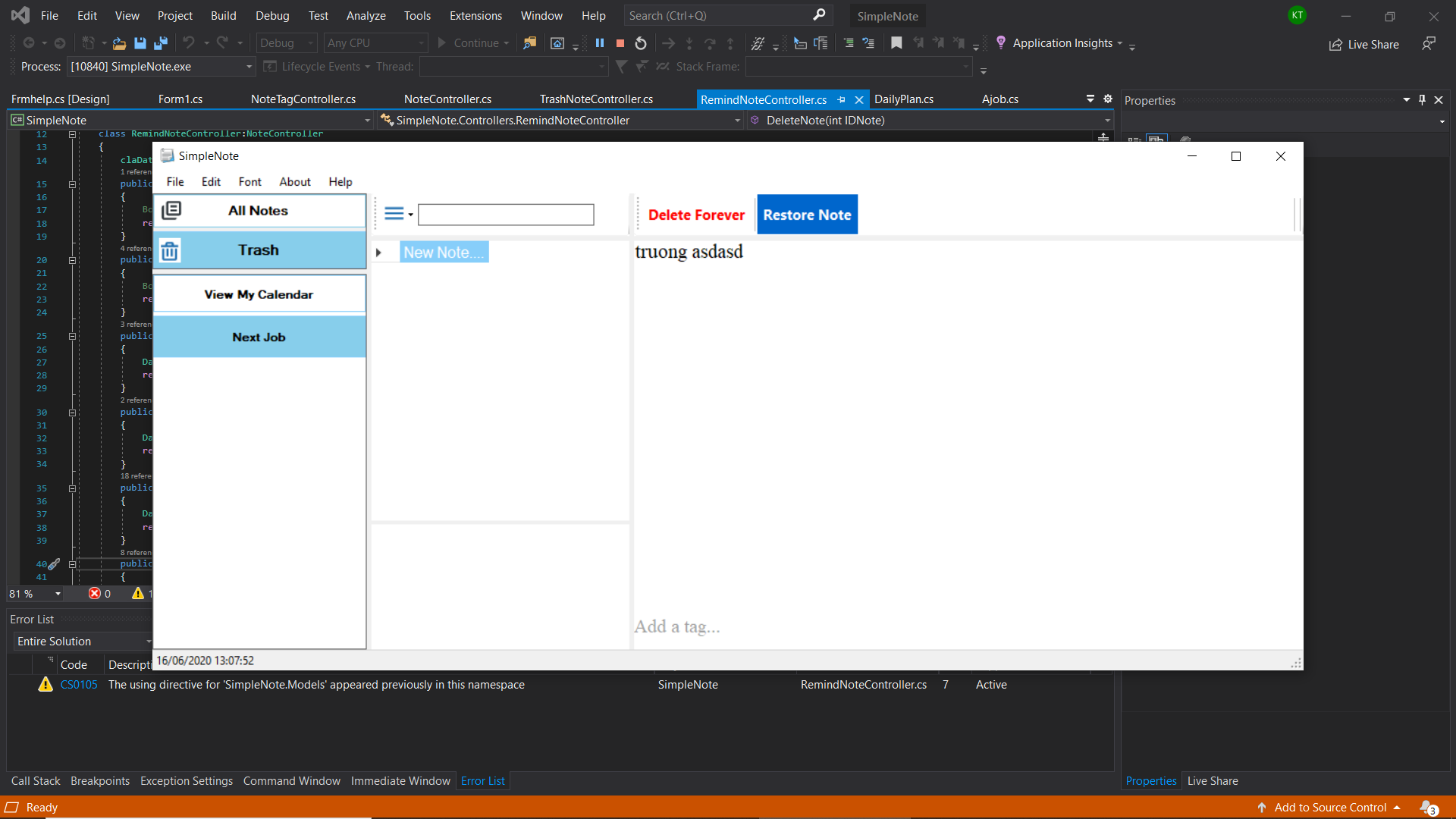
4.3.Fields in RemindNote

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Field name | Data type | Purpose |
| 1 | NoteId(primary key) | Id | Store Id of notes |
| 2 | NoteDateReminded | Datetime | Store date of notes created to remind |
| 3 | NoteStatus | nvarchar | Store status of notes(done,doing,coming,missed) |
|  |  |  |  |

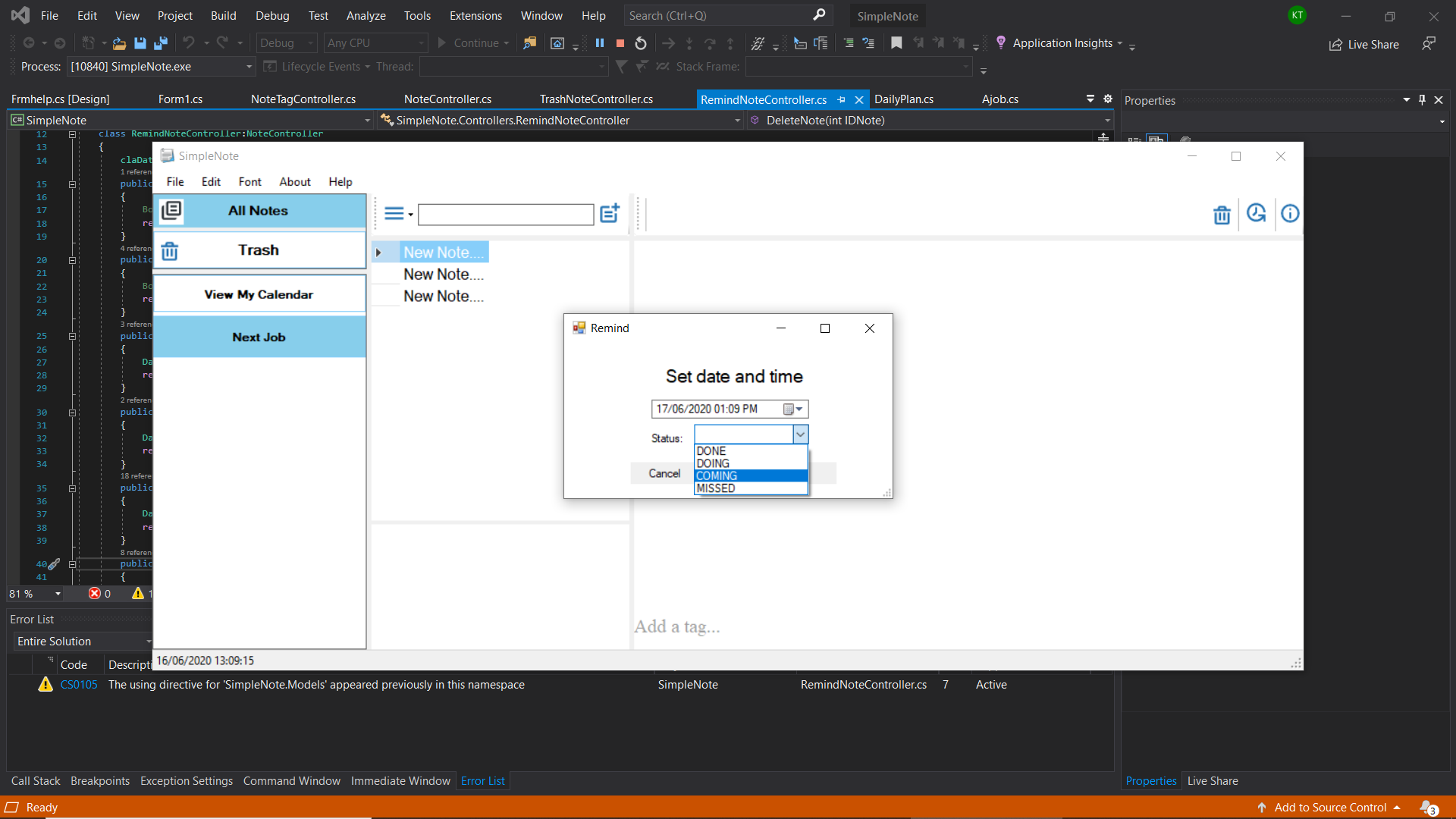
# 

# 

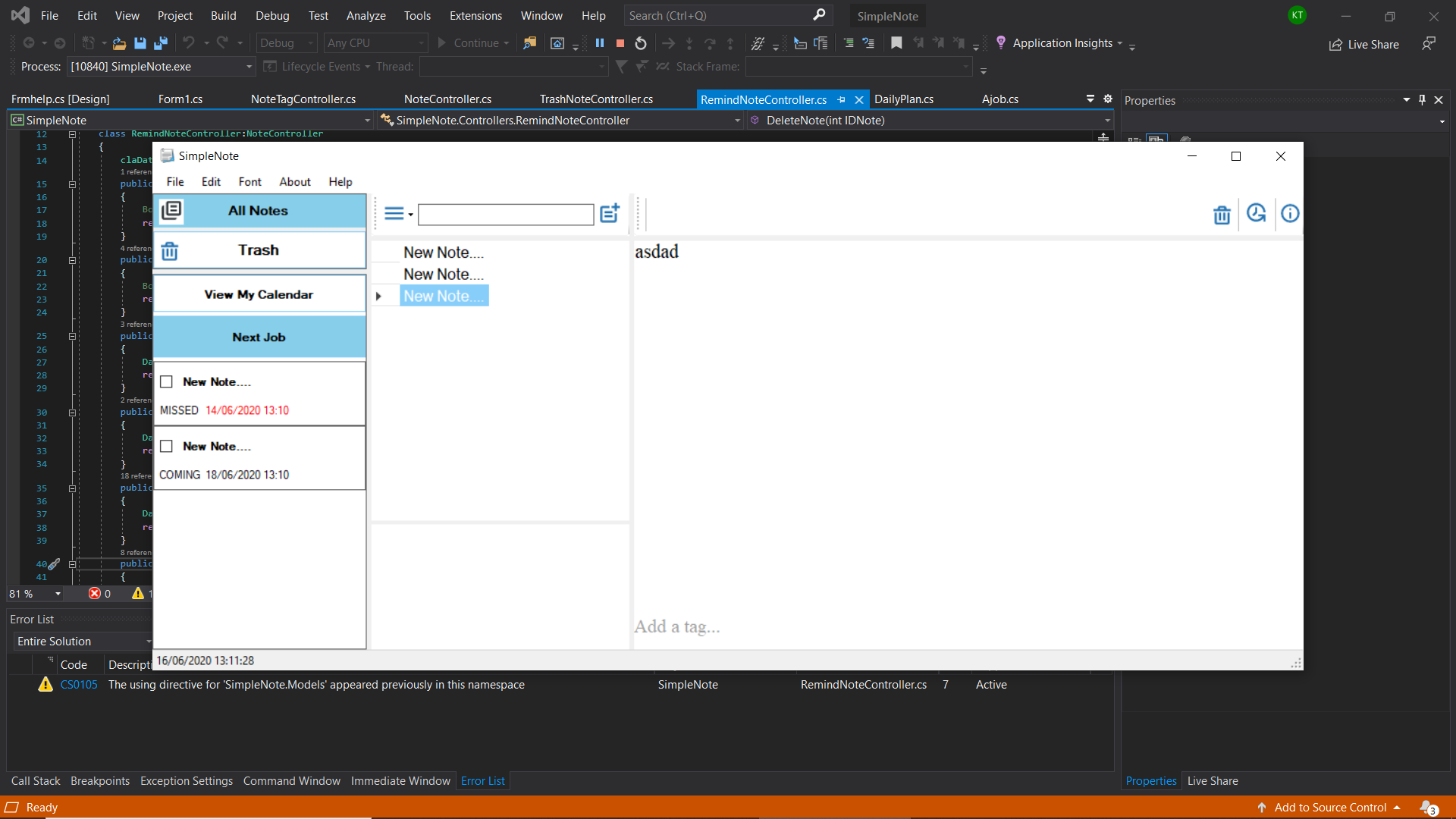
# **Chapter 4: Testing Software**



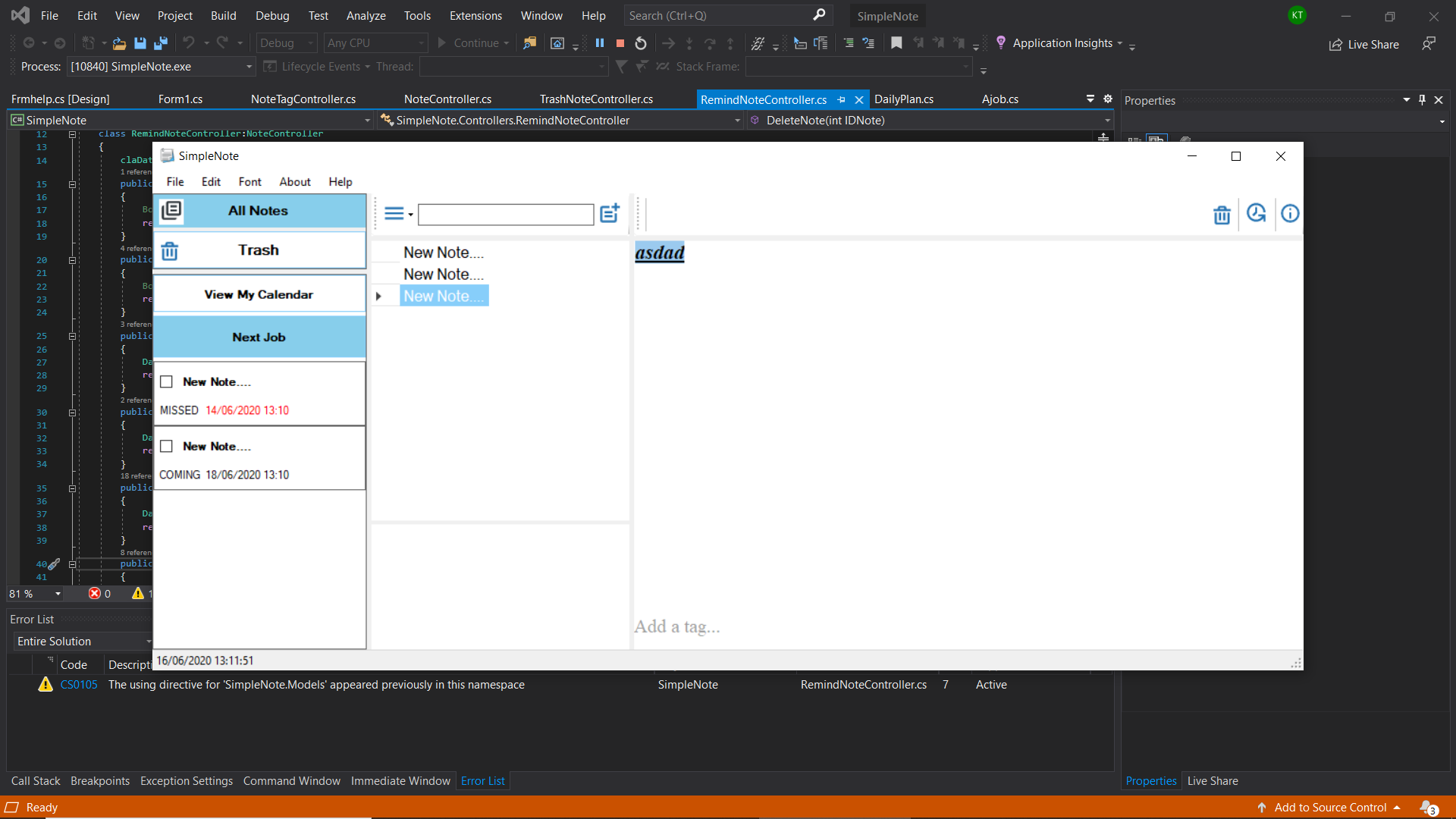
Picture 1: Test deleted note stored in trash



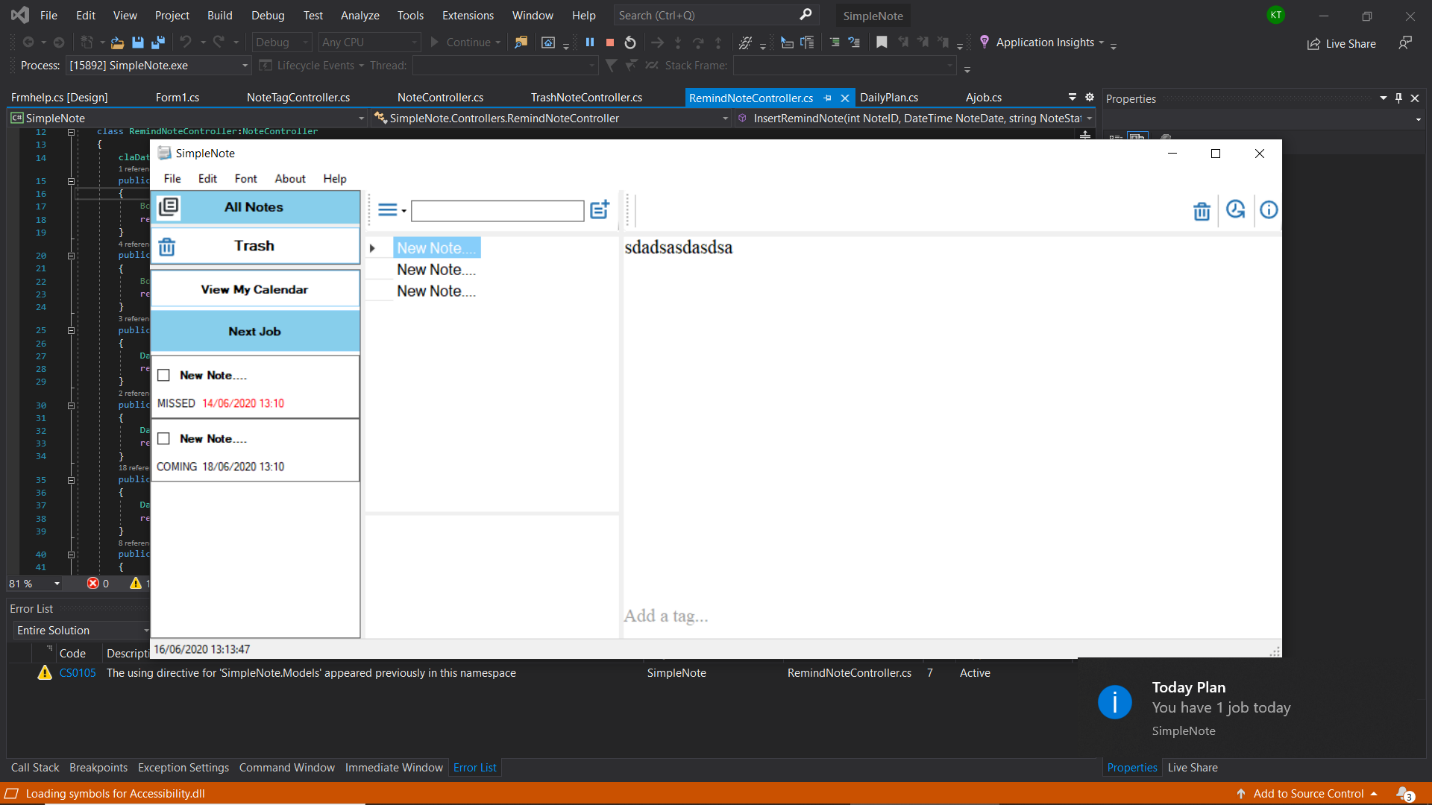
Picture2 : Set status of note



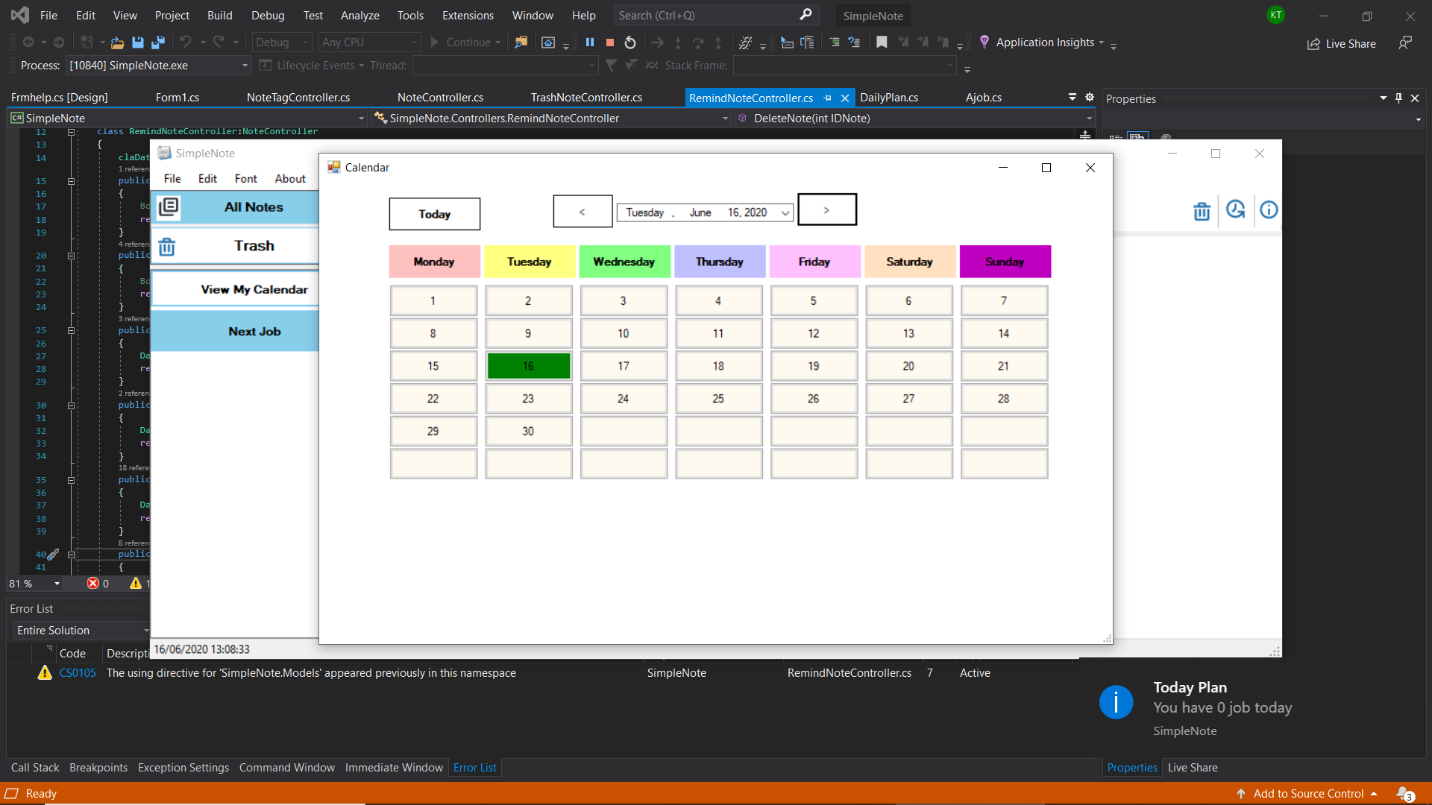
Picture 3 :A status of note will appear in the leftmost



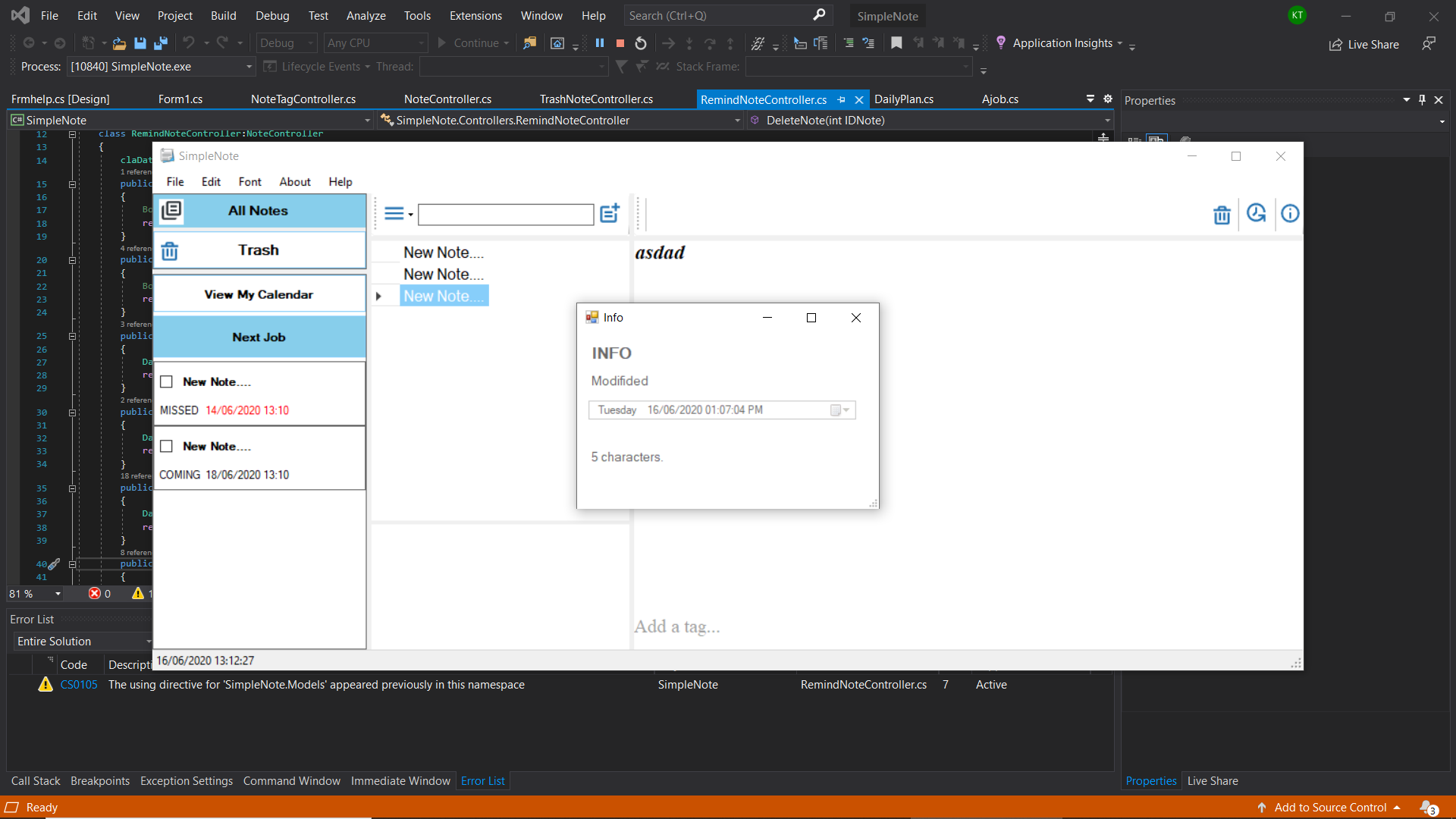
Picture4 : Test font style of note



Picture 5 : Test save note’s content



Picture 6 : show a status of date depend on status of note

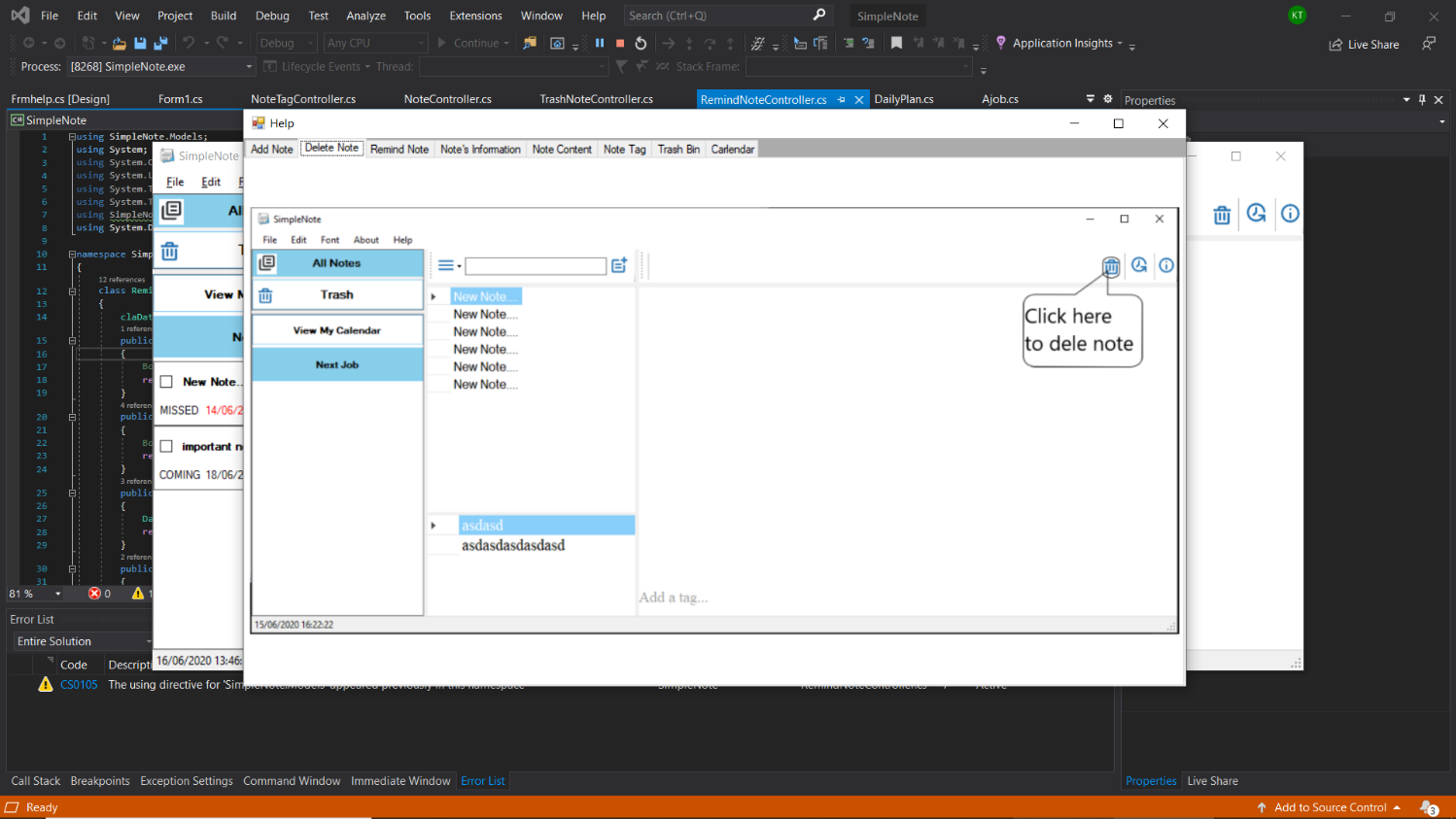


Picture 7: show information of note

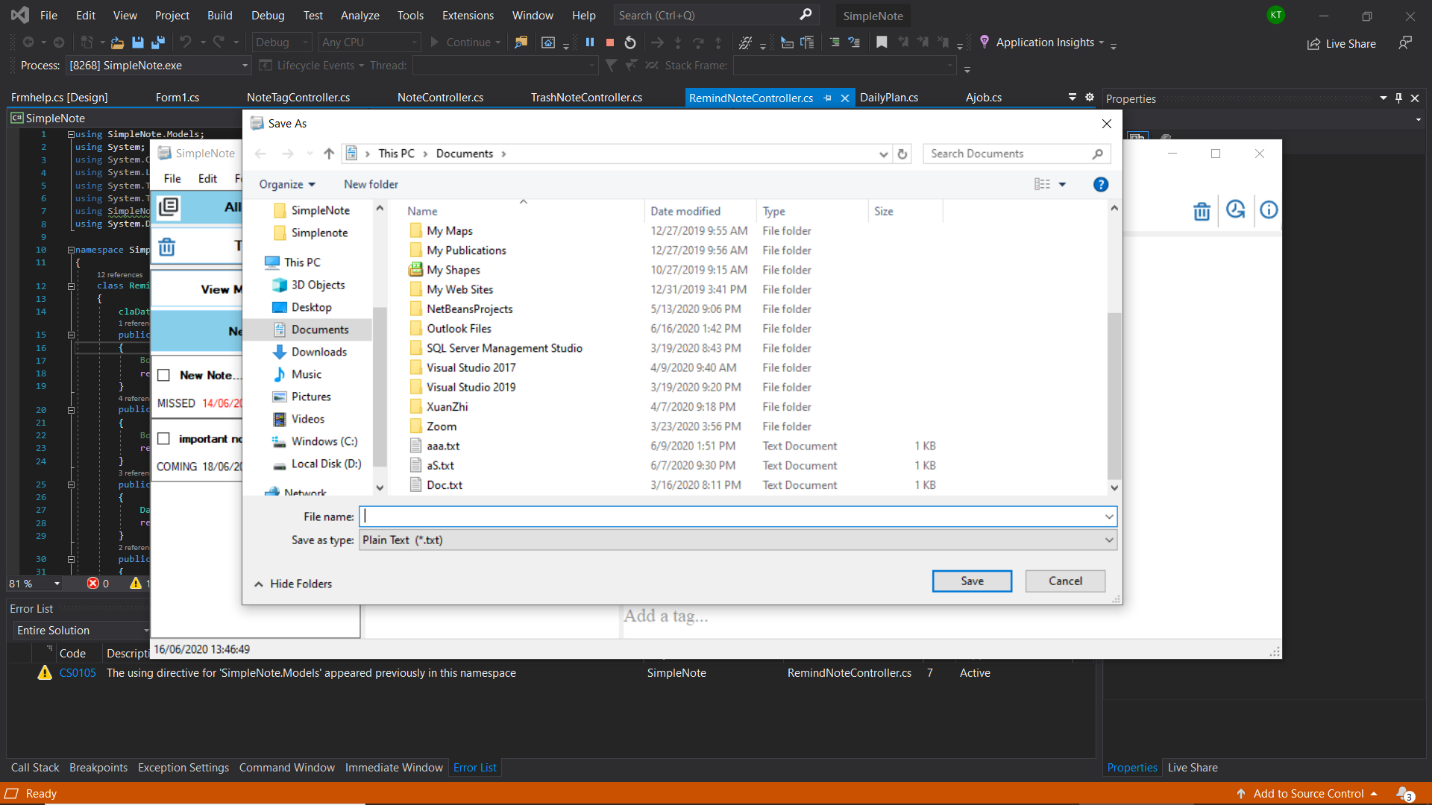
**A screenshot of a computer screen

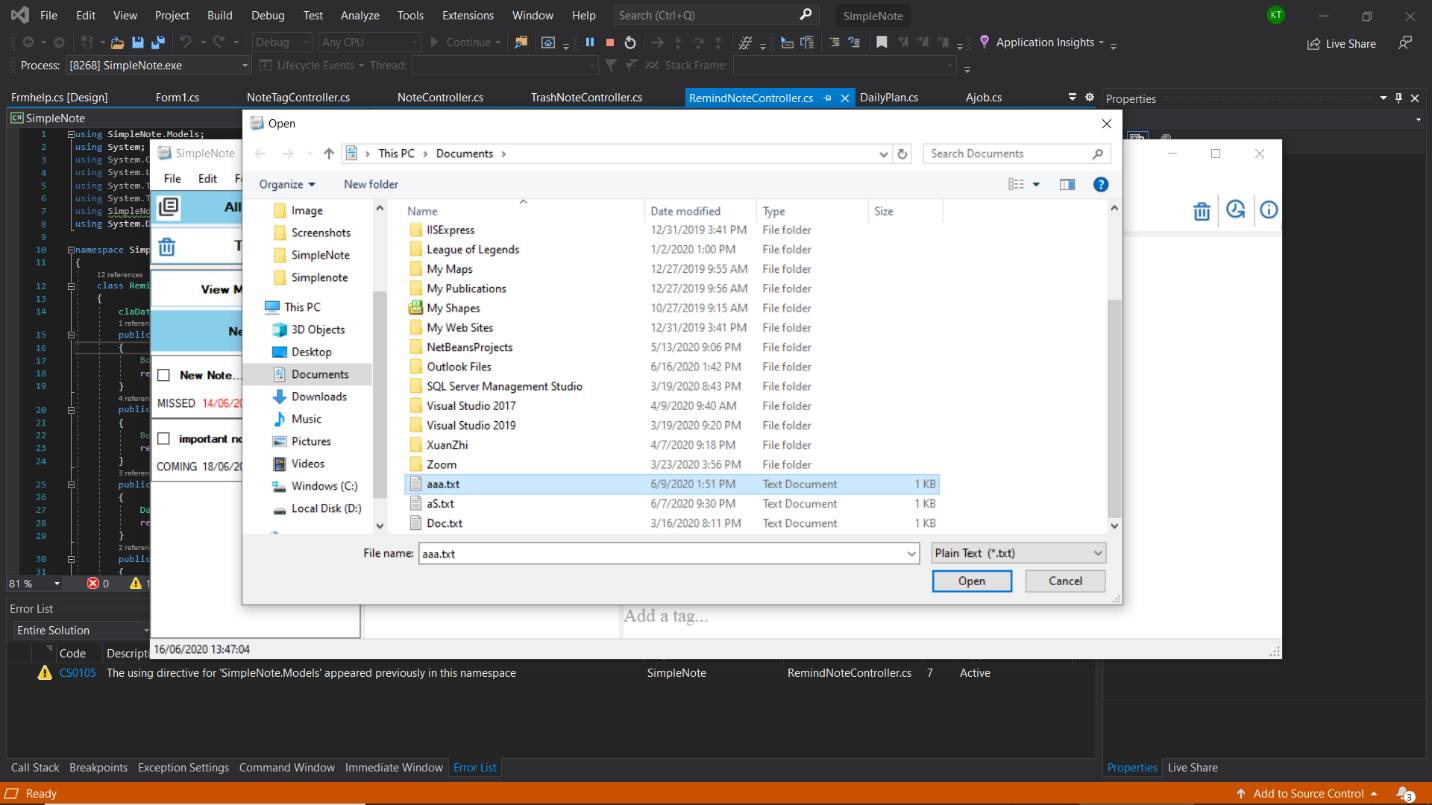
Description automatically generated**

Picture 8: test rename of note’s name

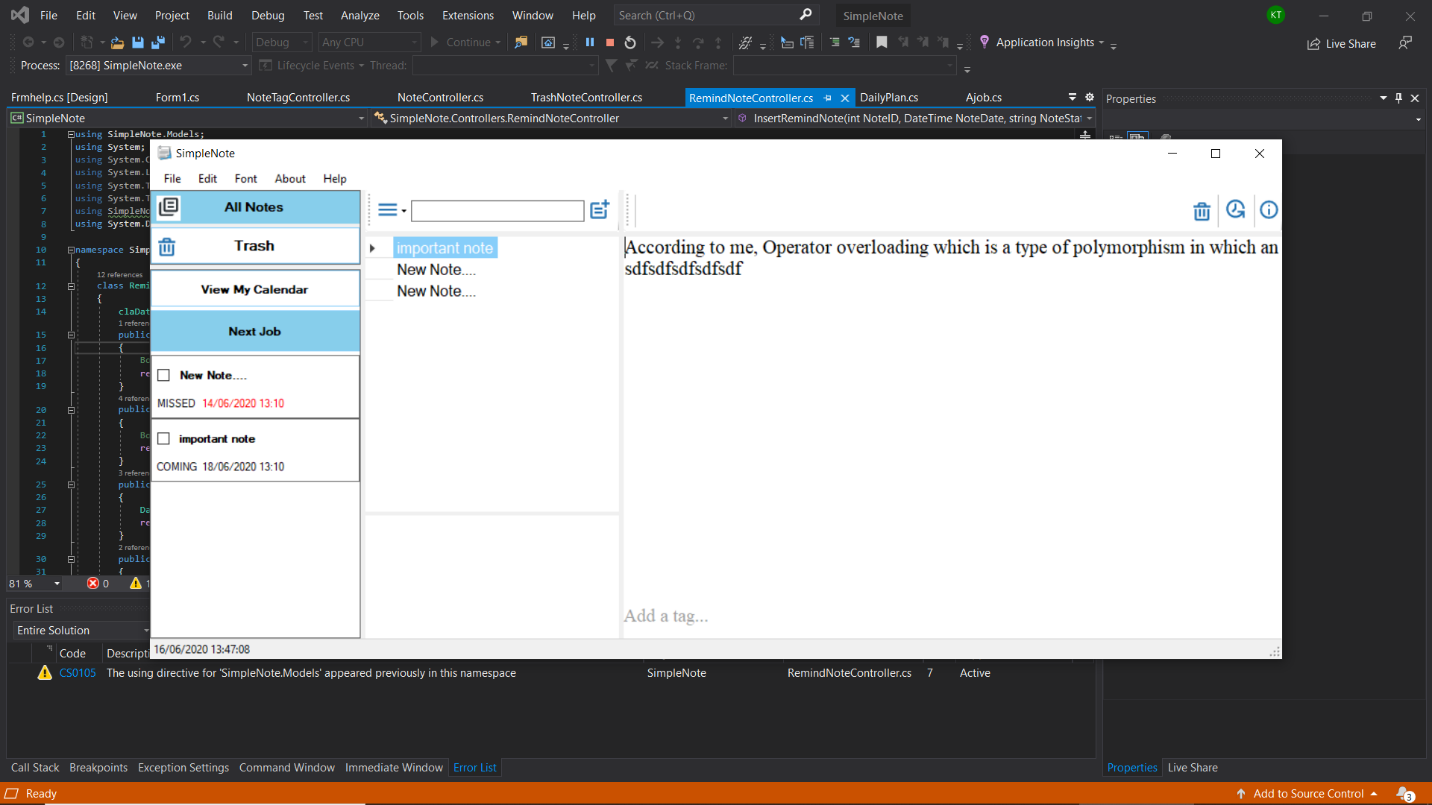


Picture 9: show help form



Picture10:test dialog save as

Picture11 : test open dialog



Picture 12 : Test change Note’s Name

# 

# **Chapter 5: Group’s conclusion**

1. **Conclusion**

The team commented that the software has reached almost 100% of the requirements and features a simple note software should have.

* ***Advantages:***

The interface is neat, elegant and accessible.

Many useful features

Capacity quite light.

The program runs stably, no errors or crashes occur during use.

Do not heat the appliance.

* ***Defect:***

Must enable database when using.

1. **Development**

* We want to develop simple note software on phone OS platforms like Android or IOS.
* Connect to school’s database and help student manage schedule time and teacher to announce student of unexpected day-off or changed class.