

```
from kivy.config import Config
Config.set('kivy', 'video', 'ffpyplayer')
from kivy.core.window import Window
from kivy.uix.boxlayout import BoxLayout
from kivy.uix.label import Label
from kivy.uix.textinput import TextInput
from kivy.uix.scrollview import ScrollView
from kivy.uix.button import Button
from kivy.uix.popup import Popup
from kivy.uix.image import Image
from kivymd.uix.button import MDIconButton
from kivymd.app import MDApp
from kivy.uix.screenmanager import ScreenManager, Screen
from kivy.graphics import Color, RoundedRectangle
from pdf2image import convert_from_path
import os
```

```
Window.size = (1000 / 2.5, 1920 / 2.5)
Window.clearcolor = (1, 1, 1, 1)
```

```
# Écran lecteur PDF
class PDFReaderScreen(Screen):
    def __init__(self, **kwargs):
        super().__init__(**kwargs)
        self.current_page = 0
        self.pdf_pages = []
        self.pdf_folder = "assets/" # Dossier contenant les PDF
        self.add_widget(self.build_pdf_viewer())
```

```
def build_pdf_viewer(self):
    layout = BoxLayout(orientation='vertical', spacing=10, padding=10)
    self.image_display = Image(size_hint=(1, 0.9))
```

```
button_layout = BoxLayout(size_hint=(1, 0.1), spacing=10)
prev_button = Button(text="⬅️ Précédent", on_press=self.prev_page)
next_button = Button(text="➡️ Suivant", on_press=self.next_page)
```

```
button_layout.add_widget(prev_button)
button_layout.add_widget(next_button)
```

```
layout.add_widget(self.build_pdf_list())
layout.add_widget(self.image_display)
layout.add_widget(button_layout)
return layout
```

```
def build_pdf_list(self):
    pdf_list_layout = BoxLayout(orientation='vertical', size_hint=(1, 0.3), spacing=5)
    scroll = ScrollView(size_hint=(1, None), height=150)
    pdf_files = [f for f in os.listdir(self.pdf_folder) if f.endswith('.pdf')]
```

```

for pdf in pdf_files:
    btn = Button(text=pdf, size_hint_y=None, height=40)
    btn.bind(on_press=lambda instance, p=pdf: self.load_pdf(os.path.join(self.pdf_folder, p)))
    pdf_list_layout.add_widget(btn)

```

```

scroll.add_widget(pdf_list_layout)
return scroll

```

```

def load_pdf(self, pdf_path):
    self.pdf_pages = convert_from_path(pdf_path)
    self.current_page = 0
    self.display_page()

```

```

def display_page(self):
    if self.pdf_pages:
        self.pdf_pages[self.current_page].save("temp_page.png", 'PNG')
        self.image_display.source = "temp_page.png"
        self.image_display.reload()

```

```

def prev_page(self, instance):
    if self.current_page > 0:
        self.current_page -= 1
        self.display_page()

```

```

def next_page(self, instance):
    if self.current_page < len(self.pdf_pages) - 1:
        self.current_page += 1
        self.display_page()

```

```

# Écran principal
class ElectMainScreen(BoxLayout):
    def __init__(self, **kwargs):
        super().__init__(**kwargs)
        self.orientation = 'vertical'
        self.screen_manager = ScreenManager()
        self.screen_manager.add_widget(Screen(name='home'))
        self.screen_manager.add_widget(PDFReaderScreen(name='pdf_reader'))
        self.add_widget(self.screen_manager)
        self.add_widget(self.build_bottom_nav())

```

```

def build_bottom_nav(self):
    bottom_nav = BoxLayout(size_hint_y=None, spacing=35, height=80)
    bottom_nav.add_widget(MDIconButton(icon="home", on_press=lambda x:
        self.change_screen('home')))
    bottom_nav.add_widget(MDIconButton(icon="file", on_press=lambda x:
        self.change_screen('pdf_reader')))
    return bottom_nav

```

```

def change_screen(self, screen_name):
    self.screen_manager.current = screen_name

```

```
class Elect241App(MDApp):
    def build(self):
        self.icon = 'assets/900.png'
        return ElectMainScreen()

if __name__ == '__main__':
    Elect241App().run()
```