

BÁO CÁO BÀI TẬP LỚN: CHƯƠNG TRÌNH WORKPAD

Môn: Lập trình giao diện

Gồm 2 sinh viên thực hiện:

Họ và tên	MSSV
Đỗ Nhất Duy.	2251050019
Trình Quốc Đạt.	2251050016

I/ GIỚI THIỆU

Bài báo cáo này tập trung trình bày quá trình phát triển và triển khai chương trình WordPad đơn giản sử dụng ngôn ngữ lập trình C#. Chương trình này nhằm mục đích cung cấp công cụ hữu ích cho người dùng soạn thảo văn bản một cách nhanh chóng và thuận lợi.

II/ MỤC TIÊU

Về mục tiêu của bài tập lớn này:

1. Tăng cường hiệu quả làm việc
2. Tối ưu hóa quy trình làm việc
3. Cải thiện sự cộng tác
4. Tăng tính minh bạch
5. Quản lý tài nguyên hiệu quả
6. Tích hợp và tùy chỉnh

III/ CÁC CHỨC NĂNG CHÍNH

Giao diện:

1. **CursorsHand:** Hiện thị con trỏ chuột hình bàn tay khi rê chuột qua các thành phần tương tác.
 - o **Cách làm:** Sử dụng thuộc tính Cursor
 - o **Bước làm:**
 1. Chọn control cần thay đổi con trỏ chuột.
 2. Trong cửa sổ Properties, tìm đến thuộc tính Cursor.
 3. Chọn giá trị Hand từ danh sách.

Thao tác với File:

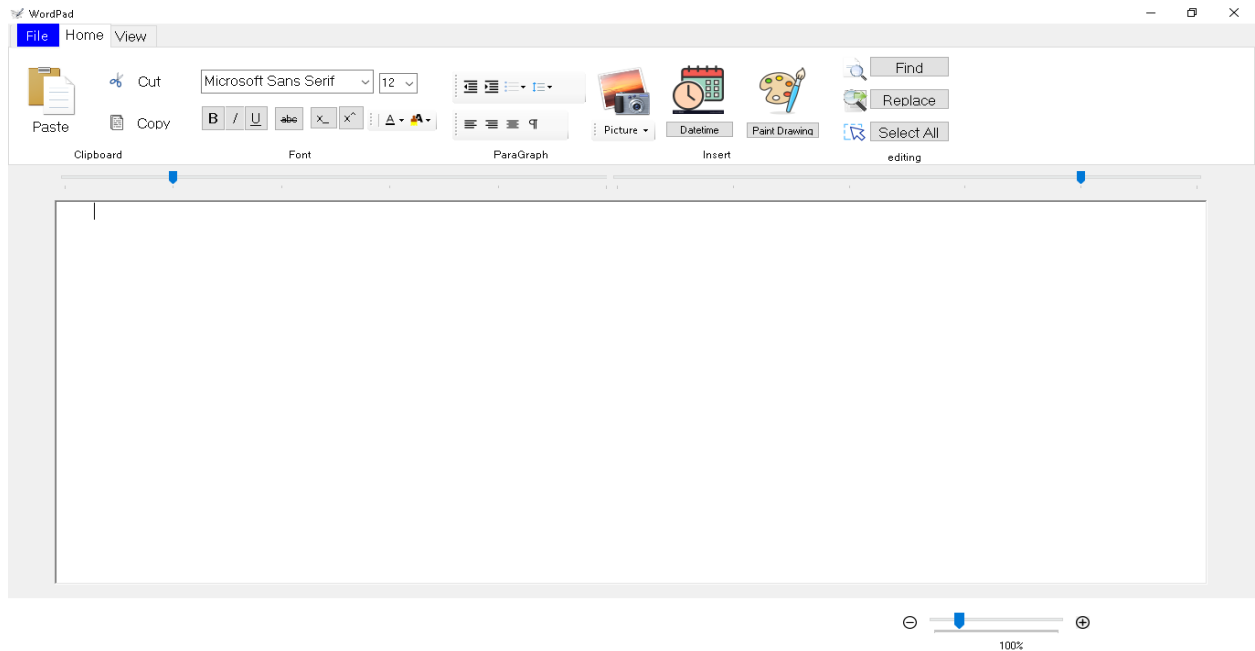
1. **Open:** Mở file văn bản đã có sẵn.
 - o **Cách làm:** Sử dụng OpenFileDialog để mở hộp thoại chọn file.
 - o **Bước làm:**
 1. Tạo một đối tượng OpenFileDialog.
 2. Thiết lập thuộc tính Filter để chỉ định loại file muốn mở.
 3. Gọi phương thức ShowDialog() để hiển thị hộp thoại.

4. Nếu người dùng chọn file và nhấn OK, đọc nội dung file bằng File.ReadAllText() và hiển thị lên RichTextBox.
2. **Tạo mới Wordpad:** Tạo file văn bản trống mới.
 - **Cách làm:** Xóa nội dung của RichTextBox.
 - **Bước làm:** Gán giá trị rỗng " " cho thuộc tính Text của RichTextBox.
3. **Lưu file Wordpad:** Lưu nội dung văn bản vào file.
 - **Cách làm:** Sử dụng SaveFileDialog để mở hộp thoại lưu file.
 - **Bước làm:**
 1. Tạo một đối tượng SaveFileDialog.
 2. Thiết lập thuộc tính Filter để chỉ định loại file muốn lưu.
 3. Gọi phương thức ShowDialog() để hiển thị hộp thoại.
 4. Nếu người dùng chọn vị trí, đặt tên file và nhấn OK, ghi nội dung của RichTextBox vào file bằng File.WriteAllText().
4. **In file Wordpad:** In nội dung văn bản.
 - **Cách làm:** Sử dụng PrintDialog và PrintDocument để in.
 - **Bước làm:**
 1. Tạo một đối tượng PrintDialog và liên kết với PrintDocument.
 2. Gọi phương thức ShowDialog() của PrintDialog để hiển thị hộp thoại cấu hình in ấn.
 3. Nếu người dùng nhấn OK, gọi phương thức Print() của PrintDocument để bắt đầu in.
 4. Xử lý sự kiện PrintPage của PrintDocument để vẽ nội dung của RichTextBox lên từng trang.

Định dạng Văn bản:

1. **Chữ nhỏ ở trên:** Biến đổi văn bản thành chữ nhỏ nằm phía trên dòng.
 - **Cách làm:** Thay đổi SelectionCharOffset và SelectionFont của RichTextBox.
 - **Bước làm:**
 1. Lấy kích thước font hiện tại từ cbSize.SelectedItem.
 2. Thiết lập rtbText.SelectionCharOffset bằng một nửa kích thước font.
 3. Tạo font mới với kích thước bằng một nửa kích thước font hiện tại và gán cho rtbText.SelectionFont.
2. **Chữ nhỏ ở dưới:** Biến đổi văn bản thành chữ nhỏ nằm phía dưới dòng.
 - **Cách làm:** Tương tự như "Chữ nhỏ ở trên", nhưng thay đổi giá trị SelectionCharOffset.
 - **Bước làm:**
 1. Lấy kích thước font hiện tại từ cbSize.SelectedItem.
 2. Thiết lập rtbText.SelectionCharOffset bằng giá trị âm của một nửa kích thước font.

3. Tạo font mới với kích thước bằng một nửa kích thước font hiện tại và gán cho `rtbText.SelectionFont`.
3. **Gạch chữ:** Thêm gạch ngang qua giữa văn bản.
 - **Cách làm:** Sử dụng toán tử XOR để bật/tắt kiểu chữ `Strikeout` cho `SelectionFont` của `RichTextBox`.
 - **Bước làm:**
 1. Lấy font hiện tại của văn bản được chọn từ `rtbText.SelectionFont`.
 2. Áp dụng toán tử XOR với `FontStyle.Strikeout` để tạo kiểu chữ mới.
 3. Gán font mới cho `rtbText.SelectionFont`.
4. **In đậm:** Làm cho văn bản đậm hơn.
 - **Cách làm:** Tương tự như "Gạch chữ", nhưng sử dụng `FontStyle.Bold`.
5. **In nghiêng:** Làm cho văn bản nghiêng.
 - **Cách làm:** Tương tự như "Gạch chữ", nhưng sử dụng `FontStyle.Italic`.
6. **Gạch dưới:** Thêm gạch chân dưới văn bản.
 - **Cách làm:** Tương tự như "Gạch chữ", nhưng sử dụng `FontStyle.Underline`.
7. **Tạo Font chữ:** Cho phép người dùng chọn font chữ.
 - **Cách làm:** Sử dụng `ComboBox` để hiển thị danh sách font chữ.
 - **Bước làm:**
 1. Lấy danh sách font từ `FontFamily.Families` và thêm vào `ComboBox`.
 2. Trong sự kiện `SelectedIndexChanged` của `ComboBox`, tạo font mới với tên font được chọn và gán cho `rtbText.SelectionFont`.
 3. Xử lý trường hợp người dùng nhập font không hợp lệ bằng cách sử dụng font mặc định "Times New Roman".
8. **Tạo Kích thước chữ:** Cho phép người dùng chọn kích thước font.
 - **Cách làm:** Sử dụng `ComboBox` để hiển thị danh sách kích thước font.
 - **Bước làm:**
 1. Thêm danh sách các kích thước font vào `ComboBox`.
 2. Trong sự kiện `SelectedIndexChanged` của `ComboBox`, tạo font mới với kích thước được chọn và gán cho `rtbText.SelectionFont`.
 3. Xử lý trường hợp người dùng nhập kích thước không hợp lệ bằng cách sử dụng kích thước mặc định 12.



Thao tác Văn bản:

1. **Copy:** Sao chép văn bản vào bộ nhớ tạm.
 - **Cách làm:** Sử dụng Clipboard.SetText().
 - **Bước làm:**
 1. Lấy văn bản được chọn từ `rtbText.SelectedText`.
 2. Gọi phương thức `Clipboard.SetText()` với văn bản được chọn làm tham số.
2. **Cut:** Di chuyển văn bản vào bộ nhớ tạm.
 - **Cách làm:** Kết hợp "Copy" và xóa văn bản đã chọn.
 - **Bước làm:**
 1. Thực hiện "Copy".
 2. Gán giá trị rỗng "" cho `rtbText.SelectedText` để xóa văn bản đã chọn.
3. **Paste:** Chèn nội dung từ bộ nhớ tạm vào văn bản.
 - **Cách làm:** Sử dụng `Clipboard.GetText()` và gán cho `rtbText.SelectedText`.
4. **Căn lề trái:** Căn lề trái cho văn bản.
 - **Cách làm:** Gán `HorizontalAlignment.Left` cho `rtbText.SelectionAlignment`.
5. **Căn lề phải:** Căn lề phải cho văn bản.
 - **Cách làm:** Gán `HorizontalAlignment.Right` cho `rtbText.SelectionAlignment`.
6. **Căn giữa:** Căn giữa cho văn bản.
 - **Cách làm:** Gán `HorizontalAlignment.Center` cho `rtbText.SelectionAlignment`.

Màu sắc:

1. **Tạo FlowLayout để chọn màu chữ cho text:** Hiển thị bảng màu cho người dùng chọn màu chữ.
 - **Cách làm:** Sử dụng FlowLayoutPanel để sắp xếp các button màu sắc.
 - **Bước làm:**
 1. Tạo một FlowLayoutPanel và thiết lập FlowDirection là TopDown hoặc LeftToRight để sắp xếp button.
 2. Tạo các button với màu sắc khác nhau và thêm vào FlowLayoutPanel.
 3. Trong sự kiện Click của mỗi button, gán màu sắc của button cho richTextBox.SelectionColor.
2. **Tạo FlowLayout để chọn màu nền chữ cho text:** Hiển thị bảng màu cho người dùng chọn màu nền.
 - **Cách làm:** Tương tự như "Tạo FlowLayout để chọn màu chữ cho text", nhưng gán màu sắc cho richTextBox.SelectionBackColor.
3. **Mở full bảng màu để tô:** Mở hộp thoại ColorDialog cho phép người dùng chọn màu chi tiết hơn.
 - **Cách làm:** Sử dụng ColorDialog.
 - **Bước làm:**
 1. Tạo một đối tượng ColorDialog.
 2. Thiết lập thuộc tính FullOpen là true để hiển thị bảng màu đầy đủ.
 3. Gọi phương thức ShowDialog().
 4. Nếu người dùng chọn màu và nhấn OK, gán màu được chọn cho richTextBox.SelectionColor.

Thao tác Hệ thống:

1. **Thoát chương trình:** Đóng ứng dụng.
 - **Cách làm:** Sử dụng phương thức Close() của form.
 - **Bước làm:**
 1. Hiển thị hộp thoại xác nhận hỏi người dùng có muốn thoát hay không.
 2. Nếu người dùng đồng ý, gọi phương thức Close() để đóng form và kết thúc chương trình.

Định dạng Đoạn Văn:

1. **Tạo đoạn văn bản mới:** Thêm dòng trống mới.
 - **Cách làm:** Chèn Environment.NewLine vào RichTextBox.
2. **Tăng giá trị thụt dòng:** Tăng thụt đầu dòng.
 - **Cách làm:** Tăng giá trị thuộc tính richTextBox.SelectionIndent.
3. **Giảm giá trị thụt dòng:** Giảm thụt đầu dòng.
 - **Cách làm:** Giảm giá trị thuộc tính richTextBox.SelectionIndent.

Zoom:

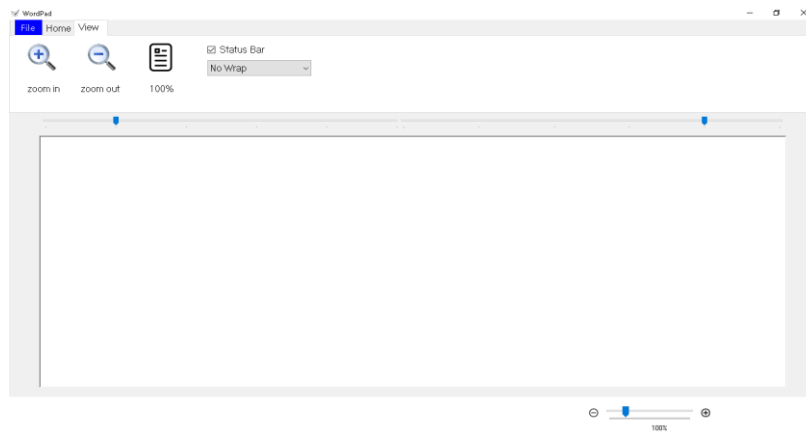
1. **Phóng to:** Tăng kích thước hiển thị của văn bản.

- **Cách làm:** Thay đổi SelectionFont của RichTextBox dựa trên giá trị của TrackBar.
- **Bước làm:**
 1. Trong sự kiện ValueChanged của TrackBar, lấy giá trị hiện tại của TrackBar.
 2. Tính toán kích thước font mới dựa trên giá trị của TrackBar.
 3. Tạo font mới với kích thước đã tính toán và gán cho rtbText.SelectionFont.

2. **Thu nhỏ:** Giảm kích thước hiển thị của văn bản.

- **Cách làm:** Tương tự như "Phóng to", nhưng tính toán kích thước font nhỏ hơn.

Thanh Status Bar:



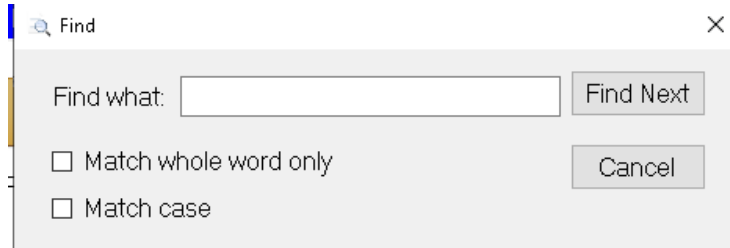
1. **Ẩn hiện thanh status bar:** Ẩn/hiện thanh trạng thái.
 - **Cách làm:** Thay đổi thuộc tính Visible của Panel chứa thanh trạng thái.
 - **Bước làm:**
 1. Trong sự kiện CheckedChanged của CheckBox điều khiển hiển thị thanh trạng thái, gán true hoặc false cho thuộc tính Visible của Panel dựa trên trạng thái của CheckBox.
2. **Trở về kích thước 100%:** Đặt kích thước font về giá trị ban đầu.
 - **Cách làm:** Tạo font mới với kích thước ban đầu và gán cho rtbText.SelectionFont.
 - **Bước làm:** Tạo font mới với kích thước mặc định (sizeChu = 12) và gán cho rtbText.SelectionFont.

Khác:

1. **Bật tắt chức năng wrap text:** Bật/tắt tự động xuống dòng.
 - **Cách làm:** Thay đổi thuộc tính WordWrap của RichTextBox.
 - **Bước làm:** Trong sự kiện SelectedIndexChanged của ComboBox điều khiển wrap text, gán true hoặc false cho thuộc tính rtbText.WordWrap dựa trên lựa chọn của người dùng.
2. **Thêm hình ảnh:** Chèn hình ảnh từ file vào văn bản.
 - **Cách làm:** Sử dụng OpenFileDialog để chọn file hình ảnh và chèn vào RichTextBox.
 - **Bước làm:**

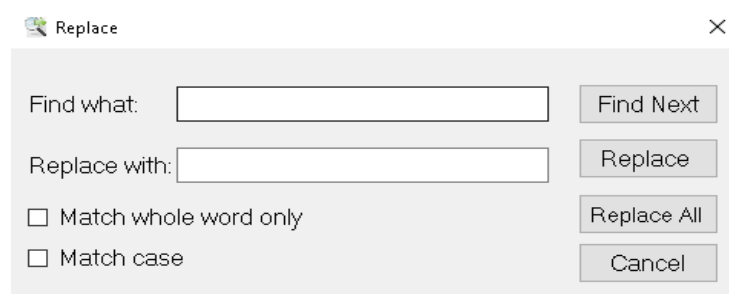
1. Tạo một đối tượng OpenFileDialog và thiết lập Filter để chỉ định loại file hình ảnh.
2. Gọi phương thức ShowDialog().
3. Nếu người dùng chọn file và nhấn OK, đọc hình ảnh từ file bằng Image.FromFile().
4. Sao chép hình ảnh vào Clipboard bằng Clipboard.SetImage().
5. Dán hình ảnh từ Clipboard vào RichTextBox bằng rtbText.Paste().

3. **Tìm kiếm:** Tìm kiếm văn bản.



- **Cách làm:** Sử dụng rtbText.Find() và các tùy chọn tìm kiếm.
- **Bước làm:**
 1. Lấy văn bản cần tìm từ TextBox.
 2. Gọi phương thức rtbText.Find() với văn bản cần tìm và các tùy chọn tìm kiếm (phân biệt chữ hoa/thường, tìm kiếm chính xác cả từ).
 3. Nếu tìm thấy, rtbText.Find() trả về vị trí bắt đầu của văn bản tìm thấy, nếu không trả về -1.
 4. Di chuyển con trỏ đến vị trí tìm thấy và tô màu đoạn văn bản đó.

4. **Thay thế:** Thay thế văn bản được chọn hoặc tất cả các lần xuất hiện.



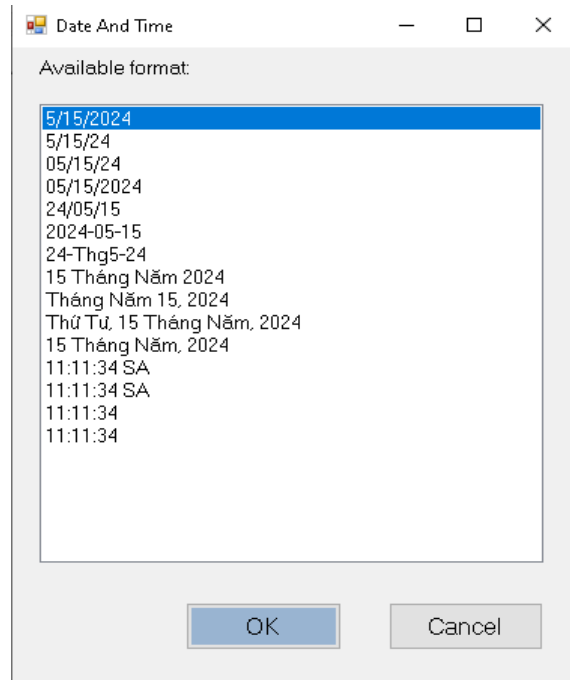
- **Cách làm:** Sử dụng rtbText.SelectedText hoặc vòng lặp để thay thế văn bản.
- **Bước làm:**
 1. Lấy văn bản cần tìm và văn bản thay thế từ TextBox.
 2. Sử dụng rtbText.SelectedText = newText để thay thế văn bản được chọn.
 3. Để thay thế tất cả các lần xuất hiện, sử dụng vòng lặp while và rtbText.IndexOf() để tìm kiếm vị trí của văn bản cần thay thế.

4. Sử dụng `String.Substring()` để tạo chuỗi mới với văn bản đã được thay thế.
5. Cập nhật nội dung của `RichTextBox` với chuỗi mới.

5. **Chọn toàn bộ văn bản:** Chọn toàn bộ văn bản.

- **Cách làm:** Sử dụng `rtbText.SelectAll()`.

6. **Thêm thời gian:** Chèn thời gian hiện tại vào văn bản.



- **Cách làm:** Sử dụng `DateTime.Now` và định dạng thời gian.

- **Bước làm:**

1. Lấy thời gian hiện tại từ `DateTime.Now`.
2. Cho phép người dùng chọn định dạng thời gian từ `ListBox`.
3. Chuyển đổi thời gian sang chuỗi theo định dạng đã chọn.
4. Chèn chuỗi thời gian vào `RichTextBox`.

7. **Dẫn dòng:** Điều chỉnh khoảng cách giữa các dòng.

- **Cách làm:** Thay đổi `rtbText.SelectionCharOffset`.

- **Bước làm:**

1. Cho phép người dùng chọn khoảng cách dẫn dòng từ menu `ToolStripMenuItem`.
2. Gán giá trị khoảng cách đã chọn cho `rtbText.SelectionCharOffset`.

8. **Vẽ:** Mở công cụ vẽ.



- **Cách làm:** Tạo một form mới cho công cụ vẽ.
- **Bước làm:**
 1. Tạo một form mới với các công cụ vẽ, bảng màu, lựa chọn kích thước nét vẽ.
 2. Cho phép người dùng vẽ trên form mới.
 3. Khi người dùng đóng form vẽ, lấy hình ảnh đã vẽ và chèn vào RichTextBox của form chính.

9. **Căn chỉnh lề trái:** Điều chỉnh lề trái.

- **Cách làm:** Thay đổi `rtbText.SelectionIndent`.
- **Bước làm:**
 1. Sử dụng `TrackBar` để cho phép người dùng điều chỉnh lề trái.
 2. Trong sự kiện `Scroll` của `TrackBar`, gán giá trị hiện tại của `TrackBar` cho `rtbText.SelectionIndent`.

10. **Căn chỉnh lề phải:** Điều chỉnh lề phải.

- **Cách làm:** Thay đổi `rtbText.SelectionRightIndent`.
- **Bước làm:**
 1. Sử dụng `TrackBar` để cho phép người dùng điều chỉnh lề phải.
 2. Trong sự kiện `Scroll` của `TrackBar`, gán giá trị hiện tại của `TrackBar` cho `rtbText.SelectionRightIndent`.

IV/ VỀ PHẦN NỘI DUNG CODE

Nhất Duy Làm các chức năng thao tác với File (Open, new Wordpad, Save, print), thao tác văn bản (copy, Cut, Paste, Căn lề trái, căn lề Phải, căn giữa). Màu sắc (Tạo `FlowLayout` để chọn màu chữ cho text, Tạo `FlowLayout` để chọn màu nền chữ cho text, Mở full bảng màu để tô). Thao tác Hệ Thống (Exit). Định Dạng đoạn văn (Tạo đoạn văn bản mới, tăng giá trị thụt dòng, Giảm giá trị thụt dòng). Thêm Hình ảnh, Tìm kiếm, thay thế đoạn văn bản. vẽ...

Quốc Đạt làm các chức năng Định dạng văn bản (Chữ nhỏ trên, chữ nhỏ dưới, Gạch CHữ, In nghiêng, Gạch dưới, tạo Font chữ, size chữ). Chức Năng ZOOM (phóng to, thu nhỏ), Thanh Status bar (Ẩn Hiện, trả về kích thước ban đầu), Bật tắt chức năng wordrap. Thêm thời gian, giãn dòng, căn chỉnh lề...

Giao Diện cả 2 phối hợp thực hiện, xử lý logic, kiểm tra lỗi và tối ưu hóa chương trình.

V/ TỔNG KẾT

Chương trình WordPad là bài tập đơn giản nhưng mang lại hiệu quả cao. Việc phát triển chương trình giúp chúng em củng cố kiến thức về lập trình giao diện C#, mà còn hiểu rõ hơn về sự phát triển của phần mềm. Trong tương lai, chúng em mong muốn được phát triển chương trình bằng cách thêm các tính năng mới và nâng cấp giao diện người dùng để có thể mang đến một chương trình tuyệt vời và hữu ích. Trong tương lai chúng em muốn phát triển chương trình word có những tính năng cao hơn và thông dụng hơn.