

**BUKU
PANDUAN PENGGUNAAN MODEL**

MUHAMMAD RAIHAN RAMADHAN (10121899)

**PREDIKSI KINERJA JAM ATOM CESIUM
MENGUNAKAN ALGORITMA LSTM**

**FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
JURUSAN SISTEM INFORMASI
2025**

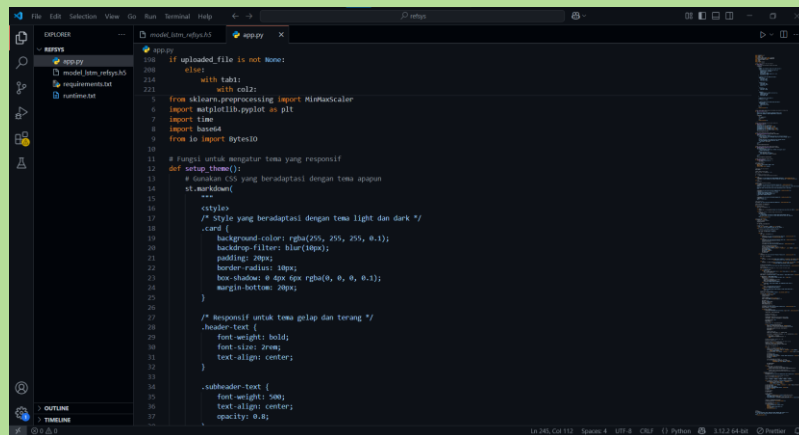
DAFTAR ISI

DAFTAR ISI	1
PETUNJUK PENGGUNAAN.....	2

PETUNJUK PENGGUNAAN

1. Membuka Folder refsys di VSCODE dan pastikan sudah menginstall modul dengan menggunakan perintah berikut :

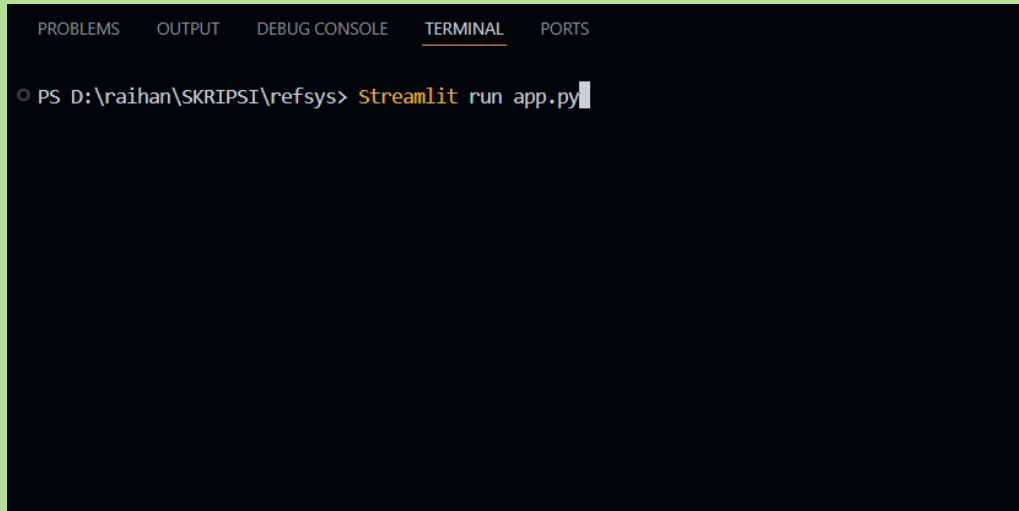
'pip install streamlit pandas numpy matplotlib scikit-learn tensorflow'



```
1 # Importing the necessary libraries
2 import streamlit as st
3 import pandas as pd
4 import numpy as np
5 from sklearn.preprocessing import MinMaxScaler
6 import matplotlib.pyplot as plt
7 import time
8 import random
9 from io import BytesIO
10
11 # Fungsi untuk mengatur tema yang responsif
12 def setup_theme():
13     # Gunakan CSS yang beradaptasi dengan tema apapun
14     st.markdown(
15         """
16         <style>
17         /* Style yang beradaptasi dengan tema light dan dark */
18         .card {
19             background-color: rgb(255, 255, 255, 0.1);
20             backdrop-filter: blur(10px);
21             padding: 10px;
22             border-radius: 10px;
23             box-shadow: 0 0px 0px rgb(0, 0, 0, 0.1);
24             margin-bottom: 20px;
25         }
26
27         /* Responsif untuk tema gelap dan terang */
28         .header-text {
29             font-weight: bold;
30             font-size: 20px;
31             text-align: center;
32         }
33
34         .subheader-text {
35             font-weight: bold;
36             text-align: center;
37             opacity: 0.8;
38         }
39     </style>
40
41     """
42     )
43
44 # Fungsi untuk menampilkan data yang responsif
45 def display_data():
46     # Simulasi data
47     data = np.random.randn(10, 2)
48     df = pd.DataFrame(data, columns=["X", "Y"])
49
50     # Menampilkan data
51     st.dataframe(df)
52
53 # Fungsi untuk menampilkan grafik yang responsif
54 def display_chart():
55     # Simulasi data
56     data = np.random.randn(10, 2)
57     df = pd.DataFrame(data, columns=["X", "Y"])
58
59     # Menampilkan grafik
60     st.plotly_chart(df)
61
62 # Fungsi untuk menampilkan tombol yang responsif
63 def display_button():
64     # Simulasi data
65     data = np.random.randn(10, 2)
66     df = pd.DataFrame(data, columns=["X", "Y"])
67
68     # Menampilkan tombol
69     st.button("Klik Saya")
70
71 # Fungsi untuk menampilkan progress bar yang responsif
72 def display_progress_bar():
73     # Simulasi data
74     data = np.random.randn(10, 2)
75     df = pd.DataFrame(data, columns=["X", "Y"])
76
77     # Menampilkan progress bar
78     st.progress(50)
79
80 # Fungsi untuk menampilkan status yang responsif
81 def display_status():
82     # Simulasi data
83     data = np.random.randn(10, 2)
84     df = pd.DataFrame(data, columns=["X", "Y"])
85
86     # Menampilkan status
87     st.status("Status Bar")
88
89 # Fungsi untuk menampilkan alert yang responsif
90 def display_alert():
91     # Simulasi data
92     data = np.random.randn(10, 2)
93     df = pd.DataFrame(data, columns=["X", "Y"])
94
95     # Menampilkan alert
96     st.alert("Alert Bar")
97
98 # Fungsi untuk menampilkan modal yang responsif
99 def display_modal():
100     # Simulasi data
101     data = np.random.randn(10, 2)
102     df = pd.DataFrame(data, columns=["X", "Y"])
103
104     # Menampilkan modal
105     st.modal("Modal Bar")
106
107 # Fungsi untuk menampilkan sidebar yang responsif
108 def display_sidebar():
109     # Simulasi data
110     data = np.random.randn(10, 2)
111     df = pd.DataFrame(data, columns=["X", "Y"])
112
113     # Menampilkan sidebar
114     st.sidebar("Sidebar Bar")
115
116 # Fungsi untuk menampilkan footer yang responsif
117 def display_footer():
118     # Simulasi data
119     data = np.random.randn(10, 2)
120     df = pd.DataFrame(data, columns=["X", "Y"])
121
122     # Menampilkan footer
123     st.footer("Footer Bar")
124
125 # Fungsi untuk menampilkan header yang responsif
126 def display_header():
127     # Simulasi data
128     data = np.random.randn(10, 2)
129     df = pd.DataFrame(data, columns=["X", "Y"])
130
131     # Menampilkan header
132     st.header("Header Bar")
133
134 # Fungsi untuk menampilkan subheader yang responsif
135 def display_subheader():
136     # Simulasi data
137     data = np.random.randn(10, 2)
138     df = pd.DataFrame(data, columns=["X", "Y"])
139
140     # Menampilkan subheader
141     st.subheader("Subheader Bar")
142
143 # Fungsi untuk menampilkan text yang responsif
144 def display_text():
145     # Simulasi data
146     data = np.random.randn(10, 2)
147     df = pd.DataFrame(data, columns=["X", "Y"])
148
149     # Menampilkan text
150     st.text("Text Bar")
151
152 # Fungsi untuk menampilkan markdown yang responsif
153 def display_markdown():
154     # Simulasi data
155     data = np.random.randn(10, 2)
156     df = pd.DataFrame(data, columns=["X", "Y"])
157
158     # Menampilkan markdown
159     st.markdown("Markdown Bar")
160
161 # Fungsi untuk menampilkan code yang responsif
162 def display_code():
163     # Simulasi data
164     data = np.random.randn(10, 2)
165     df = pd.DataFrame(data, columns=["X", "Y"])
166
167     # Menampilkan code
168     st.code("Code Bar")
169
170 # Fungsi untuk menampilkan image yang responsif
171 def display_image():
172     # Simulasi data
173     data = np.random.randn(10, 2)
174     df = pd.DataFrame(data, columns=["X", "Y"])
175
176     # Menampilkan image
177     st.image("Image Bar")
178
179 # Fungsi untuk menampilkan video yang responsif
180 def display_video():
181     # Simulasi data
182     data = np.random.randn(10, 2)
183     df = pd.DataFrame(data, columns=["X", "Y"])
184
185     # Menampilkan video
186     st.video("Video Bar")
187
188 # Fungsi untuk menampilkan audio yang responsif
189 def display_audio():
190     # Simulasi data
191     data = np.random.randn(10, 2)
192     df = pd.DataFrame(data, columns=["X", "Y"])
193
194     # Menampilkan audio
195     st.audio("Audio Bar")
196
197 # Fungsi untuk menampilkan file yang responsif
198 def display_file():
199     # Simulasi data
200     data = np.random.randn(10, 2)
201     df = pd.DataFrame(data, columns=["X", "Y"])
202
203     # Menampilkan file
204     st.file("File Bar")
205
206 # Fungsi untuk menampilkan folder yang responsif
207 def display_folder():
208     # Simulasi data
209     data = np.random.randn(10, 2)
210     df = pd.DataFrame(data, columns=["X", "Y"])
211
212     # Menampilkan folder
213     st.folder("Folder Bar")
214
215 # Fungsi untuk menampilkan link yang responsif
216 def display_link():
217     # Simulasi data
218     data = np.random.randn(10, 2)
219     df = pd.DataFrame(data, columns=["X", "Y"])
220
221     # Menampilkan link
222     st.link("Link Bar")
223
224 # Fungsi untuk menampilkan button yang responsif
225 def display_button():
226     # Simulasi data
227     data = np.random.randn(10, 2)
228     df = pd.DataFrame(data, columns=["X", "Y"])
229
230     # Menampilkan button
231     st.button("Button Bar")
232
233 # Fungsi untuk menampilkan form yang responsif
234 def display_form():
235     # Simulasi data
236     data = np.random.randn(10, 2)
237     df = pd.DataFrame(data, columns=["X", "Y"])
238
239     # Menampilkan form
240     st.form("Form Bar")
241
242 # Fungsi untuk menampilkan chart yang responsif
243 def display_chart():
244     # Simulasi data
245     data = np.random.randn(10, 2)
246     df = pd.DataFrame(data, columns=["X", "Y"])
247
248     # Menampilkan chart
249     st.chart("Chart Bar")
250
251 # Fungsi untuk menampilkan table yang responsif
252 def display_table():
253     # Simulasi data
254     data = np.random.randn(10, 2)
255     df = pd.DataFrame(data, columns=["X", "Y"])
256
257     # Menampilkan table
258     st.table("Table Bar")
259
260 # Fungsi untuk menampilkan progress bar yang responsif
261 def display_progress_bar():
262     # Simulasi data
263     data = np.random.randn(10, 2)
264     df = pd.DataFrame(data, columns=["X", "Y"])
265
266     # Menampilkan progress bar
267     st.progress_bar("Progress Bar")
268
269 # Fungsi untuk menampilkan status yang responsif
270 def display_status():
271     # Simulasi data
272     data = np.random.randn(10, 2)
273     df = pd.DataFrame(data, columns=["X", "Y"])
274
275     # Menampilkan status
276     st.status_bar("Status Bar")
277
278 # Fungsi untuk menampilkan alert yang responsif
279 def display_alert():
280     # Simulasi data
281     data = np.random.randn(10, 2)
282     df = pd.DataFrame(data, columns=["X", "Y"])
283
284     # Menampilkan alert
285     st.alert_bar("Alert Bar")
286
287 # Fungsi untuk menampilkan modal yang responsif
288 def display_modal():
289     # Simulasi data
290     data = np.random.randn(10, 2)
291     df = pd.DataFrame(data, columns=["X", "Y"])
292
293     # Menampilkan modal
294     st.modal_bar("Modal Bar")
295
296 # Fungsi untuk menampilkan sidebar yang responsif
297 def display_sidebar():
298     # Simulasi data
299     data = np.random.randn(10, 2)
300     df = pd.DataFrame(data, columns=["X", "Y"])
301
302     # Menampilkan sidebar
303     st.sidebar_bar("Sidebar Bar")
304
305 # Fungsi untuk menampilkan footer yang responsif
306 def display_footer():
307     # Simulasi data
308     data = np.random.randn(10, 2)
309     df = pd.DataFrame(data, columns=["X", "Y"])
310
311     # Menampilkan footer
312     st.footer_bar("Footer Bar")
313
314 # Fungsi untuk menampilkan header yang responsif
315 def display_header():
316     # Simulasi data
317     data = np.random.randn(10, 2)
318     df = pd.DataFrame(data, columns=["X", "Y"])
319
320     # Menampilkan header
321     st.header_bar("Header Bar")
322
323 # Fungsi untuk menampilkan subheader yang responsif
324 def display_subheader():
325     # Simulasi data
326     data = np.random.randn(10, 2)
327     df = pd.DataFrame(data, columns=["X", "Y"])
328
329     # Menampilkan subheader
330     st.subheader_bar("Subheader Bar")
331
332 # Fungsi untuk menampilkan text yang responsif
333 def display_text():
334     # Simulasi data
335     data = np.random.randn(10, 2)
336     df = pd.DataFrame(data, columns=["X", "Y"])
337
338     # Menampilkan text
339     st.text_bar("Text Bar")
340
341 # Fungsi untuk menampilkan markdown yang responsif
342 def display_markdown():
343     # Simulasi data
344     data = np.random.randn(10, 2)
345     df = pd.DataFrame(data, columns=["X", "Y"])
346
347     # Menampilkan markdown
348     st.markdown_bar("Markdown Bar")
349
350 # Fungsi untuk menampilkan code yang responsif
351 def display_code():
352     # Simulasi data
353     data = np.random.randn(10, 2)
354     df = pd.DataFrame(data, columns=["X", "Y"])
355
356     # Menampilkan code
357     st.code_bar("Code Bar")
358
359 # Fungsi untuk menampilkan image yang responsif
360 def display_image():
361     # Simulasi data
362     data = np.random.randn(10, 2)
363     df = pd.DataFrame(data, columns=["X", "Y"])
364
365     # Menampilkan image
366     st.image_bar("Image Bar")
367
368 # Fungsi untuk menampilkan video yang responsif
369 def display_video():
370     # Simulasi data
371     data = np.random.randn(10, 2)
372     df = pd.DataFrame(data, columns=["X", "Y"])
373
374     # Menampilkan video
375     st.video_bar("Video Bar")
376
377 # Fungsi untuk menampilkan audio yang responsif
378 def display_audio():
379     # Simulasi data
380     data = np.random.randn(10, 2)
381     df = pd.DataFrame(data, columns=["X", "Y"])
382
383     # Menampilkan audio
384     st.audio_bar("Audio Bar")
385
386 # Fungsi untuk menampilkan file yang responsif
387 def display_file():
388     # Simulasi data
389     data = np.random.randn(10, 2)
390     df = pd.DataFrame(data, columns=["X", "Y"])
391
392     # Menampilkan file
393     st.file_bar("File Bar")
394
395 # Fungsi untuk menampilkan folder yang responsif
396 def display_folder():
397     # Simulasi data
398     data = np.random.randn(10, 2)
399     df = pd.DataFrame(data, columns=["X", "Y"])
400
401     # Menampilkan folder
402     st.folder_bar("Folder Bar")
403
404 # Fungsi untuk menampilkan link yang responsif
405 def display_link():
406     # Simulasi data
407     data = np.random.randn(10, 2)
408     df = pd.DataFrame(data, columns=["X", "Y"])
409
410     # Menampilkan link
411     st.link_bar("Link Bar")
412
413 # Fungsi untuk menampilkan button yang responsif
414 def display_button():
415     # Simulasi data
416     data = np.random.randn(10, 2)
417     df = pd.DataFrame(data, columns=["X", "Y"])
418
419     # Menampilkan button
420     st.button_bar("Button Bar")
421
422 # Fungsi untuk menampilkan form yang responsif
423 def display_form():
424     # Simulasi data
425     data = np.random.randn(10, 2)
426     df = pd.DataFrame(data, columns=["X", "Y"])
427
428     # Menampilkan form
429     st.form_bar("Form Bar")
430
431 # Fungsi untuk menampilkan chart yang responsif
432 def display_chart():
433     # Simulasi data
434     data = np.random.randn(10, 2)
435     df = pd.DataFrame(data, columns=["X", "Y"])
436
437     # Menampilkan chart
438     st.chart_bar("Chart Bar")
439
440 # Fungsi untuk menampilkan table yang responsif
441 def display_table():
442     # Simulasi data
443     data = np.random.randn(10, 2)
444     df = pd.DataFrame(data, columns=["X", "Y"])
445
446     # Menampilkan table
447     st.table_bar("Table Bar")
448
449 # Fungsi untuk menampilkan progress bar yang responsif
450 def display_progress_bar():
451     # Simulasi data
452     data = np.random.randn(10, 2)
453     df = pd.DataFrame(data, columns=["X", "Y"])
454
455     # Menampilkan progress bar
456     st.progress_bar_bar("Progress Bar")
457
458 # Fungsi untuk menampilkan status yang responsif
459 def display_status():
460     # Simulasi data
461     data = np.random.randn(10, 2)
462     df = pd.DataFrame(data, columns=["X", "Y"])
463
464     # Menampilkan status
465     st.status_bar_bar("Status Bar")
466
467 # Fungsi untuk menampilkan alert yang responsif
468 def display_alert():
469     # Simulasi data
470     data = np.random.randn(10, 2)
471     df = pd.DataFrame(data, columns=["X", "Y"])
472
473     # Menampilkan alert
474     st.alert_bar_bar("Alert Bar")
475
476 # Fungsi untuk menampilkan modal yang responsif
477 def display_modal():
478     # Simulasi data
479     data = np.random.randn(10, 2)
480     df = pd.DataFrame(data, columns=["X", "Y"])
481
482     # Menampilkan modal
483     st.modal_bar_bar("Modal Bar")
484
485 # Fungsi untuk menampilkan sidebar yang responsif
486 def display_sidebar():
487     # Simulasi data
488     data = np.random.randn(10, 2)
489     df = pd.DataFrame(data, columns=["X", "Y"])
490
491     # Menampilkan sidebar
492     st.sidebar_bar_bar("Sidebar Bar")
493
494 # Fungsi untuk menampilkan footer yang responsif
495 def display_footer():
496     # Simulasi data
497     data = np.random.randn(10, 2)
498     df = pd.DataFrame(data, columns=["X", "Y"])
499
500     # Menampilkan footer
501     st.footer_bar_bar("Footer Bar")
502
503 # Fungsi untuk menampilkan header yang responsif
504 def display_header():
505     # Simulasi data
506     data = np.random.randn(10, 2)
507     df = pd.DataFrame(data, columns=["X", "Y"])
508
509     # Menampilkan header
510     st.header_bar_bar("Header Bar")
511
512 # Fungsi untuk menampilkan subheader yang responsif
513 def display_subheader():
514     # Simulasi data
515     data = np.random.randn(10, 2)
516     df = pd.DataFrame(data, columns=["X", "Y"])
517
518     # Menampilkan subheader
519     st.subheader_bar_bar("Subheader Bar")
520
521 # Fungsi untuk menampilkan text yang responsif
522 def display_text():
523     # Simulasi data
524     data = np.random.randn(10, 2)
525     df = pd.DataFrame(data, columns=["X", "Y"])
526
527     # Menampilkan text
528     st.text_bar_bar("Text Bar")
529
530 # Fungsi untuk menampilkan markdown yang responsif
531 def display_markdown():
532     # Simulasi data
533     data = np.random.randn(10, 2)
534     df = pd.DataFrame(data, columns=["X", "Y"])
535
536     # Menampilkan markdown
537     st.markdown_bar_bar("Markdown Bar")
538
539 # Fungsi untuk menampilkan code yang responsif
540 def display_code():
541     # Simulasi data
542     data = np.random.randn(10, 2)
543     df = pd.DataFrame(data, columns=["X", "Y"])
544
545     # Menampilkan code
546     st.code_bar_bar("Code Bar")
547
548 # Fungsi untuk menampilkan image yang responsif
549 def display_image():
550     # Simulasi data
551     data = np.random.randn(10, 2)
552     df = pd.DataFrame(data, columns=["X", "Y"])
553
554     # Menampilkan image
555     st.image_bar_bar("Image Bar")
556
557 # Fungsi untuk menampilkan video yang responsif
558 def display_video():
559     # Simulasi data
560     data = np.random.randn(10, 2)
561     df = pd.DataFrame(data, columns=["X", "Y"])
562
563     # Menampilkan video
564     st.video_bar_bar("Video Bar")
565
566 # Fungsi untuk menampilkan audio yang responsif
567 def display_audio():
568     # Simulasi data
569     data = np.random.randn(10, 2)
570     df = pd.DataFrame(data, columns=["X", "Y"])
571
572     # Menampilkan audio
573     st.audio_bar_bar("Audio Bar")
574
575 # Fungsi untuk menampilkan file yang responsif
576 def display_file():
577     # Simulasi data
578     data = np.random.randn(10, 2)
579     df = pd.DataFrame(data, columns=["X", "Y"])
580
581     # Menampilkan file
582     st.file_bar_bar("File Bar")
583
584 # Fungsi untuk menampilkan folder yang responsif
585 def display_folder():
586     # Simulasi data
587     data = np.random.randn(10, 2)
588     df = pd.DataFrame(data, columns=["X", "Y"])
589
590     # Menampilkan folder
591     st.folder_bar_bar("Folder Bar")
592
593 # Fungsi untuk menampilkan link yang responsif
594 def display_link():
595     # Simulasi data
596     data = np.random.randn(10, 2)
597     df = pd.DataFrame(data, columns=["X", "Y"])
598
599     # Menampilkan link
600     st.link_bar_bar("Link Bar")
601
602 # Fungsi untuk menampilkan button yang responsif
603 def display_button():
604     # Simulasi data
605     data = np.random.randn(10, 2)
606     df = pd.DataFrame(data, columns=["X", "Y"])
607
608     # Menampilkan button
609     st.button_bar_bar("Button Bar")
610
611 # Fungsi untuk menampilkan form yang responsif
612 def display_form():
613     # Simulasi data
614     data = np.random.randn(10, 2)
615     df = pd.DataFrame(data, columns=["X", "Y"])
616
617     # Menampilkan form
618     st.form_bar_bar("Form Bar")
619
620 # Fungsi untuk menampilkan chart yang responsif
621 def display_chart():
622     # Simulasi data
623     data = np.random.randn(10, 2)
624     df = pd.DataFrame(data, columns=["X", "Y"])
625
626     # Menampilkan chart
627     st.chart_bar_bar("Chart Bar")
628
629 # Fungsi untuk menampilkan table yang responsif
630 def display_table():
631     # Simulasi data
632     data = np.random.randn(10, 2)
633     df = pd.DataFrame(data, columns=["X", "Y"])
634
635     # Menampilkan table
636     st.table_bar_bar("Table Bar")
637
638 # Fungsi untuk menampilkan progress bar yang responsif
639 def display_progress_bar():
640     # Simulasi data
641     data = np.random.randn(10, 2)
642     df = pd.DataFrame(data, columns=["X", "Y"])
643
644     # Menampilkan progress bar
645     st.progress_bar_bar_bar("Progress Bar")
646
647 # Fungsi untuk menampilkan status yang responsif
648 def display_status():
649     # Simulasi data
650     data = np.random.randn(10, 2)
651     df = pd.DataFrame(data, columns=["X", "Y"])
652
653     # Menampilkan status
654     st.status_bar_bar_bar("Status Bar")
655
656 # Fungsi untuk menampilkan alert yang responsif
657 def display_alert():
658     # Simulasi data
659     data = np.random.randn(10, 2)
660     df = pd.DataFrame(data, columns=["X", "Y"])
661
662     # Menampilkan alert
663     st.alert_bar_bar_bar("Alert Bar")
664
665 # Fungsi untuk menampilkan modal yang responsif
666 def display_modal():
667     # Simulasi data
668     data = np.random.randn(10, 2)
669     df = pd.DataFrame(data, columns=["X", "Y"])
670
671     # Menampilkan modal
672     st.modal_bar_bar_bar("Modal Bar")
673
674 # Fungsi untuk menampilkan sidebar yang responsif
675 def display_sidebar():
676     # Simulasi data
677     data = np.random.randn(10, 2)
678     df = pd.DataFrame(data, columns=["X", "Y"])
679
680     # Menampilkan sidebar
681     st.sidebar_bar_bar_bar("Sidebar Bar")
682
683 # Fungsi untuk menampilkan footer yang responsif
684 def display_footer():
685     # Simulasi data
686     data = np.random.randn(10, 2)
687     df = pd.DataFrame(data, columns=["X", "Y"])
688
689     # Menampilkan footer
690     st.footer_bar_bar_bar("Footer Bar")
691
692 # Fungsi untuk menampilkan header yang responsif
693 def display_header():
694     # Simulasi data
695     data = np.random.randn(10, 2)
696     df = pd.DataFrame(data, columns=["X", "Y"])
697
698     # Menampilkan header
699     st.header_bar_bar_bar("Header Bar")
700
701 # Fungsi untuk menampilkan subheader yang responsif
702 def display_subheader():
703     # Simulasi data
704     data = np.random.randn(10, 2)
705     df = pd.DataFrame(data, columns=["X", "Y"])
706
707     # Menampilkan subheader
708     st.subheader_bar_bar_bar("Subheader Bar")
709
710 # Fungsi untuk menampilkan text yang responsif
711 def display_text():
712     # Simulasi data
713     data = np.random.randn(10, 2)
714     df = pd.DataFrame(data, columns=["X", "Y"])
715
716     # Menampilkan text
717     st.text_bar_bar_bar("Text Bar")
718
719 # Fungsi untuk menampilkan markdown yang responsif
720 def display_markdown():
721     # Simulasi data
722     data = np.random.randn(10, 2)
723     df = pd.DataFrame(data, columns=["X", "Y"])
724
725     # Menampilkan markdown
726     st.markdown_bar_bar_bar("Markdown Bar")
727
728 # Fungsi untuk menampilkan code yang responsif
729 def display_code():
730     # Simulasi data
731     data = np.random.randn(10, 2)
732     df = pd.DataFrame(data, columns=["X", "Y"])
733
734     # Menampilkan code
735     st.code_bar_bar_bar("Code Bar")
736
737 # Fungsi untuk menampilkan image yang responsif
738 def display_image():
739     # Simulasi data
740     data = np.random.randn(10, 2)
741     df = pd.DataFrame(data, columns=["X", "Y"])
742
743     # Menampilkan image
744     st.image_bar_bar_bar("Image Bar")
745
746 # Fungsi untuk menampilkan video yang responsif
747 def display_video():
748     # Simulasi data
749     data = np.random.randn(10, 2)
750     df = pd.DataFrame(data, columns=["X", "Y"])
751
752     # Menampilkan video
753     st.video_bar_bar_bar("Video Bar")
754
755 # Fungsi untuk menampilkan audio yang responsif
756 def display_audio():
757     # Simulasi data
758     data = np.random.randn(10, 2)
759     df = pd.DataFrame(data, columns=["X", "Y"])
760
761     # Menampilkan audio
762     st.audio_bar_bar_bar("Audio Bar")
763
764 # Fungsi untuk menampilkan file yang responsif
765 def display_file():
766     # Simulasi data
767     data = np.random.randn(10, 2)
768     df = pd.DataFrame(data, columns=["X", "Y"])
769
770     # Menampilkan file
771     st.file_bar_bar_bar("File Bar")
772
773 # Fungsi untuk menampilkan folder yang responsif
774 def display_folder():
775     # Simulasi data
776     data = np.random.randn(10, 2)
777     df = pd.DataFrame(data, columns=["X", "Y"])
778
779     # Menampilkan folder
780     st.folder_bar_bar_bar("Folder Bar")
781
782 # Fungsi untuk menampilkan link yang responsif
783 def display_link():
784     # Simulasi data
785     data = np.random.randn(10, 2)
786     df = pd.DataFrame(data, columns=["X", "Y"])
787
788     # Menampilkan link
789     st.link_bar_bar_bar("Link Bar")
790
791 # Fungsi untuk menampilkan button yang responsif
792 def display_button():
793     # Simulasi data
794     data = np.random.randn(10, 2)
795     df = pd.DataFrame(data, columns=["X", "Y"])
796
797     # Menampilkan button
798     st.button_bar_bar_bar("Button Bar")
799
800 # Fungsi untuk menampilkan form yang responsif
801 def display_form():
802     # Simulasi data
803     data = np.random.randn(10, 2)
804     df = pd.DataFrame(data, columns=["X", "Y"])
805
806     # Menampilkan form
807     st.form_bar_bar_bar("Form Bar")
808
809 # Fungsi untuk menampilkan chart yang responsif
810 def display_chart():
811     # Simulasi data
812     data = np.random.randn(10, 2)
813     df = pd.DataFrame(data, columns=["X", "Y"])
814
815     # Menampilkan chart
816     st.chart_bar_bar_bar("Chart Bar")
817
818 # Fungsi untuk menampilkan table yang responsif
819 def display_table():
820     # Simulasi data
821     data = np.random.randn(10, 2)
822     df = pd.DataFrame(data, columns=["X", "Y"])
823
824     # Menampilkan table
825     st.table_bar_bar_bar("Table Bar")
826
827 # Fungsi untuk menampilkan progress bar yang responsif
828 def display_progress_bar():
829     # Simulasi data
830     data = np.random.randn(10, 2)
831     df = pd.DataFrame(data, columns=["X", "Y"])
832
833     # Menampilkan progress bar
834     st.progress_bar_bar_bar_bar("Progress Bar")
835
836 # Fungsi untuk menampilkan status yang responsif
837 def display_status():
838     # Simulasi data
839     data = np.random.randn(10, 2)
840     df = pd.DataFrame(data, columns=["X", "Y"])
841
842     # Menampilkan status
843     st.status_bar_bar_bar_bar("Status Bar")
844
845 # Fungsi untuk menampilkan alert yang responsif
846 def display_alert():
847     # Simulasi data
848     data = np.random.randn(10, 2)
849     df = pd.DataFrame(data, columns=["X", "Y"])
850
851     # Menampilkan alert
852     st.alert_bar_bar_bar_bar("Alert Bar")
853
854 # Fungsi untuk menampilkan modal yang responsif
855 def display_modal():
856     # Simulasi data
857     data = np.random.randn(10, 2)
858     df = pd.DataFrame(data, columns=["X", "Y"])
859
860     # Menampilkan modal
861     st.modal_bar_bar_bar_bar("Modal Bar")
862
863 # Fungsi untuk menampilkan sidebar yang responsif
864 def display_sidebar():
865     # Simulasi data
866     data = np.random.randn(10, 2)
867     df = pd.DataFrame(data, columns=["X", "Y"])
868
869     # Menampilkan sidebar
870     st.sidebar_bar_bar_bar_bar("Sidebar Bar")
871
872 # Fungsi untuk menampilkan footer yang responsif
873 def display_footer():
874     # Simulasi data
875     data = np.random.randn(10, 2)
876     df = pd.DataFrame(data, columns=["X", "Y"])
877
878     # Menampilkan footer
879     st.footer_bar_bar_bar_bar("Footer Bar")
880
881 # Fungsi untuk menampilkan header yang responsif
882 def display_header():
883     # Simulasi data
884     data = np.random.randn(10, 2)
885     df = pd.DataFrame(data, columns=["X", "Y"])
886
887     # Menampilkan header
888     st.header_bar_bar_bar_bar("Header Bar")
889
890 # Fungsi untuk menampilkan subheader yang responsif
891 def display_subheader():
892     # Simulasi data
893     data = np.random.randn(10, 2)
894     df = pd.DataFrame(data, columns=["X", "Y"])
895
896     # Menampilkan subheader
897     st.subheader_bar_bar_bar_bar("Subheader Bar")
898
899 # Fungsi untuk menampilkan text yang responsif
900 def display_text():
901     # Simulasi data
902     data = np.random.randn(10, 2)
903     df = pd.DataFrame(data, columns=["X", "Y"])
904
905     # Menampilkan text
906     st.text_bar_bar_bar_bar("Text Bar")
907
908 # Fungsi untuk menampilkan markdown yang responsif
909 def display_markdown():
910     # Simulasi data
911     data = np.random.randn(10, 2)
912     df = pd.DataFrame(data, columns=["X", "Y"])
913
914     # Menampilkan markdown
915     st.markdown_bar_bar_bar_bar("Markdown Bar")
916
917 # Fungsi untuk menampilkan code yang responsif
918 def display_code():
919     # Simulasi data
920     data = np.random.randn(10, 2)
921     df = pd.DataFrame(data, columns=["X", "Y"])
922
923     # Menampilkan code
924     st.code_bar_bar_bar_bar("Code Bar")
925
926 # Fungsi untuk menampilkan image yang responsif
927 def display_image():
928     # Simulasi data
929     data = np.random.randn(10, 2)
930     df = pd.DataFrame(data, columns=["X", "Y"])
931
932     # Menampilkan image
933     st.image_bar_bar_bar_bar("Image Bar")
934
935 # Fungsi untuk menampilkan video yang responsif
936 def display_video():
937     # Simulasi data
938     data = np.random.randn(10, 2)
939     df = pd.DataFrame(data, columns=["X", "Y"])
940
941     # Menampilkan video
942     st.video_bar_bar_bar_bar("Video Bar")
943
944 # Fungsi untuk menampilkan audio yang responsif
945 def display_audio():
946     # Simulasi data
947     data = np.random.randn(10, 2)
948     df = pd.DataFrame(data, columns=["X", "Y"])
949
950     # Menampilkan audio
951     st.audio_bar_bar_bar_bar("Audio Bar")
952
953 # Fungsi untuk menampilkan file yang responsif
954 def display_file():
955     # Simulasi data
956     data = np.random.randn(10, 2)
957     df = pd.DataFrame(data, columns=["X", "Y"])
958
959     # Menampilkan file
960     st.file_bar_bar_bar_bar("File Bar")
961
962 # Fungsi untuk menampilkan folder yang responsif
963 def display_folder():
964     # Simulasi data
965     data = np.random.randn(10, 2)
966     df = pd.DataFrame(data, columns=["X", "Y"])
967
968     # Menampilkan folder
969     st.folder_bar_bar_bar_bar("Folder Bar")
970
971 # Fungsi untuk menampilkan link yang responsif
972 def display_link():
973     # Simulasi data
974     data = np.random.randn(10, 2)
975     df = pd.DataFrame(data, columns=["X", "Y"])
976
977     # Menampilkan link
978     st.link_bar_bar_bar_bar("Link Bar")
979
980 # Fungsi untuk menampilkan button yang responsif
981 def display_button():
982     # Simulasi data
983     data = np.random.randn(10, 2)
984     df = pd.DataFrame(data, columns=["X", "Y"])
985
986     # Menampilkan button
987     st.button_bar_bar_bar_bar("Button Bar")
988
989 # Fungsi untuk menampilkan form yang responsif
990 def display_form():
991     # Simulasi data
992     data = np.random.randn(10, 2)
993     df = pd.DataFrame(data, columns=["X", "Y"])
994
995     # Menampilkan form
996     st.form_bar_bar_bar_bar("Form Bar")
997
998 # Fungsi untuk menampilkan chart yang responsif
999 def display_chart():
1000     # Simulasi data
1001     data = np.random.randn(10, 2)
1002     df = pd.DataFrame(data, columns=["X", "Y"])
1003
1004     # Menampilkan chart
1005     st.chart_bar_bar_bar_bar("Chart Bar")
1006
1007 # Fungsi untuk menampilkan table yang responsif
1008 def display_table():
1009     # Simulasi data
1010     data = np.random.randn(10, 2)
1011     df = pd.DataFrame(data, columns=["X", "Y"])
1012
1013     # Menampilkan table
1014     st.table_bar_bar_bar_bar("Table Bar")
1015
1016 # Fungsi untuk menampilkan progress bar yang responsif
1017 def display_progress_bar():
1018     # Simulasi data
1019     data = np.random.randn(10, 2)
1020     df = pd.DataFrame(data, columns=["X", "Y"])
1021
1022     # Menampilkan progress bar
1023     st.progress_bar_bar_bar_bar_bar("Progress Bar")
1024
1025 # Fungsi untuk menampilkan status yang responsif
1026 def display_status():
1027     # Simulasi data
1028     data = np.random.randn(10, 2)
1029     df = pd.DataFrame(data, columns=["X", "Y"])
1030
1031     # Menampilkan status
1032     st.status_bar_bar_bar_bar_bar("Status Bar")
1033
1034 # Fungsi untuk menampilkan alert yang responsif
1035 def display_alert():
1036     # Simulasi data
1037     data = np.random.randn(10, 2)
1038     df = pd.DataFrame(data, columns=["X", "Y"])
1039
1040     # Menampilkan alert
1041     st.alert_bar_bar_bar_bar_bar("Alert Bar")
1042
1043 # Fungsi untuk menampilkan modal yang responsif
1044 def display_modal():
1045     # Simulasi data
1046     data = np.random.randn(10, 2)
1047     df = pd.DataFrame(data, columns=["X", "Y"])
1048
1049     # Menampilkan modal
1050     st.modal_bar_bar_bar_bar_bar("Modal Bar")
1051
1052 # Fungsi untuk menampilkan sidebar yang responsif
1053 def display_sidebar():
1054     # Simulasi data
1055     data = np.random.randn(10, 2)
1056     df = pd.DataFrame(data, columns=["X", "Y"])
1057
1058     # Menampilkan sidebar
1059     st.sidebar_bar_bar_bar_bar_bar("Sidebar Bar")
1060
1061 # Fungsi untuk menampilkan footer yang responsif
1062 def display_footer():
1063     # Simulasi data
1064     data = np.random.randn(10, 2)
1065     df = pd.DataFrame(data, columns=["X", "Y"])
1066
1067     # Menampilkan footer
1068     st.footer_bar_bar_bar_bar_bar("Footer Bar")
1069
1070 # Fungsi untuk menampilkan header yang responsif
1071 def display_header():
1072     # Simulasi data
1073     data = np.random.randn(10, 2)
1074     df = pd.DataFrame(data, columns=["X", "Y"])
1075
1076     # Menampilkan header
1077     st.header_bar_bar_bar_bar_bar("Header Bar")
1078
1079 # Fungsi untuk menampilkan subheader yang responsif
1080 def display_subheader():
1081     # Simulasi data
1082     data = np.random.randn(10, 2)
1083     df = pd.DataFrame(data, columns=["X", "Y"])
1084
1085     # Menampilkan subheader
1086     st.subheader_bar_bar_bar_bar_bar("Subheader Bar")
1087
1088 # Fungsi untuk menampilkan text yang responsif
1089 def display_text():
1090     # Simulasi data
1091     data = np.random.randn(10, 2)
1092     df = pd.DataFrame(data, columns=["X", "Y"])
1093
1094     # Menampilkan text
1095     st.text_bar_bar_bar_bar_bar("Text Bar")
1096
1097 # Fungsi untuk menampilkan markdown yang responsif
1098 def display_markdown():
1099     # Simulasi data
1100     data = np.random.randn(10, 2)
1101     df = pd.DataFrame(data, columns=["X", "Y"])
1102
1103     # Menampilkan markdown
1104     st.markdown_bar_bar_bar_bar_bar("Markdown Bar")
1105
1106 # Fungsi untuk menampilkan code yang responsif
1107 def display_code():
1108     # Simulasi data
1109     data = np.random.randn(10, 2)
1110     df = pd.DataFrame(data, columns=["X", "Y"])
1111
1112     # Menampilkan code
1113     st.code_bar_bar_bar_bar_bar("Code Bar")
1114
1115 # Fungsi untuk menampilkan image yang responsif
1116 def display_image():
1117     # Simulasi data
1118     data = np.random.randn(10, 2)
1119     df = pd.DataFrame(data, columns=["X", "Y"])
1120
1121     # Menampilkan image
1122     st.image_bar_bar_bar_bar_bar("Image Bar")
1123
1124 # Fungsi untuk menampilkan video yang responsif
1125 def display_video():
1126     # Simulasi data
1127     data = np.random.randn(10, 2)
1128     df = pd.DataFrame(data, columns=["X", "Y"])
1129
1130     # Menampilkan video
1131     st.video_bar_bar_bar_bar_bar("Video Bar")
1132
1133 # Fungsi untuk menampilkan audio yang responsif
1134 def display_audio():
1135     # Simulasi data
1136     data = np.random.randn(10, 2)
1137     df = pd.DataFrame(data, columns=["X", "Y"])
1138
1139     # Menampilkan audio
1140     st.audio_bar_bar_bar_bar_bar("Audio Bar")
1141
1142 # Fungsi untuk menampilkan file yang responsif
1143 def display_file():
1144     # Simulasi data
1145     data = np.random.randn(10, 2)
1146     df = pd.DataFrame(data, columns=["X", "Y"])
1147
1148     # Menampilkan
```

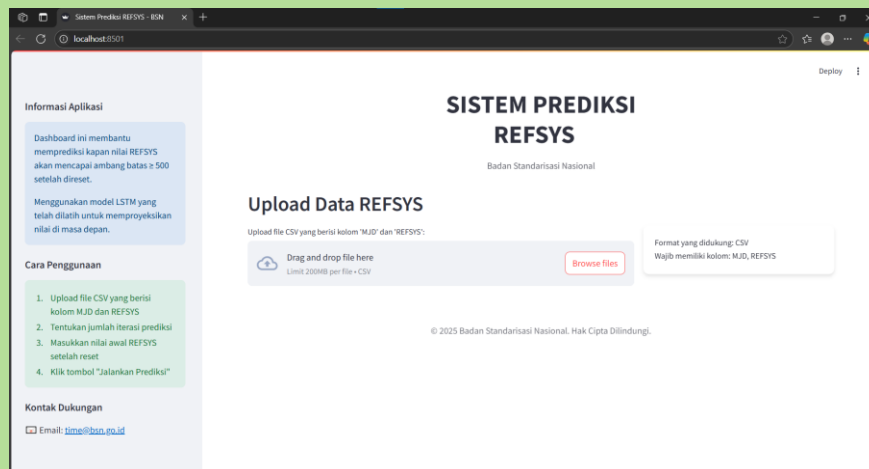
PETUNJUK PENGGUNAAN

2. Jalankan Perintah : Streamlit run app.py



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\raihan\SKRIPSI\refsys> Streamlit run app.py
```

3. Mengupload csv merged_output



PETUNJUK PENGGUNAAN

Informasi Aplikasi

Dashboard ini membantu memprediksi kapan nilai REFSYS akan mencapai ambang batas ≥ 500 setelah direset.

Menggunakan model LSTM yang telah dilatih untuk memproyeksikan nilai di masa depan.

Cara Penggunaan

1. Upload file CSV yang berisi kolom MJD
2. Tentukan jumlah hari
3. Masukkan nilai awal setelah reset
4. Klik tombol "Jalankan Prediksi"

Kontak Dukungan

Email: time@bsn.go.id

PREDIKSI REFSYS

Standarisasi Nasional

Browse files

asi Nasional. Hak Cipta Dilindungi.

Open

This PC > Downloads

Search Downloads

Organize New folder

OneDrive - Person

This PC

3D Objects

Desktop

Documents

Downloads

Music

Pictures

Videos

System (C:)

Data (D:)

User Interface Testing (Responses) - Form Responses 1 Microsoft Excel Comma Separate... 130 KB

PCOS_dataset Microsoft Excel Comma Separate... 130 KB

nutrition Microsoft Excel Comma Separate... 191 KB

Data Anak - Data Anak (2) Microsoft Excel Comma Separate... 162 bytes

template_jadwal_pijat_anak Microsoft Excel Comma Separate... 613 bytes

exported Microsoft Excel Comma Separate... 83 bytes

merged_output (1) Microsoft Excel Comma Separate... 8.45 KB

Data Anak - Data Anak (1) Microsoft Excel Comma Separate... 47 bytes

Data Anak - Data Anak Microsoft Excel Comma Separate... 162 bytes

merged_output Microsoft Excel Comma Separate... 7.81 KB

File name: Microsoft Excel Comma Separa

Upload from mobile

Open

Cancel

4. Pindah ke Tab Prediksi

Informasi Aplikasi

Dashboard ini membantu memprediksi kapan nilai REFSYS akan mencapai ambang batas ≥ 500 setelah direset.

Menggunakan model LSTM yang telah dilatih untuk memproyeksikan nilai di masa depan.

Cara Penggunaan

1. Upload file CSV yang berisi

merged_output.csv 7.8KB

File berhasil diunggah dan dibaca!

Data Prediksi

Pratinjau Data

	MJD	Tanggal	REFSYS
0	60398	3/29/2024	-486
1	60399	3/30/2024	-445
2	60400	3/31/2024	-442

PETUNJUK PENGGUNAAN

5. Sesuaikan Parameter Prediksi dan Nilai awal reset refsysnya serta tetap gunakan model default

The screenshot displays the 'Sistem Prediksi REFSYS - BSN' web application running on localhost:8501. The interface is divided into a left sidebar and a main content area.

Informasi Aplikasi

Dashboard ini membantu memprediksi kapan nilai REFSYS akan mencapai ambang batas ≥ 500 setelah direset.

Menggunakan model LSTM yang telah dilatih untuk memproyeksikan nilai di masa depan.

Cara Penggunaan

1. Upload file CSV yang berisi kolom MJD dan REFSYS
2. Tentukan jumlah iterasi prediksi
3. Masukkan nilai awal REFSYS setelah reset
4. Klik tombol "Jalankan Prediksi"

Kontak Dukungan

Email: tjme@bsn.go.id

Main Content Area:

- A green success message: "File berhasil diunggah dan dibaca!"
- Tabs: "Data" and "Prediksi" (selected).
- Parameter Prediksi**
 - Jumlah iterasi prediksi: Input field with value "1" and a range from 1 to 500.
 - Nilai ambang batas: Input field with value "500".
- Nilai Awal REFSYS Setelah Reset**
 - Reset #1: Input field with value "-400.00".
 - Pilih model: Radio buttons for "Model Default" (selected) and "Upload Model Kustom".
 - A blue button labeled "Menggunakan model default".
- A "Jalankan Prediksi" button at the bottom right.

PETUNJUK PENGGUNAAN

6. Jalankan Prediksi dan Tunggu Hasilnya Keluar

