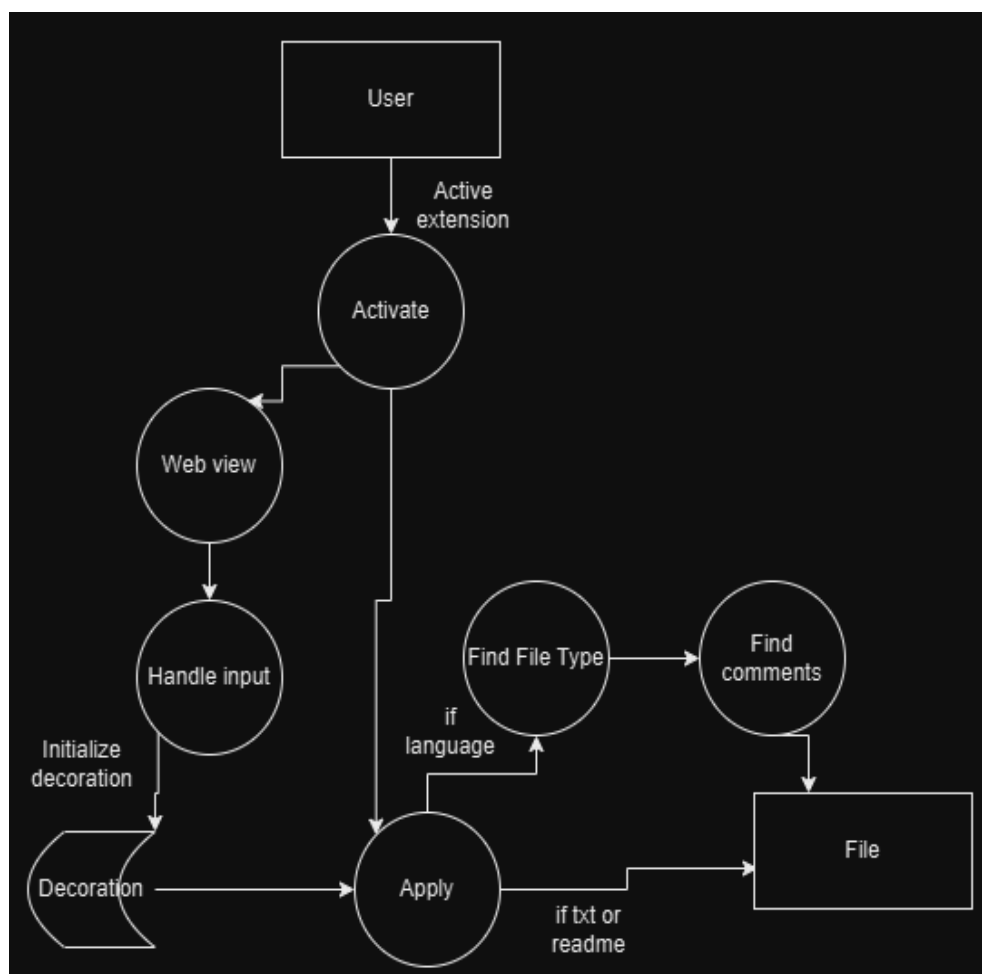
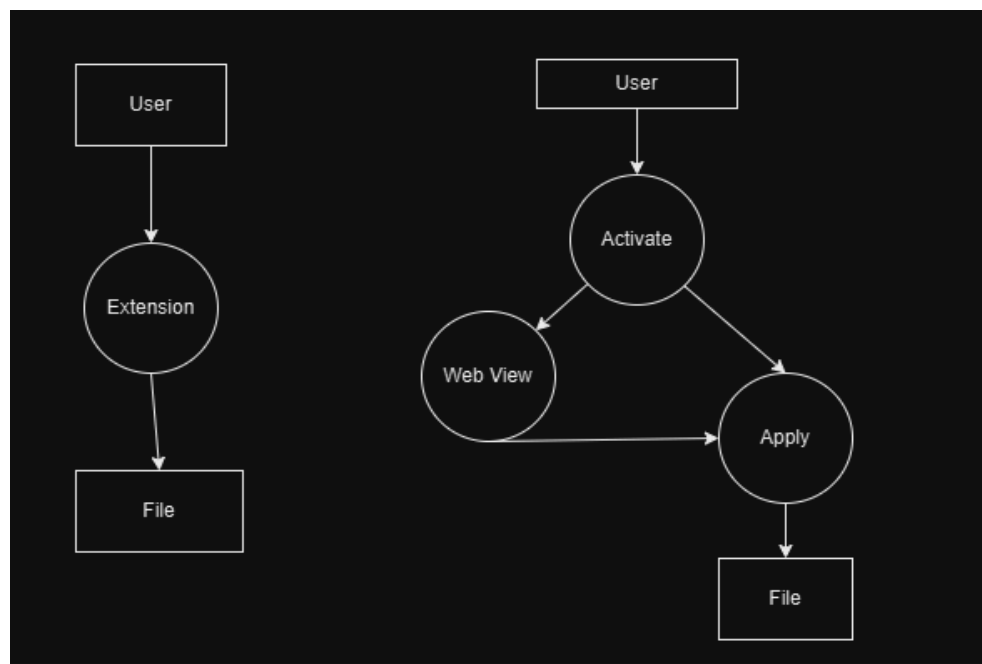


DFD



1. تابع `activate`

- نقش در دیاگرام: بخش **Activate** در دیاگرام.
 - توضیح عملکرد: این تابع زمانی که افزونه توسط کاربر فعال می‌شود، اجرا می‌شود. در این تابع، تزئینات اولیه شامل رنگ، شفافیت، و سایر تنظیمات برای متن تعریف می‌شوند. همچنین، دو نوع تزئین ایجاد می‌شود:
 - `colorDecorationType`: برای تزئینات با رنگ خاص.
 - `fadedDecorationType`: برای تزئینات محو با شفافیت خاص.
-

2. تابع `applyDecorations`

- نقش در دیاگرام: بخش **Apply** در دیاگرام.
 - توضیح عملکرد: این تابع به عنوان اصلی‌ترین بخش برای اعمال تزئینات استفاده می‌شود.
`applyDecorations` کل متن را پردازش کرده و بسته به نوع فایل، نواحی خاصی از متن را به صورت رنگی یا محو مشخص می‌کند. این فرآیند به دو حالت عمل می‌کند:
 - فایل‌های متنی (مانند **txt** یا **README**): در این حالت، کلمات مستقیماً تزئین می‌شوند.
 - فایل‌های کد: ابتدا نوع فایل و زبان شناسایی می‌شود و سپس نواحی کامنت‌ها تزئین می‌گردند.
-

3. تابع `guessFileType`

- نقش در دیاگرام: بخش **Find File Type** در دیاگرام.
 - توضیح عملکرد: این تابع نوع فایل و زبان برنامه‌نویسی سند را شناسایی می‌کند و الگوهای کامنت‌ها را برای زبان‌های مختلف (مانند `//` در جاوااسکریپت یا `#` در پایتون) مشخص می‌کند. خروجی این تابع شامل نواحی کامنت است که برای تزئین استفاده می‌شود.
-

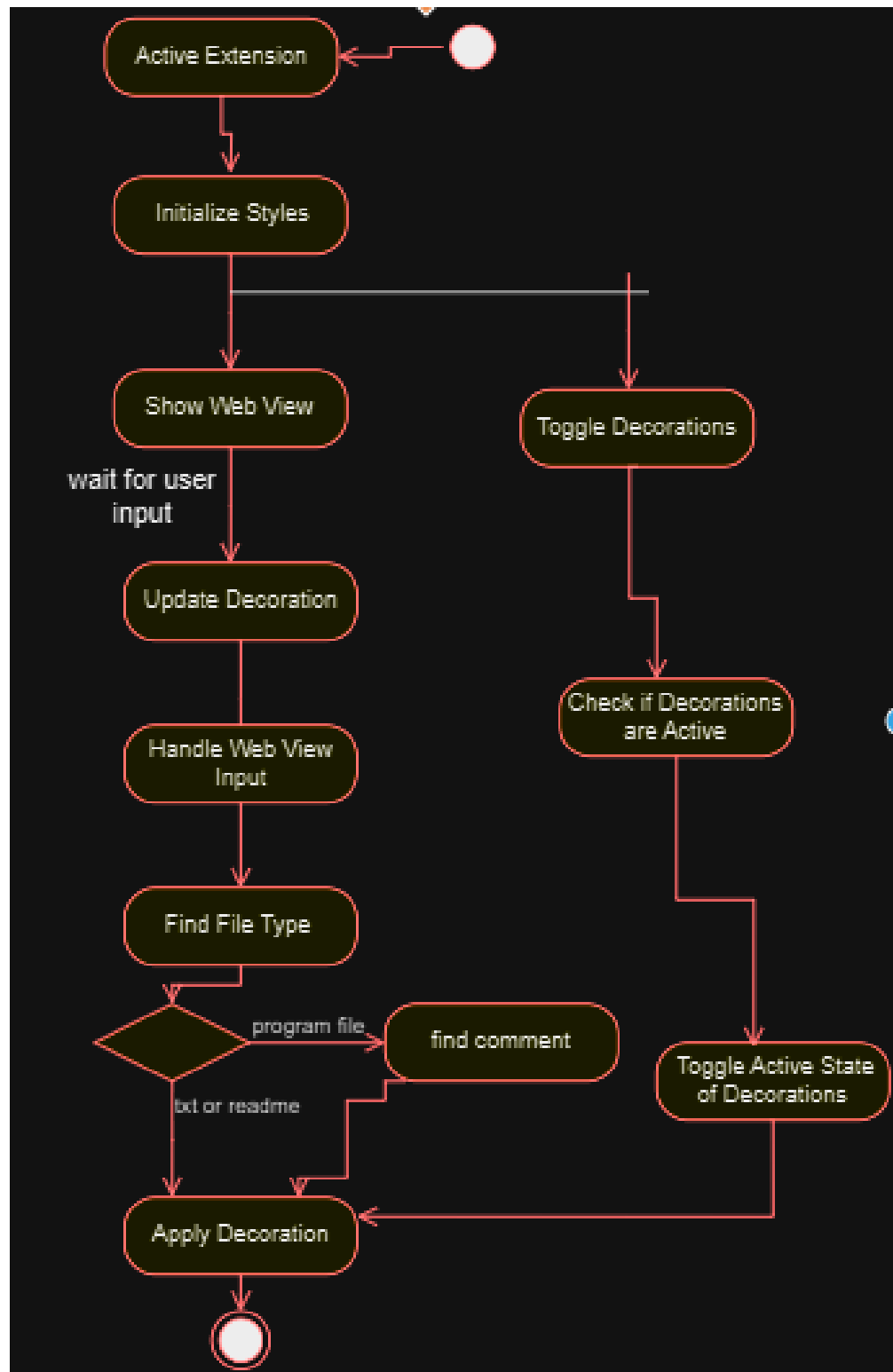
4. تابع `findComments`

- نقش در دیاگرام: بخش **Find Comments** در دیاگرام.
 - توضیح عملکرد: این تابع الگوهای کامنت را در متن سند پیدا کرده و موقعیت آن‌ها را در یک آرایه ذخیره می‌کند. این اطلاعات به تابع `applyDecorations` ارسال می‌شود تا تزئینات بر اساس نواحی کامنت‌ها اعمال شود.
-

5. تابع `showWebView` و `handleWebViewInput`

- نقش در دیاگرام: بخش **Web View** و **Handle Input**.
- توضیح عملکرد:
 - `showWebView`: این تابع یک رابط وب‌ویو ایجاد می‌کند که به کاربر اجازه می‌دهد تنظیمات رنگ، پس‌زمینه، و شفافیت را تغییر دهد.
 - `handleWebViewInput`: این تابع ورودی‌های کاربر از وب‌ویو را پردازش می‌کند و تزئینات جدید را بر اساس تنظیمات وارد شده اعمال می‌کند.

Activity Diagram



Sequence Diagram

