

Logbook – Secure Communication and Protocol Enforcement of R25 -039



Project ID: R25 - 039

Project Title: Data-Privacy Focused Federated Learning Framework for Industrial IoT

Student Details:

Names:

Dissanayaka K.D.A.R. A

Student IDs:

IT21828348

Supervisor: Mr. Amila Seneratha

Co-Supervisor: Mr. Tharaniyawarma Kumaralingam

Date of Submission: 2025

Contents

1.	Group Details.....	3
2.	Project Details.....	3
3.	Communication Methods	4
4.	Meetings With Supervisors	6
5.	Task Details	7
5.1	Personal task Assigning and Completion	7
6.	System Details	8
6.1	System completion status.....	8
6.2	System Design	10
I.	System Architecture.....	10
II.	Module Architecture	10
6.3	System Testing.....	11
6.4	System Codes	12
7.	GitHub Upload	17
8.	Documentation.....	17
8.1	Proposal.....	17
8.2	Presentation 1	18
8.3	Presentation 2.....	18
8.4	Final Presentation.....	19
8.5	Final Product.....	19
8.6	Research Paper.....	20
III.	Conference Appetence	20
9.	CDAP upload.....	21
10.	Website	22
10.1	Development	22
10.2	Finalize.....	25

1. Group Details

Student Details:		
Names:	Student IDs:	Research Component
Nanayakkara Y.D.T. D	IT21826368	Attack Defense and Resilience
Mendis H.R.M	IT21822612	Privacy Preservation
Weerasinghe K.M	IT21831904	Secure Aggregation
Dissanayaka K.D.A.R. A	IT21828348	Secure Communication and Protocol Enforcement

2. Project Details

Topic - Data-Privacy Focused Federated Learning Framework for Industrial IoT

Aim – To develop a product that going to full fill the research

Deliverables – Federated Learning Framework designed for industrial internet of things

This project was initiated to develop a secure and private **Federated Learning (FL) framework** specifically for **Industrial IoT (IIoT)** environments.

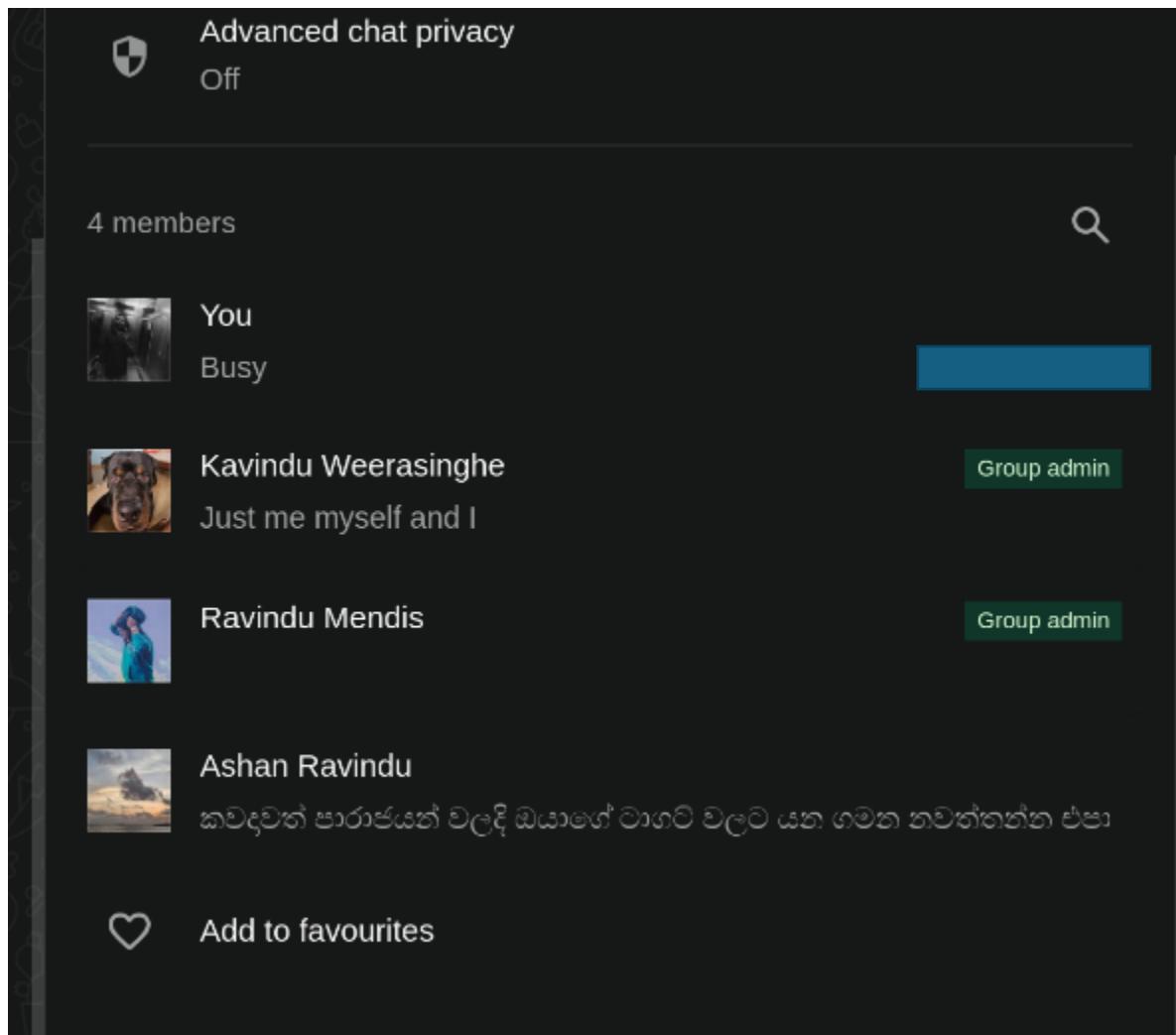
The Challenge: Traditional AI methods require centralizing sensitive factory data, which poses major **privacy risks** and clashes with the distributed nature of industrial operations. Existing FL solutions are insufficient because they fail to simultaneously provide robust security, data privacy, and efficient operation on **resource-limited IIoT devices**.

The Solution: The developed framework is a multi-layered system that provides **end-to-end protection**.

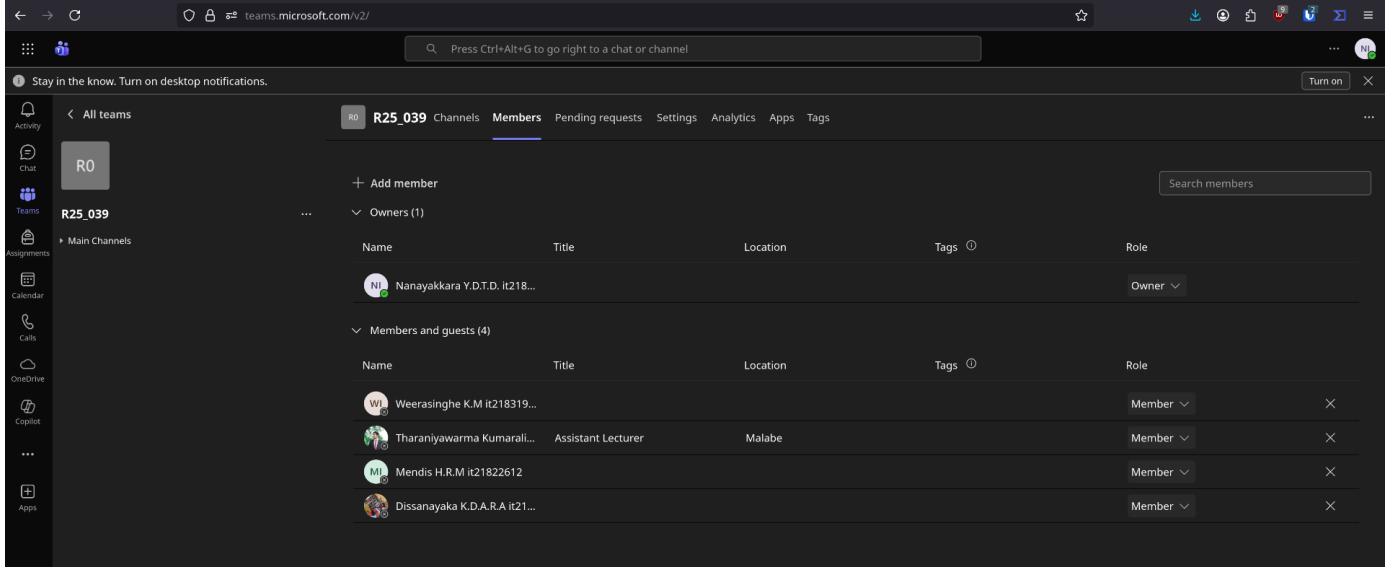
- It uses techniques like **Differential Privacy (DP)** and **Homomorphic Encryption (HE)** to guarantee data confidentiality.
- It implements a robust protocol that uses **client/server validation** to actively block cyber threats such as **Model Poisoning and Byzantine Attacks**.
- The system is optimized for **efficiency** to reduce overhead on IIoT devices.

3. Communication Methods

WhatsApp Group – Team

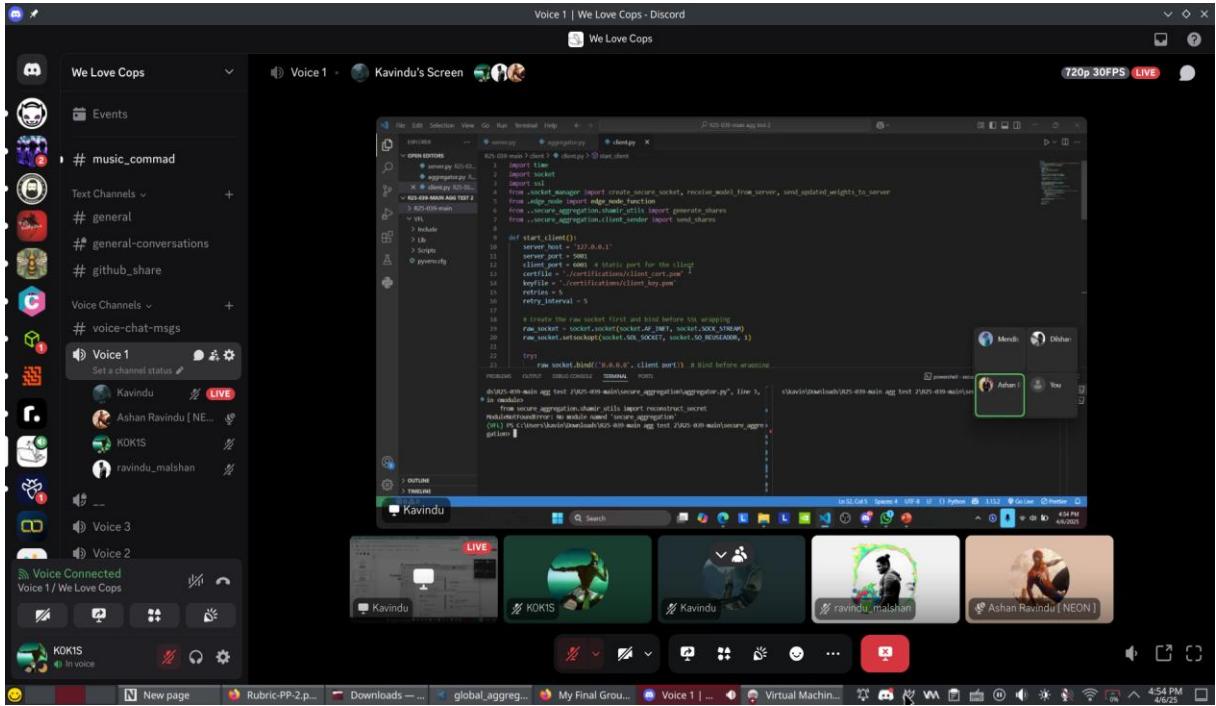


Microsoft Teams - All



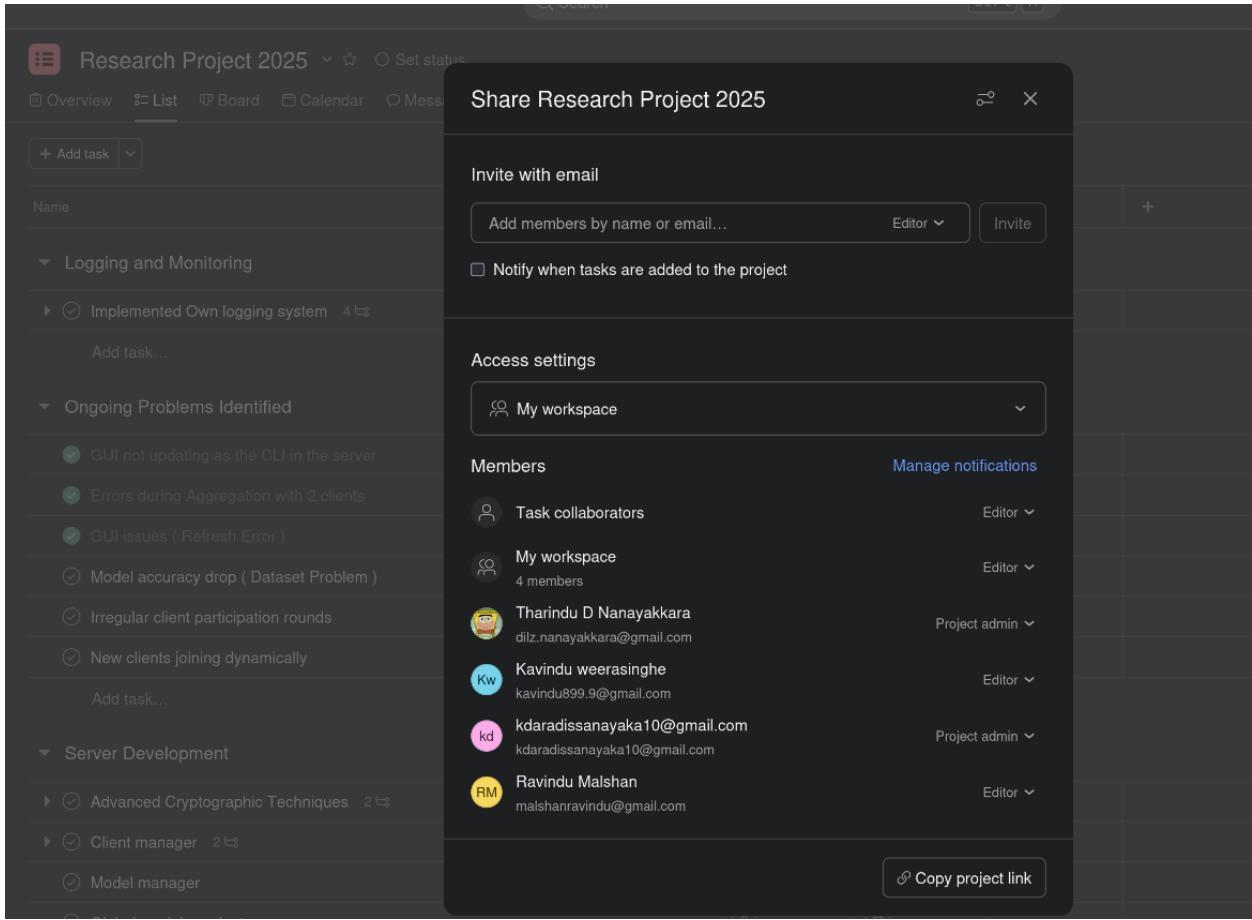
The screenshot shows the Microsoft Teams interface for the 'R25_039' channel. On the left, there's a sidebar with various team-related links like Activity, Chat, Teams, Assignments, Calendar, Calls, OneDrive, Copilot, and Apps. The main area displays the channel members. At the top, there are tabs for Channels, Members (which is selected), Pending requests, Settings, Analytics, Apps, and Tags. Below the tabs, there's a search bar labeled 'Search members'. Under the 'Owners (1)' section, there's one owner listed: Nanayakkara Y.D.T.D. it218... with the role 'Owner'. Under the 'Members and guests (4)' section, there are four members listed: Weerasinghe K.M it218319..., Tharaniyawarma Kumarali... Assistant Lecturer from Malabe, Mendis H.R.M it21822612, and Dissanayaka K.D.A.R.A it21... All four members have the role 'Member'.

Group Meetings – Discord – Team



The screenshot shows a Discord server interface for 'We Love Cops'. On the left, there's a sidebar with channels like '# music_command', '# general', '# general-conversations', '# github_share', '# voice-chat-msgs', 'Voice 1' (which is currently active and has a 'LIVE' indicator), 'Voice 3', and 'Voice 2'. The 'Voice 1' channel has several users in it, including Kavindu, Ashan Ravindu [NEON], KOKIS, and ravindu_malshan. In the center, there's a code editor window titled 'R25-039 MAIN AGG TEST 2' showing Python code related to socket aggregation. On the right, there's a video feed of Kavindu. The bottom of the screen shows the Windows taskbar with various open applications like File Explorer, Python, and a browser.

Asana – Task Assigning – Team

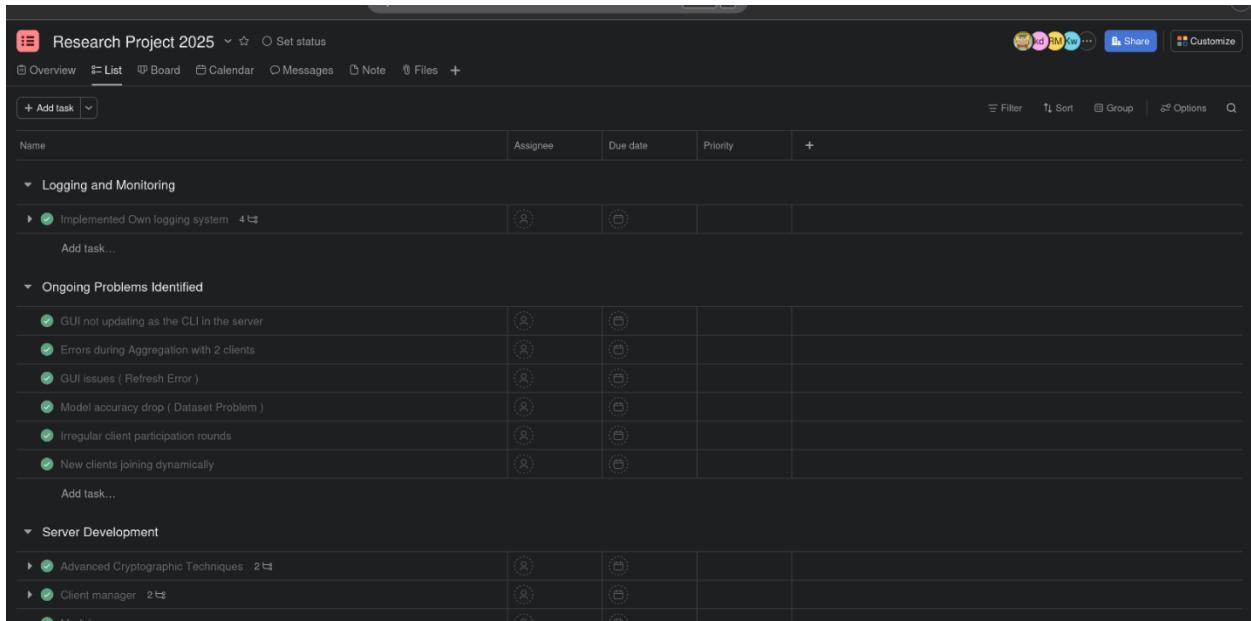


The screenshot shows a project management interface for 'Research Project 2025'. On the left, there's a sidebar with sections like 'Logging and Monitoring', 'Ongoing Problems Identified', 'Server Development', and 'Add task...'. The main area displays a list of tasks under these sections. A modal window titled 'Share Research Project 2025' is open in the center. It has sections for 'Invite with email' (with a field to 'Add members by name or email...' and a 'Notify when tasks are added to the project' checkbox), 'Access settings' (set to 'My workspace'), 'Members' (listing four users with their roles: Tharindu D Nanayakkara (Project admin), Kavindu weerasinghe (Editor), kdaradissanayaka10@gmail.com (Project admin), and Ravindu Malshan (Editor)), and a 'Manage notifications' section. At the bottom of the modal is a 'Copy project link' button.

4. Meetings With Supervisors

All the meetings were conducted in person and only WhatsApp calls were taken to organize the meeting

5. Task Details



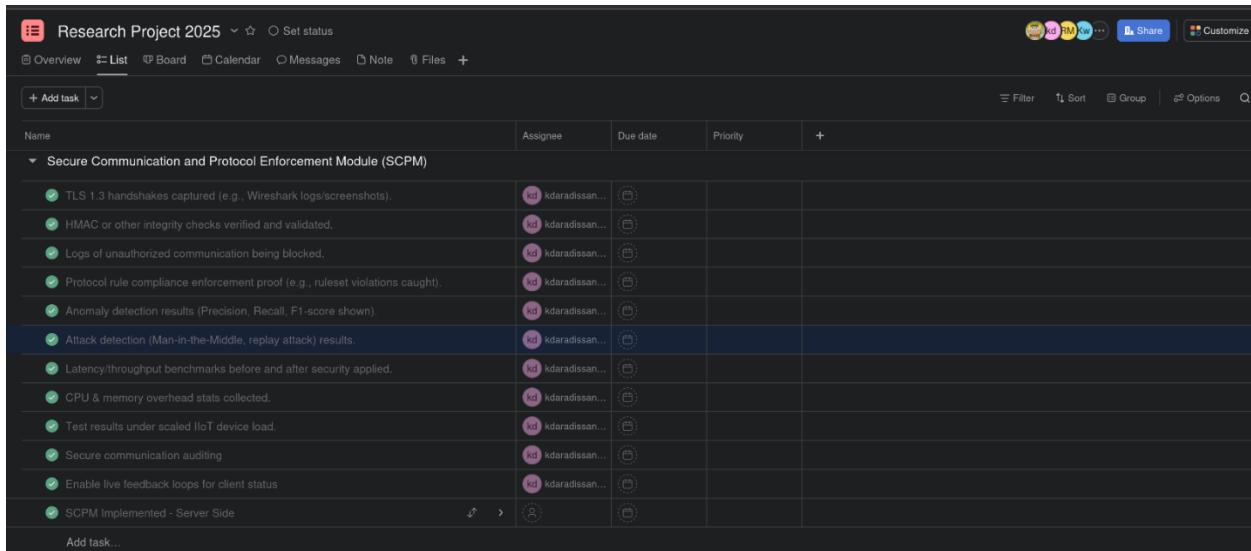
The screenshot shows a task management interface for a project titled "Research Project 2025". The interface includes a header with navigation links like Overview, List, Board, Calendar, Messages, Note, Files, and a search bar. Below the header is a toolbar with icons for filter, sort, group, options, and a search field.

The main area displays a table of tasks:

Name	Assignee	Due date	Priority	Actions
Logging and Monitoring				
Implemented Own logging system	[User Icon]	[Icon]		
Add task...				
Ongoing Problems Identified				
GUI not updating as the CLI in the server	[User Icon]	[Icon]		
Errors during Aggregation with 2 clients	[User Icon]	[Icon]		
GUI issues (Refresh Error)	[User Icon]	[Icon]		
Model accuracy drop (Dataset Problem)	[User Icon]	[Icon]		
Irregular client participation rounds	[User Icon]	[Icon]		
New clients joining dynamically	[User Icon]	[Icon]		
Add task...				
Server Development				
Advanced Cryptographic Techniques	[User Icon]	[Icon]		
Client manager	[User Icon]	[Icon]		
Add task...				

Secure communication and protocol enforcement module were assigned to me.

5.1 Personal task Assigning and Completion



The screenshot shows a task management interface for a project titled "Research Project 2025". The interface includes a header with navigation links like Overview, List, Board, Calendar, Messages, Note, Files, and a search bar. Below the header is a toolbar with icons for filter, sort, group, options, and a search field.

The main area displays a table of tasks:

Name	Assignee	Due date	Priority	Actions
Secure Communication and Protocol Enforcement Module (SCPM)				
TLS 1.3 handshakes captured (e.g., Wireshark logs/screenshots).	[User Icon]	[Icon]		
HMAC or other integrity checks verified and validated.	[User Icon]	[Icon]		
Logs of unauthorized communication being blocked.	[User Icon]	[Icon]		
Protocol rule compliance enforcement proof (e.g., ruleset violations caught).	[User Icon]	[Icon]		
Anomaly detection results (Precision, Recall, F1-score shown).	[User Icon]	[Icon]		
Attack detection (Man-in-the-Middle, replay attack) results.	[User Icon]	[Icon]		
Latency/throughput benchmarks before and after security applied.	[User Icon]	[Icon]		
CPU & memory overhead stats collected.	[User Icon]	[Icon]		
Test results under scaled IIoT device load.	[User Icon]	[Icon]		
Secure communication auditing	[User Icon]	[Icon]		
Enable live feedback loops for client status	[User Icon]	[Icon]		
SCPM Implemented - Server Side	[User Icon]	[Icon]		
Add task...				

6. System Details

6.1 System completion status

Finished

SCPM TUI

Whole system is related

Proposed System – Frontend Terminal User Interface(TUI)

Federated Learning Framework v2.0 2025-09-15 12:15:28

(1) Overview | (2) Model Manager | (3) Client Health | (4) Modules | (5) Logs | (6) TUI Details

Overview

Server Overview	Orchestrator	Secure Aggregation Module	Attack Defense And Resilience Module
Server State: Online Progress: Round 0 of 1 Round Duration: N/A Last Aggregation: No aggregation yet	Is Training In Progress: True Current State: WAITING_FOR_UPDATES Current Round: 1 Number: Total Rounds: 100 Clients Per Round: 2 Updates Received: 2 Selected Clients: 3 Count: Round Start TIME: 1737918572.9624783 Round Threashold: 1000	Aggregation Protocol: SGAGG2 Security Level: High Updates In Queue: 2 Failed Sessions: 0 Last Aggregation: N/A Time: Status: active	Status: running_ml_node Blocked Clients: 0 Count: Champion Is: False Tainted: 0 Challenger Is: True Tainted: 2 Training Buffer Size: Performance: Champion: 0.0 Challenger: 0.0 History: []
Client Managed	Model Manager	Privacy Preserving Module	Server Communication And Protocol Enforcement Module
Total Clients: 3 Connected Clients: 3 Status Checker: True Running:	Version: 0 Status: N/A	Module Name: PPM Status: active Epsilon: 1.0 Delta: 1e-05 Is Active: True Description: Privacy-Preserving Mechanism for policy auditing	Module Name: SCPM Status: active TLS Enabled: False Client Authentication: False Enabled: True Last Security Event: System initialized. Active Protocols: gRPC, REST API Description: Manages communication protocols and server state.

Client Health Status (Total: 3, Active: 3, Blocked: 0)

Client ID	Status	Reputati...	Uptime	Client Type	Details
client_2	Connected	100	0:06:08	gRPC	unknown
client_1	Connected	100	0:06:09	gRPC	unknown
client_0	Connected	100	0:02:31	gRPC	unknown

Recent Logs

```

2025-09-15 12:15:21 INFO  [Background status update] Connected clients count is 3
2025-09-15 12:15:21 INFO  [Processing global model] for client client_1 for round 1.
127.0.0.1 [15/Sep/2025:12:15:21 +0530] "GET /api/overview HTTP/1.1" 200 655 "-"
"Python/3.10 aiohttp/3.22.1"
127.0.0.1 [15/Sep/2025:12:15:21 +0530] "GET /api/model/metrics_details HTTP/1.1" 200 388 -
"Python/3.10 aiohttp/3.22.1"
127.0.0.1 [15/Sep/2025:12:15:21 +0530] "GET /api/client_health HTTP/1.1" 200 1383 "-"
"Python/3.10 aiohttp/3.22.1"

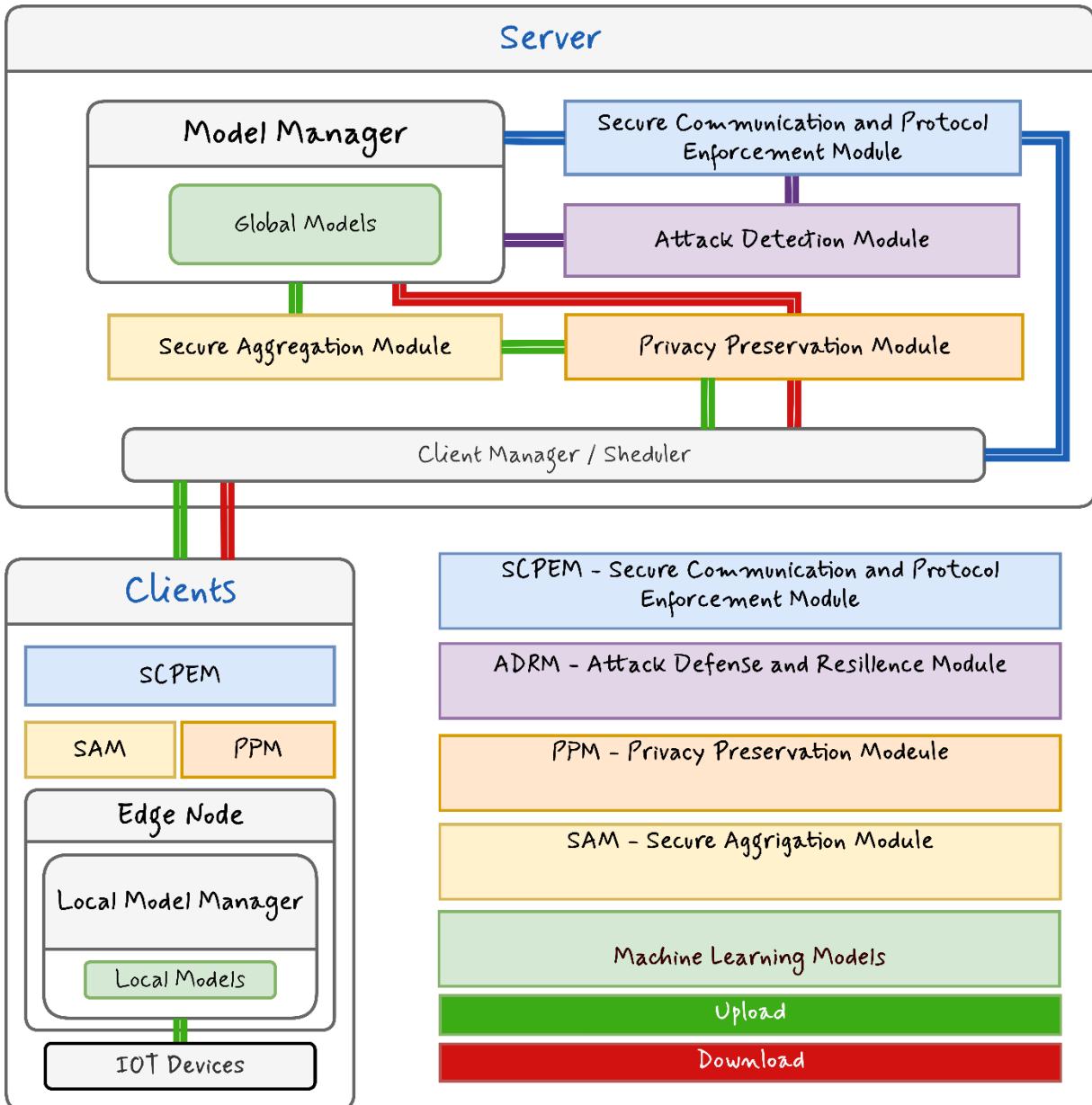
```

Log - Manager

Federated Learning Framework v2.0				2025-09-15 02:39:31
				(1) Overview (2) Model Manager (3) Client Health (4) Modules (5) Logs (6) TUI Details
Logs				
Level: ALL Search: None				f: Filter Level s: Search x: Reset
Timestamp	Level	Logger	Message	Recent Logs
2025-09-15 02:39:13,177	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/logs HTTP/1.1" 200 54505 "-" "Python/3.10 aiohttp/3.12.15"	
2025-09-15 02:39:13,178	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/status HTTP/1.1" 200 1988 "-" "Python/3.10 aiohttp/3.12.15"	
2025-09-15 02:39:13,179	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/module_status/mm HTTP/1.1" 200 485 "-" "Python/3.10 aiohttp/3.12.15"	
2025-09-15 02:39:13,180	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/module_status/sam HTTP/1.1" 200 365 "-" "Python/3.10 aiohttp/3.12.15"	
2025-09-15 02:39:13,181	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/module_status/adm HTTP/1.1" 200 417 "-" "Python/3.10 aiohttp/3.12.15"	
2025-09-15 02:39:13,181	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/module_status/ppm HTTP/1.1" 200 353 "-" "Python/3.10 aiohttp/3.12.15"	
2025-09-15 02:39:13,182	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/module_status/scpm HTTP/1.1" 200 453 "-" "Python/3.10 aiohttp/3.12.15"	
2025-09-15 02:39:13,183	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:13 +0530] "GET /api/module_status/orchestrator HTTP/1.1" 200 485 "-" "Python/3.10 aiohttp/3.12.15"	
2025-09-15 02:39:13,264	INFO	ServerControlPlaneManager	Dashboard status updated: Connected clients count is 4.	
2025-09-15 02:39:13,266	WARNING	Orchestrator	Denied model request from client_2. Orchestrator state is PAUSED_INSUFFICIENT_CLIENTS.	
2025-09-15 02:39:15,257	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:15 +0530] "GET /api/overview HTTP/1.1" 200 680 "-" "Python/3.10 aiohttp/3.12.15"	
2025-09-15 02:39:15,258	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:15 +0530] "GET /api/model/metrics_details HTTP/1.1" 200 6890 "-" "Python/3.10 aiohttp/3.12.15"	
2025-09-15 02:39:15,259	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:15 +0530] "GET /api/client_health HTTP/1.1" 200 1769 "-" "Python/3.10 aiohttp/3.12.15"	
2025-09-15 02:39:15,267	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:15 +0530] "GET /api/logs HTTP/1.1" 200 54016 "-" "Python/3.10 aiohttp/3.12.15"	
2025-09-15 02:39:15,268	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:15 +0530] "GET /api/status HTTP/1.1" 200 1988 "-" "Python/3.10 aiohttp/3.12.15"	
2025-09-15 02:39:15,268	INFO	aiohttp.access	127.0.0.1 [15/Sep/2025:02:39:15 +0530] "GET /api/module_status/mm HTTP/1.1" 200 485 "-" "Python/3.10 aiohttp/3.12.15"	

6.2 System Design

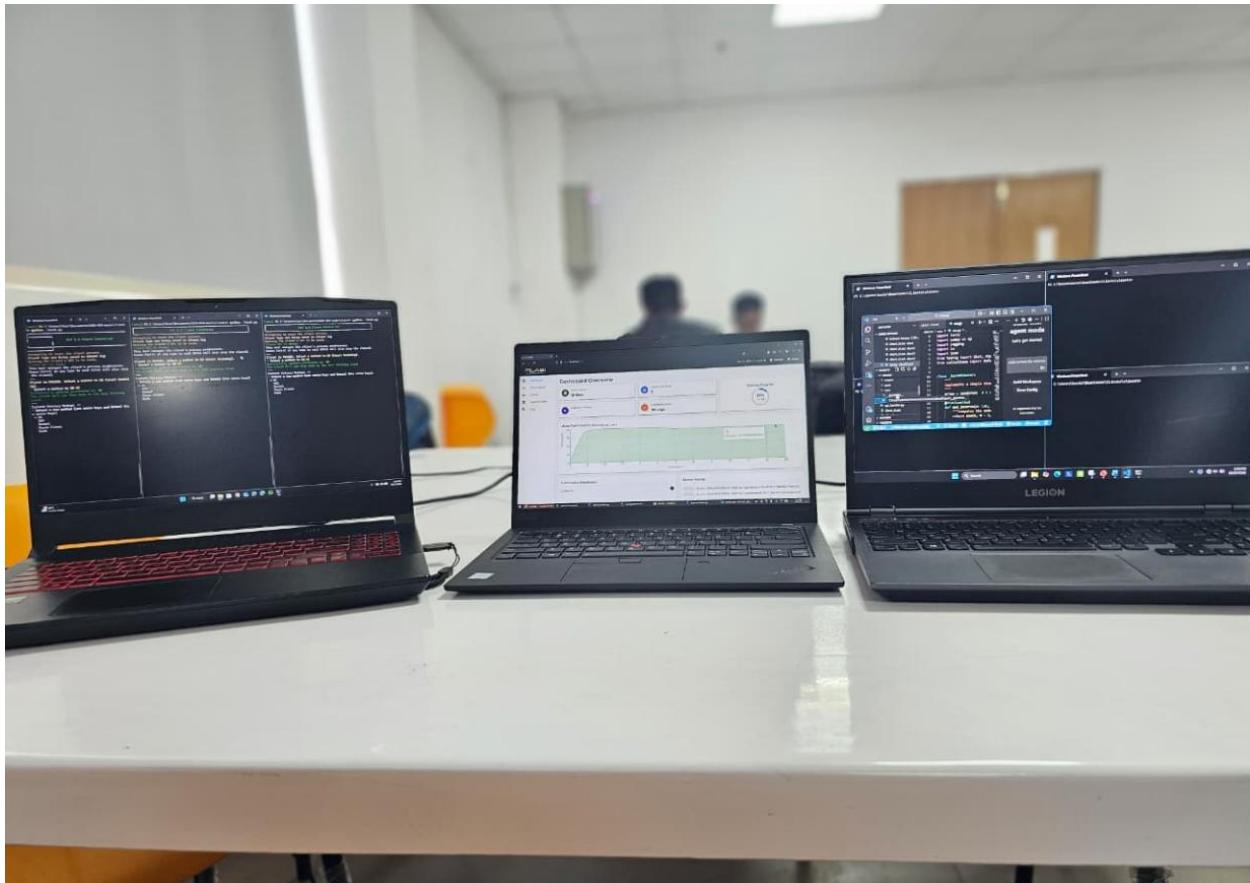
I. System Architecture



II. Module Architecture

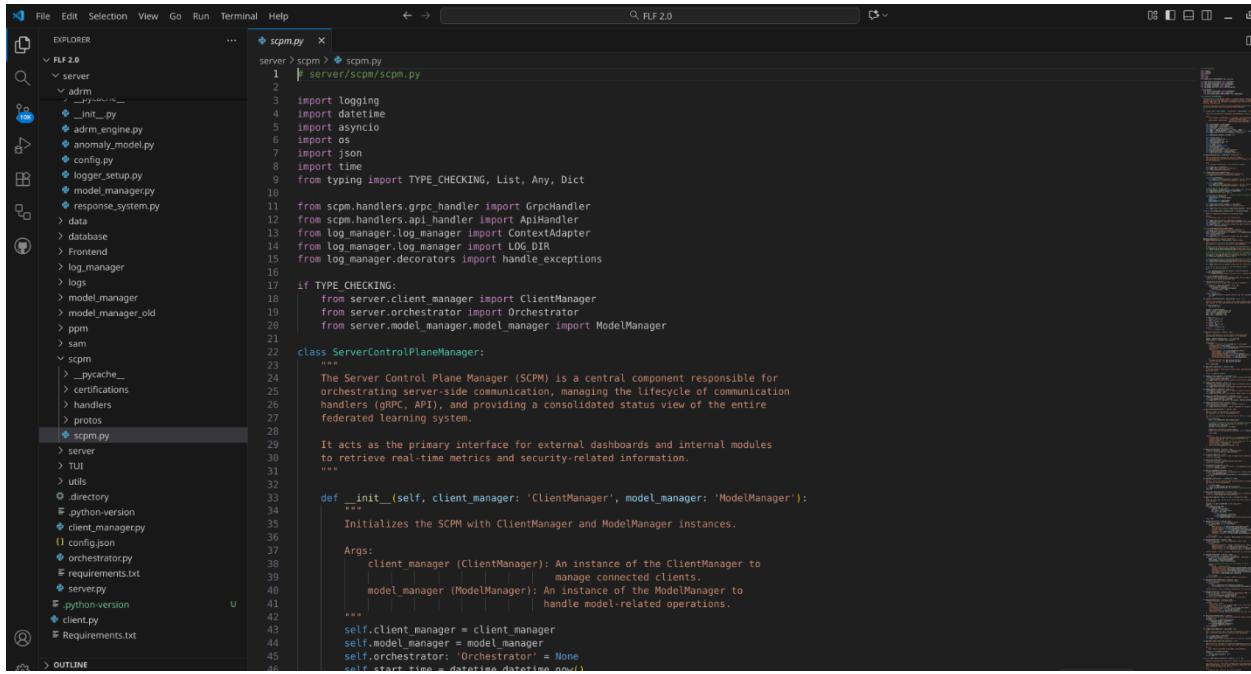


6.3 System Testing



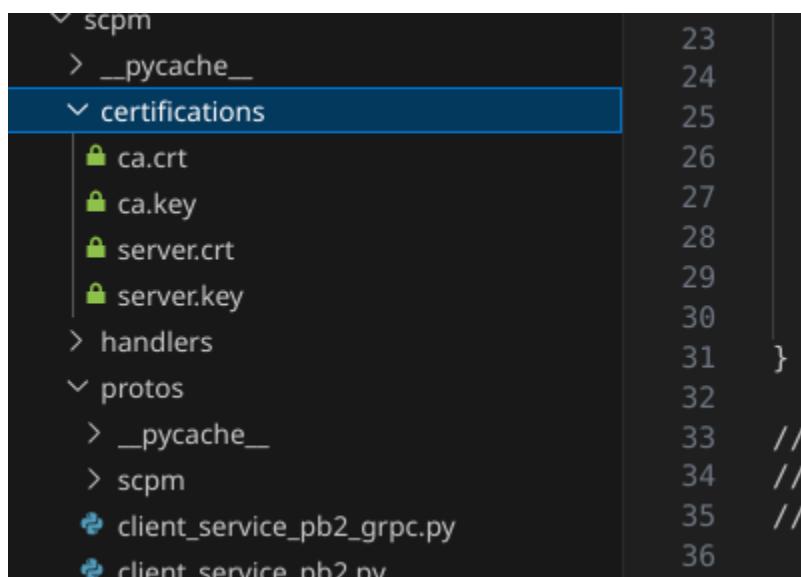
6.4 System Codes

SCPM Server



```

File Edit Selection View Go Run Terminal Help
EXPLORER server > scpm > scpm.py
server > scpm > handlers > base_handler.py
1 import logging
2 import typing
3 from log_manager.log_manager import ContextAdapter
4 from log_manager.log_manager import LOG_DIR
5 from log_manager.decorators import handle_exceptions
6
7 if TYPE_CHECKING:
8     from server.client_manager import ClientManager
9     from server.orchestrator import Orchestrator
10    from server.model_manager.model_manager import ModelManager
11
12 class ServerControlPlaneManager:
13     """
14         The Server Control Plane Manager (SCPM) is a central component responsible for
15             orchestrating server-side communication, managing the lifecycle of communication
16             handlers (gRPC, API), and providing a consolidated status view of the entire
17             federated learning system.
18
19     It acts as the primary interface for external dashboards and internal modules
20     to retrieve real-time metrics and security-related information.
21     """
22
23     def __init__(self, client_manager: 'ClientManager', model_manager: 'ModelManager'):
24         """
25             Initializes the SCPM with ClientManager and ModelManager instances.
26
27             Args:
28                 client_manager (ClientManager): An instance of the ClientManager to
29                     manage connected clients.
30
31                 model_manager (ModelManager): An instance of the ModelManager to
32                     handle model-related operations.
33
34             """
35
36         self.client_manager = client_manager
37         self.model_manager = model_manager
38         self.orchestrator: 'Orchestrator' = None
39         self.start_time = datetime.datetime.now()
40
41     def start_listening(self):
42         """
43             Starts listening for incoming connections.
44
45             Returns:
46                 None
47
48             Raises:
49                 NotImplementedError: Subclasses must implement this method.
50
51     def stop_listening(self):
52         """
53             Stops listening for incoming connections.
54
55             Returns:
56                 None
57
58             Raises:
59                 NotImplementedError: Subclasses must implement this method.
60
61     def handle_exception(self, exception):
62         """
63             Handles an exception that occurred during processing.
64
65             Args:
66                 exception (Exception): The exception that occurred.
67
68             Returns:
69                 None
70
71             Raises:
72                 None
73
74     def handle(self, request):
75         """
76             Handles a request from a client.
77
78             Args:
79                 request (Any): The request object.
80
81             Returns:
82                 Any
83
84             Raises:
85                 NotImplementedError: Subclasses must implement this method.
86
87     def handle_model_update(self, update):
88         """
89             Handles a model update from a client.
90
91             Args:
92                 update (Any): The model update object.
93
94             Returns:
95                 None
96
97             Raises:
98                 NotImplementedError: Subclasses must implement this method.
99
100            """
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
288
289
289
290
291
292
293
294
295
296
297
298
299
299
300
301
302
303
304
305
306
307
308
309
309
310
311
312
313
314
315
316
317
318
319
319
320
321
322
323
324
325
326
327
328
329
329
330
331
332
333
334
335
336
337
337
338
339
339
340
341
342
343
344
345
345
346
347
347
348
349
349
350
351
352
353
354
355
356
356
357
358
359
359
360
361
362
363
364
365
366
367
367
368
369
369
370
371
372
373
374
375
376
377
377
378
379
379
380
381
382
383
384
385
386
387
387
388
389
389
390
391
392
393
394
395
396
397
398
399
399
400
401
402
403
404
405
406
407
408
409
409
410
411
412
413
414
415
416
417
417
418
419
419
420
421
422
423
424
425
426
427
427
428
429
429
430
431
432
433
434
435
436
437
437
438
439
439
440
441
442
443
444
445
446
447
447
448
449
449
450
451
452
453
454
455
456
456
457
458
458
459
459
460
461
462
463
464
465
466
467
467
468
469
469
470
471
472
473
474
475
476
477
477
478
479
479
480
481
482
483
484
485
486
487
487
488
489
489
490
491
492
493
494
495
496
497
498
499
499
500
501
502
503
504
505
506
507
508
509
509
510
511
512
513
514
515
516
517
517
518
519
519
520
521
522
523
524
525
526
527
527
528
529
529
530
531
532
533
534
535
536
537
537
538
539
539
540
541
542
543
544
545
546
547
547
548
549
549
550
551
552
553
554
555
556
557
557
558
559
559
560
561
562
563
564
565
566
567
567
568
569
569
570
571
572
573
574
575
576
577
577
578
579
579
580
581
582
583
584
585
586
587
587
588
589
589
590
591
592
593
594
595
596
597
597
598
599
599
600
601
602
603
604
605
606
607
607
608
609
609
610
611
612
613
614
615
616
616
617
618
618
619
619
620
621
622
623
624
625
626
626
627
628
628
629
629
630
631
632
633
634
635
636
637
637
638
639
639
640
641
642
643
644
645
646
646
647
648
648
649
649
650
651
652
653
654
655
656
657
657
658
659
659
660
661
662
663
664
665
666
666
667
668
668
669
669
670
671
672
673
674
675
676
676
677
678
678
679
679
680
681
682
683
684
685
686
686
687
688
688
689
689
690
691
692
693
694
695
696
696
697
698
698
699
699
700
701
702
703
704
705
706
706
707
708
708
709
709
710
711
712
713
714
715
715
716
717
717
718
718
719
719
720
721
722
723
724
725
725
726
727
727
728
728
729
729
730
731
732
733
734
735
735
736
736
737
737
738
738
739
739
740
741
742
743
744
745
745
746
746
747
747
748
748
749
749
750
751
752
753
754
755
755
756
756
757
757
758
758
759
759
760
761
762
763
764
765
765
766
766
767
767
768
768
769
769
770
771
772
773
774
775
775
776
776
777
777
778
778
779
779
780
781
782
783
784
785
785
786
786
787
787
788
788
789
789
790
791
792
793
794
794
795
795
796
796
797
797
798
798
799
799
800
801
802
803
803
804
804
805
805
806
806
807
807
808
808
809
809
810
811
812
813
813
814
814
815
815
816
816
817
817
818
818
819
819
820
821
822
823
823
824
824
825
825
826
826
827
827
828
828
829
829
830
831
832
833
833
834
834
835
835
836
836
837
837
838
838
839
839
840
841
842
843
843
844
844
845
845
846
846
847
847
848
848
849
849
850
851
852
853
853
854
854
855
855
856
856
857
857
858
858
859
859
860
861
862
863
863
864
864
865
865
866
866
867
867
868
868
869
869
870
871
872
873
873
874
874
875
875
876
876
877
877
878
878
879
879
880
881
882
883
883
884
884
885
885
886
886
887
887
888
888
889
889
890
891
892
892
893
893
894
894
895
895
896
896
897
897
898
898
899
899
900
901
902
902
903
903
904
904
905
905
906
906
907
907
908
908
909
909
910
911
912
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1490
1491
1491
1492
1492
1493
1493
1494
1494
1495
1495
1496
1496
1497
1497
1498
1498
1499
1499
1500
1500
1501
1501
1502
1502
1503
1503
1504
1504
1505
1505
1506
1506
1507
1507
1508
1508
1509
1509
1510
1510
1511
1511
1512
1512
1513
1513
1514
1514
1515
1515
1516
1516
1517
1517
1518
1518
1519
1519
1520
1520
1521
1521
1522
1522
1523
1523
1524
1524
1525
1525
1526
1526
1527
1527
1528
1528
1529
1529
1530
1530
1531
1531
1532
1532
1533
1533
1534
1534
1535
1535
1536
1536
1537
1537
1538
1538
1539
1539
1540
1540
1541
1541
1542
1542
1543
1543
1544
1544
1545
1545
1546
1546
1547
1547
1548
1548
1549
1549
1550
1550
1551
1551
1552
1552
1553
1553
1554
1554
1555
1555
1556
1556
1557
1557
1558
1558
1559
1559
1560
1560
1561
1561
1562
1562
1563
15
```



Protos – server

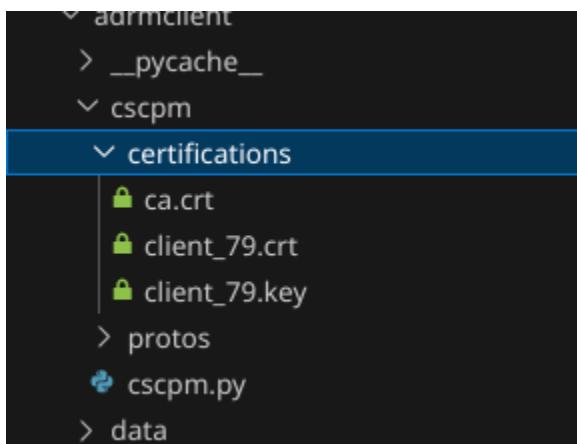
```
File Edit Selection View Go Run Terminal Help ↺ ↽ FLF 2.0

EXPLORER
  ...
  client_service.proto
  ...
  server > scpm > protos > client_service.proto
  1 syntax = "proto3";
  2
  3 package fl_service;
  4
  5 // =====
  6 // Service Definitions
  7 //
  8
  9 // Defines the Greeter service for the insecure registration channel.
10 service Greeter {
11   // A one-time RPC used to register a new client and issue a certificate.
12   rpc RegisterClient (ClientRegistrationRequest) returns (ClientRegistrationResponse) {
13     ...
14   }
15
16   // Defines the main gRPC service for secure client-server communication.
17   service ClientService {
18     // RPC for a client to register with the FL session after mTLS auth.
19     rpc RegisterClient (RegisterClientRequest) returns (RegisterClientResponse) {
20       ...
21     }
22     // RPC for client heartbeats and to check for new training rounds.
23     rpc SendHeartbeat (HeartbeatRequest) returns (HeartbeatResponse);
24
25     // RPC for a client to fetch the latest global model from the server.
26     rpc FetchModel (FetchModelRequest) returns (FetchModelResponse);
27
28     // RPC for a client to send its local model update (using HE).
29     rpc SendModelUpdate (SendModelUpdateRequest) returns (SendModelUpdateResponse) {
30       ...
31     }
32
33     // RPC for a client to send its secret shares to the server (using HE).
34     rpc SendModelUpdateShares (SendModelUpdateSharesRequest) returns (SendModelUpdateSharesResponse) {
35       ...
36     }
37
38     // Request for the one-time insecure registration to get a certificate.
39     message ClientRegistrationRequest {
40       string client_id = 1;
41       bytes certificate_signing_request = 2;
42       string registration_token = 3;
43     }
44
45     // Response for the one-time insecure registration.
46     message ClientRegistrationResponse {
47       bool success = 1;
48       string message = 2;
49     }
50   }
51 }

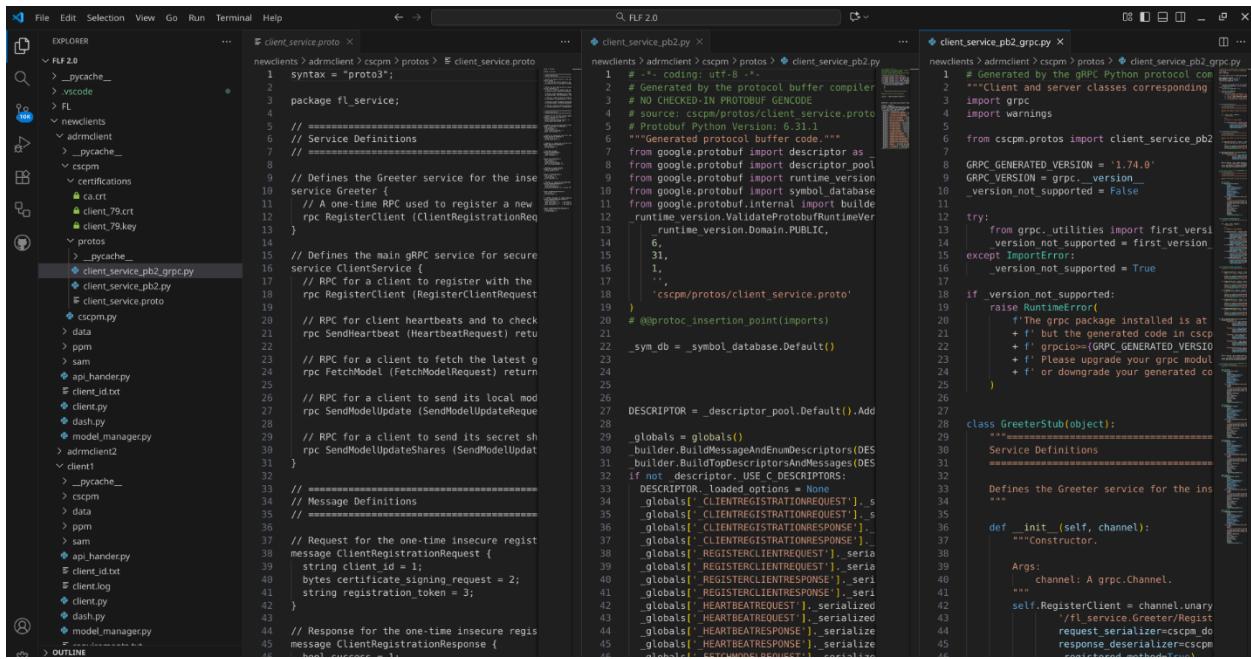
client_service.proto
  ...
  server > scpm > protos > client_service.proto
  1 syntax = "proto3";
  2
  3 package fl_service;
  4
  5 // =====
  6 // Service Definitions
  7 //
  8
  9 // Defines the Greeter service for the insecure registration channel.
10 service Greeter {
11   // A one-time RPC used to register a new client and issue a certificate.
12   rpc RegisterClient (ClientRegistrationRequest) returns (ClientRegistrationResponse) {
13     ...
14   }
15
16   // Defines the main gRPC service for secure client-server communication.
17   service ClientService {
18     // RPC for a client to register with the FL session after mTLS auth.
19     rpc RegisterClient (RegisterClientRequest) returns (RegisterClientResponse) {
20       ...
21     }
22     // RPC for client heartbeats and to check for new training rounds.
23     rpc SendHeartbeat (HeartbeatRequest) returns (HeartbeatResponse);
24
25     // RPC for a client to fetch the latest global model from the server.
26     rpc FetchModel (FetchModelRequest) returns (FetchModelResponse);
27
28     // RPC for a client to send its local model update (using HE).
29     rpc SendModelUpdate (SendModelUpdateRequest) returns (SendModelUpdateResponse) {
30       ...
31     }
32
33     // RPC for a client to send its secret shares to the server (using HE).
34     rpc SendModelUpdateShares (SendModelUpdateSharesRequest) returns (SendModelUpdateSharesResponse) {
35       ...
36     }
37
38     // Request for the one-time insecure registration to get a certificate.
39     message ClientRegistrationRequest {
40       string client_id = 1;
41       bytes certificate_signing_request = 2;
42       string registration_token = 3;
43     }
44
45     // Response for the one-time insecure registration.
46     message ClientRegistrationResponse {
47       bool success = 1;
48       string message = 2;
49     }
50   }
51 }
```

SCPM (Client SCPM) – Client

Certifications Create – Registered Client



Protos



The screenshot shows the FL 2.0 IDE with three tabs open:

- client_service.proto**: The original protocol buffer definition file.
- client_service_pb2.py**: The generated Python protocol buffer classes corresponding to the proto file.
- client_service_pb2_grpc.py**: The generated gRPC Python code.

The code in these files is as follows:

```

syntax = "proto3";
package fl_service;
// =====
// Service Definitions
// =====
// Defines the Greeter service for the insecure Greeter {
//   // A one-time RPC used to register a new client.
//   rpc RegisterClient (ClientRegistrationRequest)
//     // Defines the main gRPC service for secure service ClientService {
//       // RPC for a client to register with the
//       rpc RegisterClient (RegisterClientRequest)
//         // RPC for client heartbeats and to check
//         rpc SendHeartbeat (HeartbeatRequest) returns (HeartbeatResponse)
//         // RPC for a client to fetch the latest g
//         rpc FetchModel (FetchModelRequest) returns (FetchModelResponse)
//         // RPC for a client to send its local model
//         rpc SendModelUpdate (SendModelUpdateRequest)
//         // RPC for a client to send its secret share
//         rpc SendModelUpdateShares (SendModelUpdateSharesRequest)
//       }
// =====
// Message Definitions
// =====
// Request for the one-time insecure registration
message ClientRegistrationRequest {
  string client_id = 1;
  bytes certificate_signing_request = 2;
  string registration_token = 3;
}
// Response for the one-time insecure registration
message ClientRegistrationResponse {
  bool success = 1;
}

// Generated by the protocol buffer compiler
// from google.protobuf import descriptor as _descriptor
// from google.protobuf import message as _message
// from google.protobuf import symbol_database as _symbol_database
// from google.protobuf import internal.importer as _internal_importer
// from google.protobuf import runtime_version
// runtime_version.ValidateRuntimeVersion(
//   runtime_version.Domain.PUBLIC,
//   6,
//   31,
//   1,
//   '',
//   'cscpm/protos/client_service.proto'
// )
// @proto_insertion_point(importss)
_sym_db = _symbol_database.Default()

// Generated by the gRPC Python protocol compiler
// from cscpm.protos import client_service_pb2
// GRPC GENERATED VERSION = '1.74.0'
// GRPC VERSION = grpc._version_
// version_not_supported = False
try:
  from grpc import utilities
  _version_not_supported = first_version < grpc._version_
except ImportError:
  _version_not_supported = True

if _version_not_supported:
  raise ImportError(
    f'The gRPC package installed is at {first_version}, but the generated code in cscpm/protos/client_service.proto was generated with gRPC GENERATED VERSION {GRPC_GENERATED_VERSION}. Please upgrade your gRPC module or downgrade your generated code.'
  )

class GreeterStub(object):
    """Service Definitions
    =====
    Defines the Greeter service for the insecure Greeter
    """
    def __init__(self, channel):
        """Constructor.
        Args:
            channel: A grpc.Channel.
        """
        self.RegisterClient = channel.unary_unary(
            '/fl_service.Greeter/RegisterClient',
            request_serializer=client_service_pb2.ClientRegistrationRequest.SerializeToString(),
            response_deserializer=client_service_pb2.ClientRegistrationResponse.DeserializeToString()
        )

```

CSCPM

File View Go Run Terminal Help

csmp.py

```

1  from enum import Enum
2  from typing import Dict, Any, Callable, Optional
3
4  # Enums for different connection methods
5  class ConnectionMethod(Enum):
6      GRPC = "grpc"
7      TLS_SOCKETS = "tls_sockets"
8      RAW_SOCKETS = "raw_sockets"
9      HTTP = "http"
10     # Add other production-grade connection methods here (e.g., MQTT, WebSockets with specific protocols)
11
12 class ConnectionPolicyManager:
13     """
14         Manages and applies different connection policies and methods.
15         This class would encapsulate the logic for establishing, managing,
16         and closing connections using various protocols (gRPC, TLS Sockets, etc.).
17     """
18
19     def __init__(self):
20         # A dictionary to store "connect" functions for different methods
21         self._connection_handlers: Dict[ConnectionMethod, Callable[[Dict[str, Any]], Any]] = {
22             ConnectionMethod.GRPC: self._connect_grpc,
23             ConnectionMethod.TLS_SOCKETS: self._connect_tls_sockets,
24             ConnectionMethod.RAW_SOCKETS: self._connect_raw_sockets,
25             ConnectionMethod.HTTP: self._connect_http,
26         }
27         # A dictionary to store "send" functions for different methods
28         self._send_handlers: Dict[ConnectionMethod, Callable[[Any, Any], bool]] = {
29             ConnectionMethod.GRPC: self._send_grpc,
30             ConnectionMethod.TLS_SOCKETS: self._send_tls_sockets,
31             ConnectionMethod.RAW_SOCKETS: self._send_raw_sockets,
32             ConnectionMethod.HTTP: self._send_http,
33         }
34         # A dictionary to store "disconnect" functions for different methods
35         self._disconnect_handlers: Dict[ConnectionMethod, Callable[[Any], None]] = {
36             ConnectionMethod.GRPC: self._disconnect_grpc,
37             ConnectionMethod.TLS_SOCKETS: self._disconnect_tls_sockets,
38             ConnectionMethod.RAW_SOCKETS: self._disconnect_raw_sockets,
39             ConnectionMethod.HTTP: self._disconnect_http,
40         }
41
42     def register_connection_handler(self, method: ConnectionMethod, handler_func: Callable[[Dict[str, Any]], Any]):
43         """Registers a custom connection handler for a given method."""
44         self._connection_handlers[method] = handler_func
45
46

```

Log Manager

File Edit Selection View Go Run Terminal Help

log_manager.py

```

1  # utils/log_manager.py
2
3  import logging
4  import os
5  from pythonjsonlogger.json import JsonFormatter
6  from logging.handlers import TimedRotatingFileHandler
7
8  # ... FIX START: Define a robust, absolute path for the log directory
9  # Get the directory where the main server script is likely located.
10 # It assumes a structure where the log_manager package is at the same
11 # project root as os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
12 LOG_DIR = os.path.join(__projct_root, 'server', 'logs')
13 # ... FIX END ...
14
15 class ContextAdapter(logging.LoggerAdapter):
16     def process(self, msg, kwargs):
17         if 'extra' not in kwargs:
18             kwargs['extra'] = {}
19         kwargs['extra'].update(self.extra)
20         return msg, kwargs
21
22     def configure_root_logging(logger: logging.Logger):
23         """
24             Configures the given logger (intended for the root logger) with a
25             class existing handlers to prevent duplicate logs.
26         """
27         logger.setLevel(os.getenv("LOG_LEVEL", "INFO").upper())
28
29         if logger.handlers:
30             for handler in logger.handlers:
31                 logger.removeHandler(handler)
32
33         console_handler = logging.StreamHandler()
34         console_formatter = logging.Formatter(
35             "%(asctime)s - %(levelname)s - %(name)s - %(funcName)s:%(lineNo)d"
36         )
37         console_handler.setFormatter(console_formatter)
38         logger.addHandler(console_handler)
39
40         logging.getLogger('websockets.protocol').setLevel(logging.WARNING)
41         logging.getLogger('grpc').setLevel(logging.WARNING)
42         logging.getLogger('asyncio').setLevel(logging.WARNING)
43
44     def add_json_file_handler(logger_name: str, log_file_name: str):
45

```

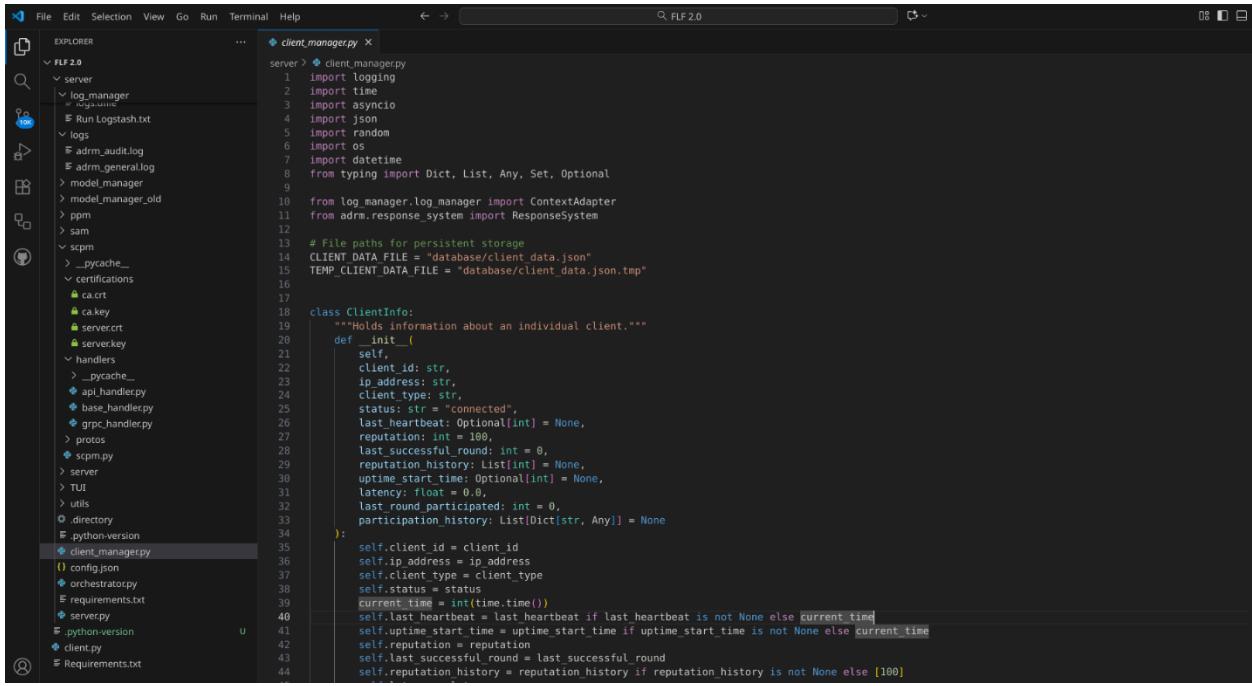
model_metrics_history.json

```

1  [
2  ]
3  [
4  ]
5  [
6  ]
7  [
8  ]
9  [
10 ]
11 [
12 ]
13 [
14 ]
15 [
16 ]
17 [
18 ]
19 [
20 ]
21 [
22 ]
23 [
24 ]
25 [
26 ]
27 [
28 ]
29 [
30 ]
31 [
32 ]
33 [
34 ]
35 [
36 ]
37 [
38 ]
39 [
40 ]
41 [
42 ]
43 [
44 ]

```

Client Manager



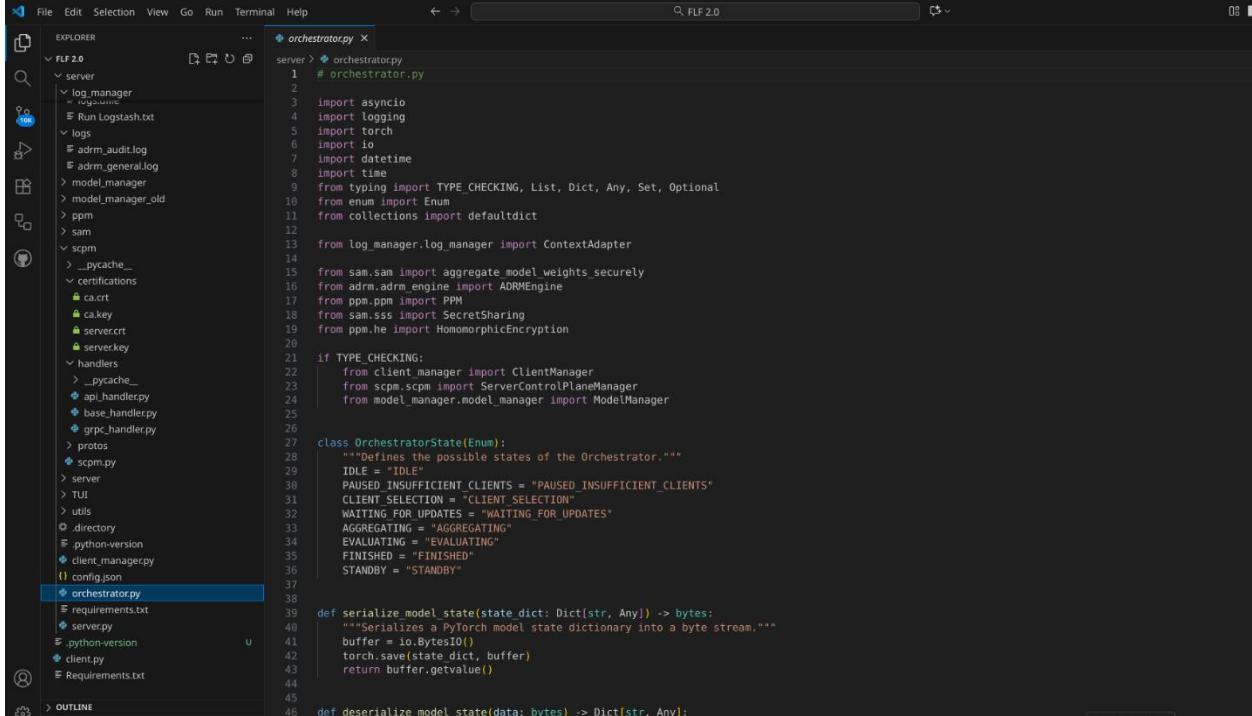
```

client_manager.py

server > client_manager.py
1 import logging
2 import time
3 import asyncio
4 import json
5 import random
6 import os
7 import datetime
8 from typing import Dict, List, Any, Set, Optional
9
10 from log_manager.log_manager import ContextAdapter
11 from adm.response_system import ResponseSystem
12
13 # File paths for persistent storage
14 CLIENT_DATA_FILE = "database/client_data.json"
15 TEMP_CLIENT_DATA_FILE = "database/client_data.json.tmp"
16
17
18 class ClientInfo:
19     """Holds information about an individual client."""
20     def __init__(self,
21                  client_id: str,
22                  ip_address: str,
23                  client_type: str,
24                  status: str = "connected",
25                  last_heartbeat: Optional[int] = None,
26                  reputation: int = 100,
27                  last_successful_round: int = 0,
28                  reputation_history: List[int] = None,
29                  uptime_start_time: Optional[int] = None,
30                  latency: float = 0.0,
31                  last_round_participated: int = 0,
32                  participation_history: List[Dict[str, Any]] = None):
33         self.client_id = client_id
34         self.ip_address = ip_address
35         self.client_type = client_type
36         self.status = status
37         self.current_time = int(time.time())
38         self.last_heartbeat = last_heartbeat if last_heartbeat is not None else current_time
39         self.uptime_start_time = uptime_start_time if uptime_start_time is not None else current_time
40         self.reputation = reputation
41         self.last_successful_round = last_successful_round
42         self.reputation_history = reputation_history if reputation_history is not None else [100]
43
44
45

```

Orchestrator



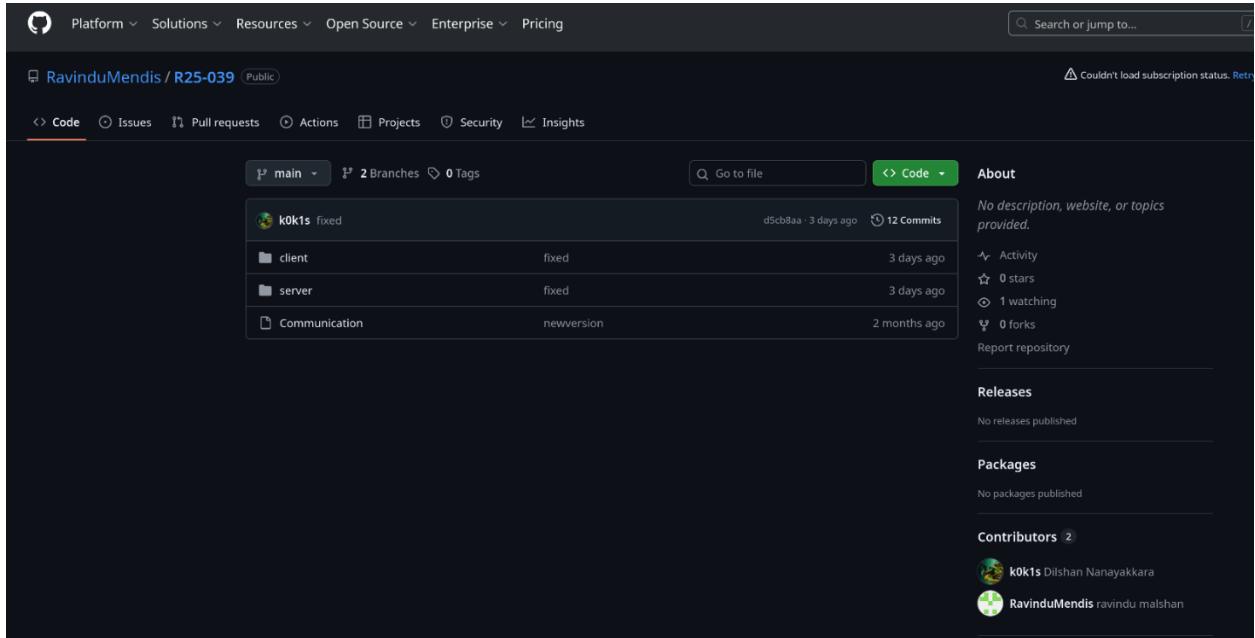
```

orchestrator.py

server > orchestrator.py
1 # orchestrator.py
2
3 import asyncio
4 import logging
5 import torch
6 import io
7 import datetime
8 import time
9 from typing import TYPE_CHECKING, List, Dict, Any, Set, Optional
10 from enum import Enum
11 from collections import defaultdict
12
13 from log_manager.log_manager import ContextAdapter
14
15 from sam.sam import aggregate_model_weights_securely
16 from adm.adm_engine import ADMEngine
17 from ppm.ppm import PPM
18 from sam.sss import SecretSharing
19 from ppm.he import HomomorphicEncryption
20
21 if TYPE_CHECKING:
22     from client_manager import ClientManager
23     from scpm.scp import ServerControlPlaneManager
24     from model_manager.model_manager import ModelManager
25
26
27 class OrchestratorState(Enum):
28     """Defines the possible states of the Orchestrator."""
29     IDLE = "IDLE"
30     PAUSED_INSUFFICIENT_CLIENTS = "PAUSED_INSUFFICIENT_CLIENTS"
31     CLIENT_SELECTION = "CLIENT_SELECTION"
32     WAITING_FOR_UPDATES = "WAITING_FOR_UPDATES"
33     AGGREGATING = "AGGREGATING"
34     EVALUATING = "EVALUATING"
35     FINISHED = "FINISHED"
36     STANDBY = "STANDBY"
37
38     def serialize_model_state(state_dict: Dict[str, Any]) -> bytes:
39         """Serializes a PyTorch model state dictionary into a byte stream."""
40         buffer = io.BytesIO()
41         torch.save(state_dict, buffer)
42         return buffer.getvalue()
43
44
45     def deserialize_model_state(data: bytes) -> Dict[str, Any]:
46

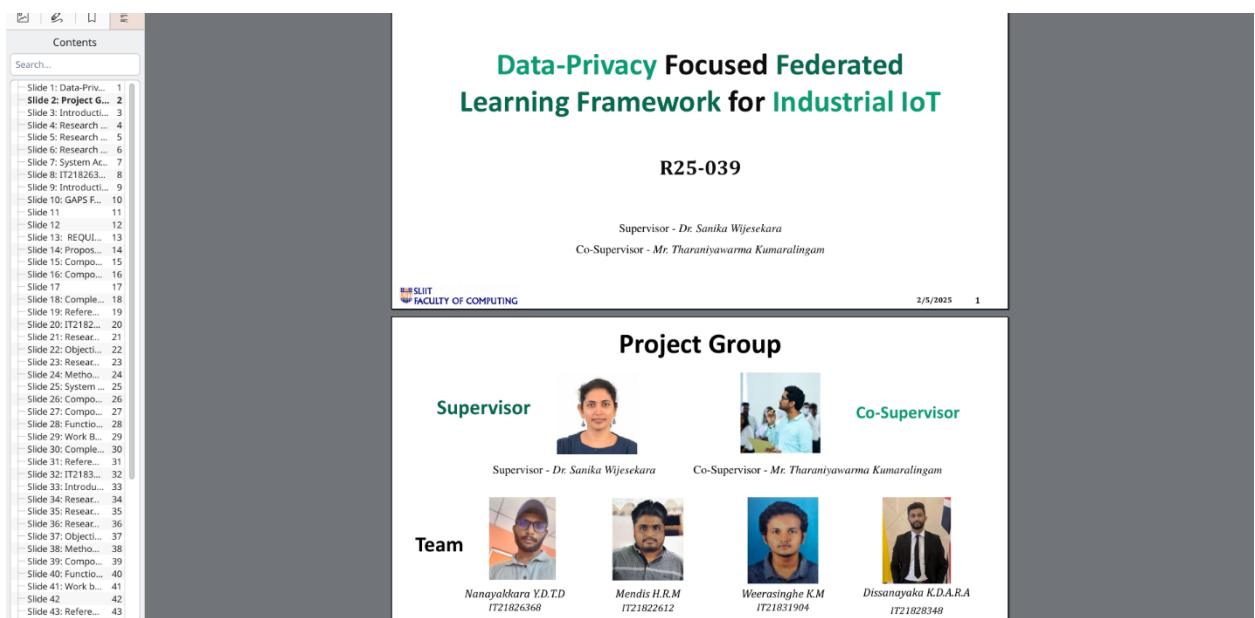
```

7. GitHub Upload



8. Documentation

8.1 Proposal



Data-Privacy Focused Federated Learning Framework for Industrial IoT

R25-039

Supervisor - Dr. Sanika Wijesekara
Co-Supervisor - Mr. Tharaniyawarma Kumaralingam

Project Group

Supervisor		Co-Supervisor	
-------------------	--	----------------------	--

Team

--	--	--	--

Table of Contents

- Slide 1: Data-Priv... 1
- Slide 2: Project ... 2
- Slide 3: Introduc... 3
- Slide 4: Research ... 4
- Slide 5: Research ... 5
- Slide 6: Research ... 6
- Slide 7: System Arc... 7
- Slide 8: IT218263... 8
- Slide 9: Introduc... 9
- Slide 10: GAPS ... 10
- Slide 11: ... 11
- Slide 12: ... 12
- Slide 13: REQU... 13
- Slide 14: Propos... 14
- Slide 15: Compo... 15
- Slide 16: Compo... 16
- Slide 17: ... 17
- Slide 18: Comple... 18
- Slide 19: Refere... 19
- Slide 20: IT2182... 20
- Slide 21: Resear... 21
- Slide 22: Objecti... 22
- Slide 23: Resear... 23
- Slide 24: Metho... 24
- Slide 25: System ... 25
- Slide 26: Compo... 26
- Slide 27: Compatib... 27
- Slide 28: Protectio... 28
- Slide 29: Work B... 29
- Slide 30: Comple... 30
- Slide 31: Refere... 31
- Slide 32: IT2183... 32
- Slide 33: Introduc... 33
- Slide 34: Resear... 34
- Slide 35: Resear... 35
- Slide 36: Resear... 36
- Slide 37: Protecti... 37
- Slide 38: Metho... 38
- Slide 39: Compo... 39
- Slide 40: Functio... 40
- Slide 41: Work b... 41
- Slide 42: ... 42
- Slide 43: Refere... 43

8.2 Presentation 1

Contents

Search...

- Slide 1: Data-Priv... 1
- Slide 2: Project G... 2
- Slide 3: Introduc... 3
- Slide 4: Researc... 4
- Slide 5: Research ... 5
- Slide 6: Research ... 6
- Slide 7: System Ar... 7
- Slide 8: IT218263... 8
- Slide 9: Introduc... 9
- Slide 10: GAPS F... 10
- Slide 11 11
- Slide 12 12
- Slide 13: REQUIS... 13
- Slide 14: Propos... 14
- Slide 15: Compo... 15
- Slide 16: Compo... 16
- Slide 17 17
- Slide 18: Comple... 18
- Slide 19: Refere... 19
- Slide 20: IT2182... 20
- Slide 21: Compo... 21
- Slide 22: Objecti... 22
- Slide 23: Resear... 23
- Slide 24: Metho... 24**
- Slide 25: System ... 25
- Slide 26: Compo... 26
- Slide 27: Compo... 27
- Slide 28: Functio... 28
- Slide 29: Visual B... 29
- Slide 30: Compon... 30
- Slide 31: Refere... 31
- Slide 32: IT2183... 32
- Slide 33: Introduc... 33
- Slide 34: Resear... 34
- Slide 35: Resear... 35
- Slide 36: Resear... 36
- Slide 37: Objecti... 37
- Slide 38: Compon... 38
- Slide 39: Compo... 39
- Slide 40: Function... 40
- Slide 41: Work b... 41

There is a lack of large-scale IIoT datasets for testing privacy-preserving techniques under real-world conditions. Existing research often relies on synthetic data, limiting the generalizability of results.

Regulatory compliance Privacy-preserving methods need to be compliant with existing data protection laws like GDPR, but there's a lack of specific guidelines for IIoT systems.

Resistance to privacy attacks While current privacy-preserving methods are in place, the robustness of these methods against evolving privacy attacks in IIoT systems remains insufficiently addressed.

Energy consumption and efficiency Many privacy-preserving methods are computationally intensive, posing significant challenges to resource-constrained IIoT devices, affecting their energy efficiency.

IT21822612 | Mendis H.R.M. | R25-039 23

Methodology

Approach:

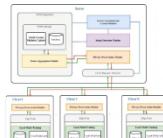
- Analyze existing FL privacy vulnerabilities.
- Combine HE and DP for enhanced privacy.
- Optimize techniques for IIoT-specific constraints.
- Validate Using real-world Datasets

Key Techniques:

- **Homomorphic Encryption (HE):** Encrypts gradients, allowing computations on encrypted data without decrypting it. Prevents data leakage even if adversaries intercept communications.
- **Differential Privacy (DP):** Ensure that individual data points cannot be separated by adding controlled noise to gradients. Balances model accuracy with privacy.

IT21822612 | Mendis H.R.M. | R25-039 24

System Architecture



8.3 Presentation 2

Contents

Search...

- > OVERALL 1
- > ADMR COMPONENTS 15
- > COMPONENT 2 PRI... 23
- > COMPONENT 3 SEC... 33
- > COMPONENT 4 SCPM 40
- > GENERAL END SLIDES 48

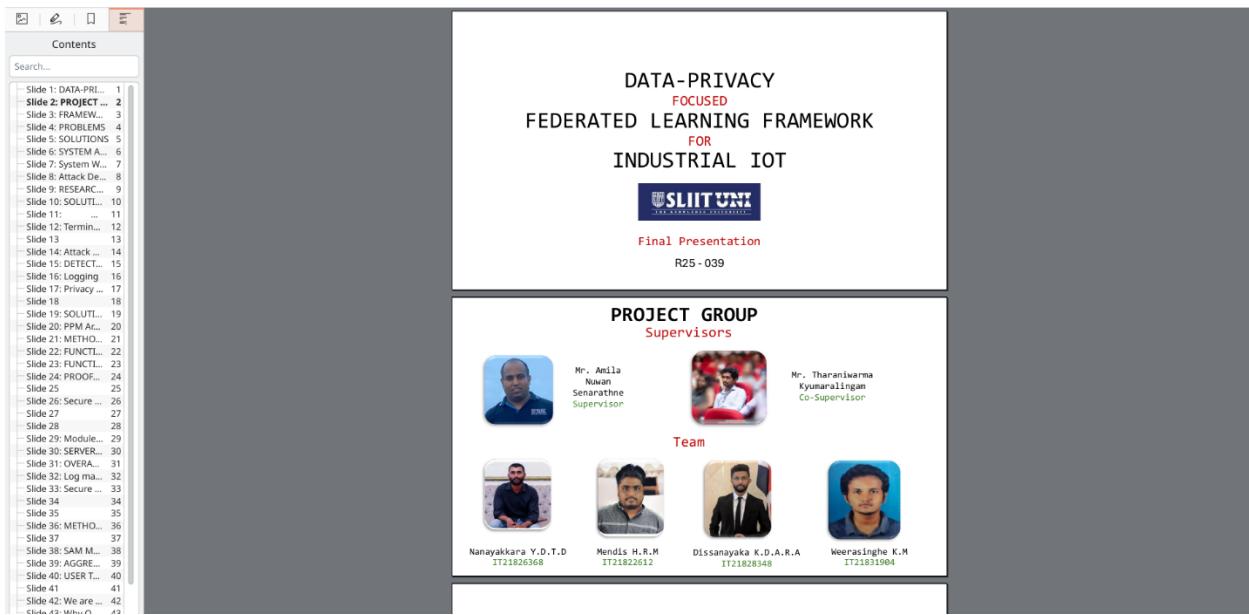
DATA-PRIVACY FOCUSED FEDERATED LEARNING FRAMEWORK FOR INDUSTRIAL IOT

R25 - 039

PROJECT GROUP

Team	Supervisors
 <i>Nanayakkara Y.D.T.D</i> <i>IT21826368</i>	 <i>Mr. Amila Nuwan Senarathne</i> <i>Supervisor</i>

8.4 Final Presentation



The slide displays the title of the presentation:

**DATA-PRIVACY
FOCUSSED
FEDERATED LEARNING FRAMEWORK
FOR
INDUSTRIAL IOT**

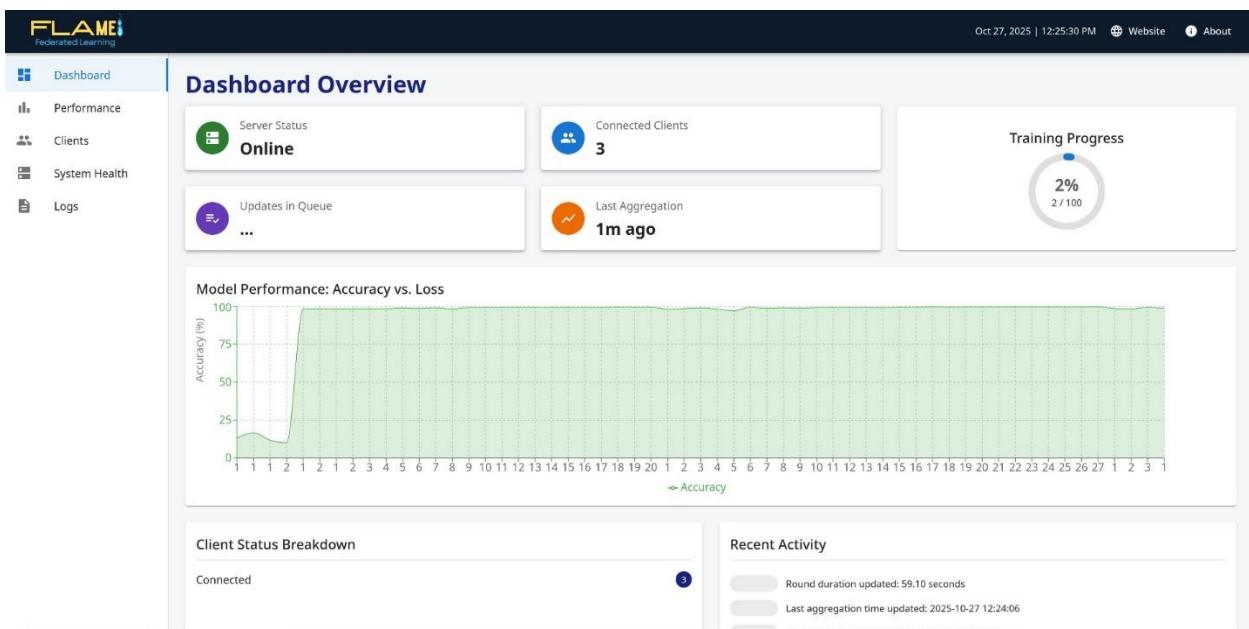
Below the title is the SLIIT UNI logo. The text "Final Presentation" and "R25 - 039" are also present.

On the left side, there is a sidebar containing a table of contents with 42 slides, ranging from Slide 1 to Slide 42. The sidebar also includes a search bar and a "Contents" button.

The right side of the slide shows the "PROJECT GROUP" section, which includes "Supervisors" and "Team". The "Supervisors" section lists Mr. Amila Nuwan Senarathne (Supervisor) and Mr. Tharanikarma Kyunaratnasingham (Co-Supervisor). The "Team" section lists four members: Nanayakkara Y.D.T (IT21826368), Mendis H.R.M (IT2182612), Dissanayaka K.D.A.R.A (IT21828348), and Weerasinghe K.M (IT21831904).

8.5 Final Product

Web Portal Frontend



The dashboard overview section includes the following metrics:

- Server Status: Online
- Connected Clients: 3
- Updates in Queue: ...
- Last Aggregation: 1m ago

A circular progress bar indicates Training Progress at 2% (2/100).

The Model Performance section shows a graph of Accuracy vs. Loss. The accuracy starts around 10%, drops to near 0% at step 1, and then rises sharply to nearly 100% by step 2, remaining stable thereafter.

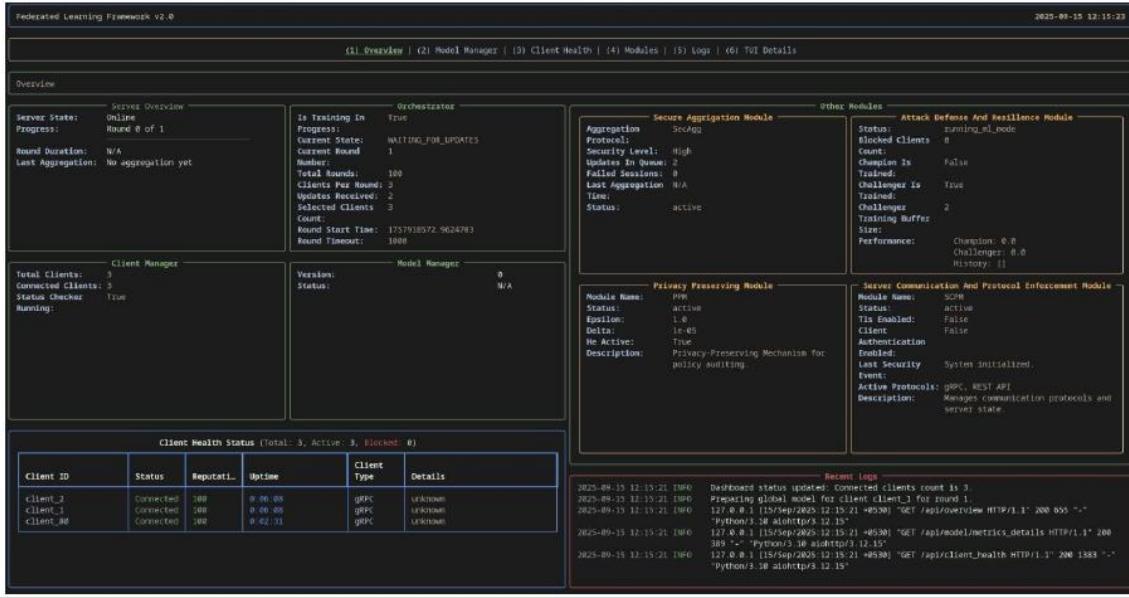
The Client Status Breakdown shows all clients as Connected.

The Recent Activity section displays the following updates:

- Round duration updated: 59.10 seconds
- Last aggregation time updated: 2025-10-27 12:24:06
- Dashboard status updated: 2 updates in queue

Terminal User interface (Frontend)

Proposed System – Frontend Terminal User Interface(TUI)



The screenshot displays the 'Federated Learning Framework v2.0' interface with the following sections:

- Overview:**
 - Server State:** Online, Progress: Round 0 of 1.
 - Round Duration:** N/A, Last Aggregation: No aggregation yet.
 - Orchestrator:** Is Training In Progress: True, Current Round: 1, Total Rounds: 100, Clients Per Round: 2, Selected Clients: 3, Count: 3, Round Start Time: 1757918572.9624783, Round Timeout: 1000.
 - Secure Aggregation Module:** Status: True, Aggregation Protocol: SCAGQ, Security Level: High, Updates In Queue: 2, Failed Sessions: 0, Last Aggregation: N/A, Total: 0, Status: active.
 - Attack Defense And Resilience Module:** Status: running_ml_node, Blocked Clients: 0, Count: 0, Champion Is: False, Challenger Is: True, Total: 0, Champion: 2, Challenger: 2, Training Buffer Size: 0, Performance: Champion: 0.0, Challenger: 0.0, History: [].
 - Client Manager:** Total Clients: 3, Connected Clients: 3, Status Checker: True, Running:.
 - Model Manager:** Version: 0, Status: N/A.
 - Privacy Preserving Module:** Module Name: PPM, Status: active, Epsilon: 1.0, Delta: 1e-05, Is Active: True, Description: Privacy-Preserving Mechanism for policy auditing.
 - Server Communication And Protocol Enforcement Module:** Module Name: SPP, Status: active, TLS Enabled: False, Client: False, Authentication: Enabled, Local Security: System initialized, Event: Active Protocols: gRPC, REST API, Description: Manage communication protocols and server state.
- Client Health Status (Total: 3, Active: 3, Blocked: 0):**

Client ID	Status	Reputati..	Uptime	Client Type	Details
Client_2	Connected	100	0:00:08	gRPC	Unknown
Client_1	Connected	100	0:00:08	gRPC	Unknown
Client_0	Connected	100	0:02:31	gRPC	Unknown
- Recent Logs:**
 - 2025-09-15 12:15:21 INFO Dashboard status updated: Connected clients count is 3.
 - 2025-09-15 12:15:21 INFO Preserving global model for client client_1 for round 1.
 - 2025-09-15 12:15:21 INFO 127.0.0.1 [15/Sep/2025:12:15:21 +0530] "GET /api/overview HTTP/1.1" 200 655 "-"
 - 2025-09-15 12:15:21 INFO 127.0.0.1 [15/Sep/2025:12:15:21 +0530] "GET /api/model/metrics_details HTTP/1.1" 200 389 "-" Python/3.10 aiohttp/3.22.5"
 - 2025-09-15 12:15:21 INFO 127.0.0.1 [15/Sep/2025:12:15:21 +0530] "GET /api/client_health HTTP/1.1" 200 1383 "-" Python/3.10 aiohttp/3.22.5"

8.6 Research Paper

III. Conference Apperence

To Tharindu D Nanayakkara <dilz.nanayakkara@gmail.com> @

10/29/25, 11:27 AM

Acceptance Notification

Dear Tharindu D Nanayakkara,

Congratulations! We are pleased to inform you that your paper has been accepted as a regular paper to be presented at the 7th International Conference on Advancements in Computing 2025.

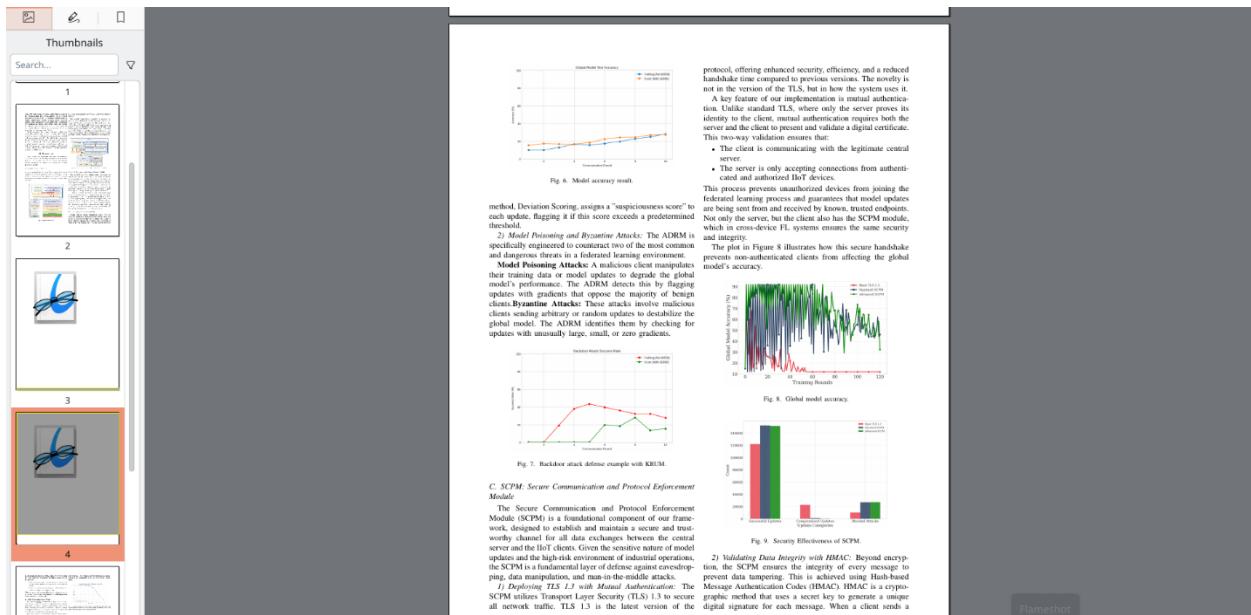
Paper ID: 469

Paper Title: Data-privacy based Federated Learning Framework for Industrial IOT

Please visit <https://cmt3.research.microsoft.com/7ICAC2025/Submission/Index> to view the reviews given during the double-blind review process.

When preparing the camera-ready version of your paper, please address all the review comments and follow the camera-ready guidelines given in the <https://icac.lk/for-authors>

Please note that the camera-ready deadline is 10th November 2025 and camera-ready submission portal on CMT will be available starting from 22nd October 2025.



9. CDAP upload

 **CDAPSubmissionCloud** 

Private group

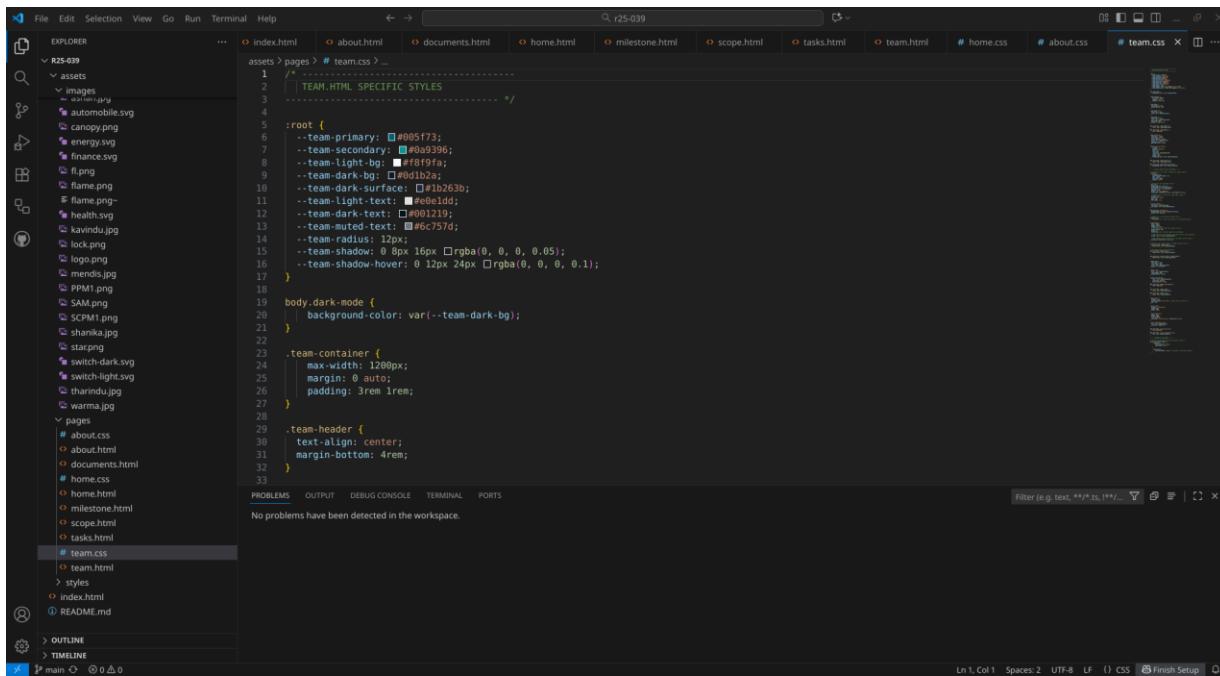
+ New  Upload  Edit in grid view  Share  Copy link  Add shortcut to OneDrive  Download  Export to Excel  Automate  Integrate  Sync

2025RegCloud > R25-039-Students

Name	Modified	Modified By
1. Project Proposal	January 27	Tharaniyawarma Kumaralingam
2. Progress Presentation - 1	January 27	Tharaniyawarma Kumaralingam
3. Progress Presentation - 2	January 27	Tharaniyawarma Kumaralingam
4. Research Paper	January 27	Tharaniyawarma Kumaralingam
5. Final Report & Presentation	January 27	Tharaniyawarma Kumaralingam
6. Check List Documents	April 29	CDAP SLIIT
7. Website	January 27	Tharaniyawarma Kumaralingam
8. Log Book	January 27	Tharaniyawarma Kumaralingam
Marking Schemes	January 27	Tharaniyawarma Kumaralingam
Project Registration Documents	January 27	Tharaniyawarma Kumaralingam
Panel Comments for the Students.xlsx	September 20	CDAP SLIIT

10. Website

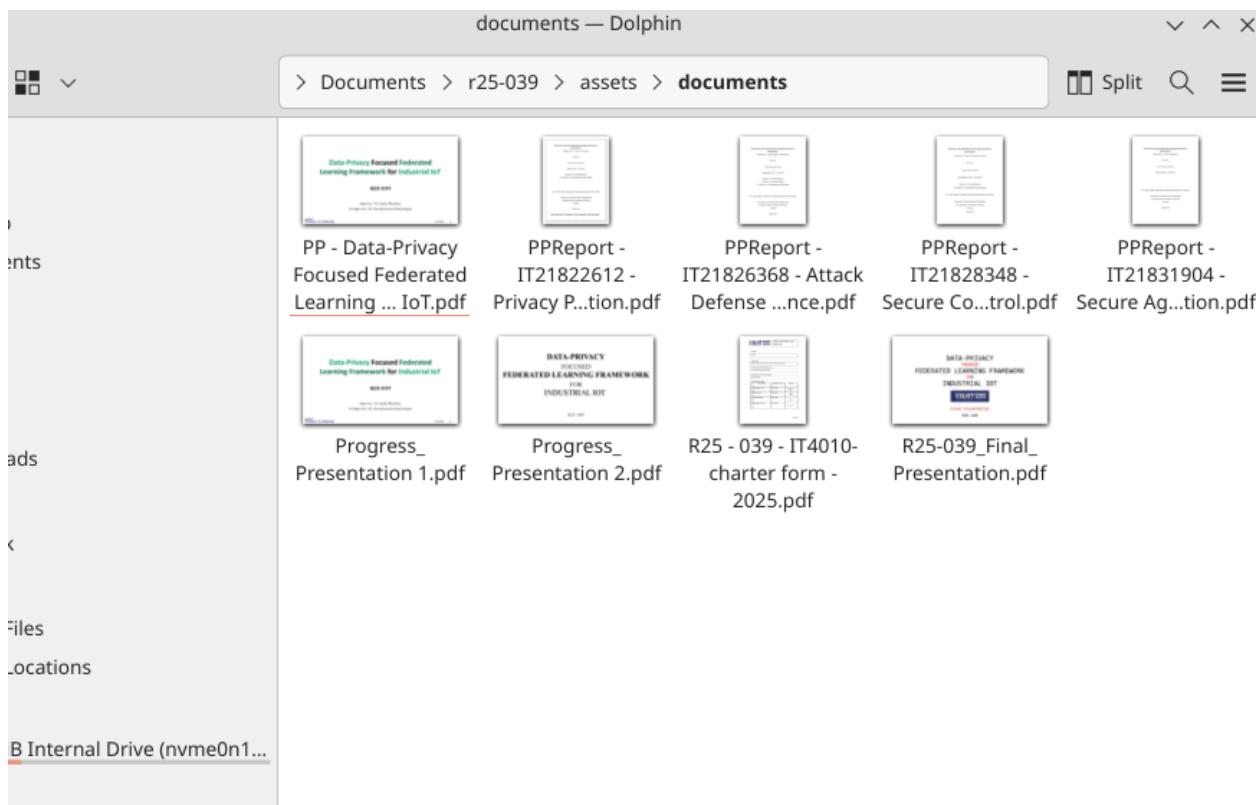
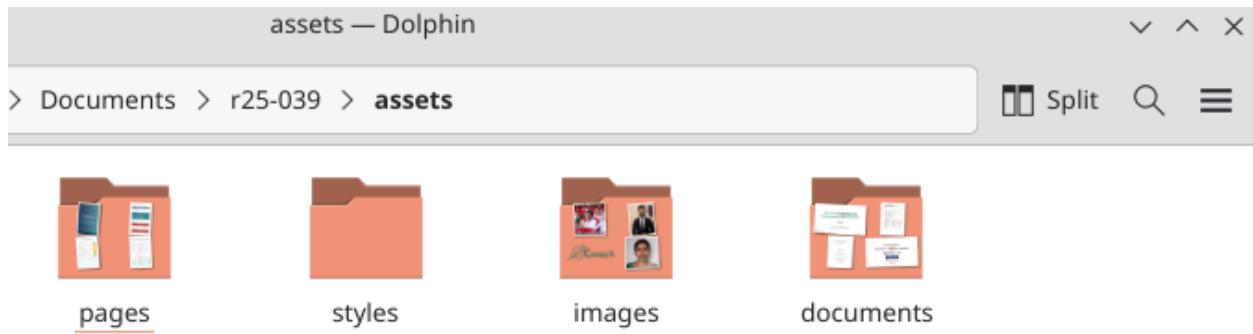
10.1 Development

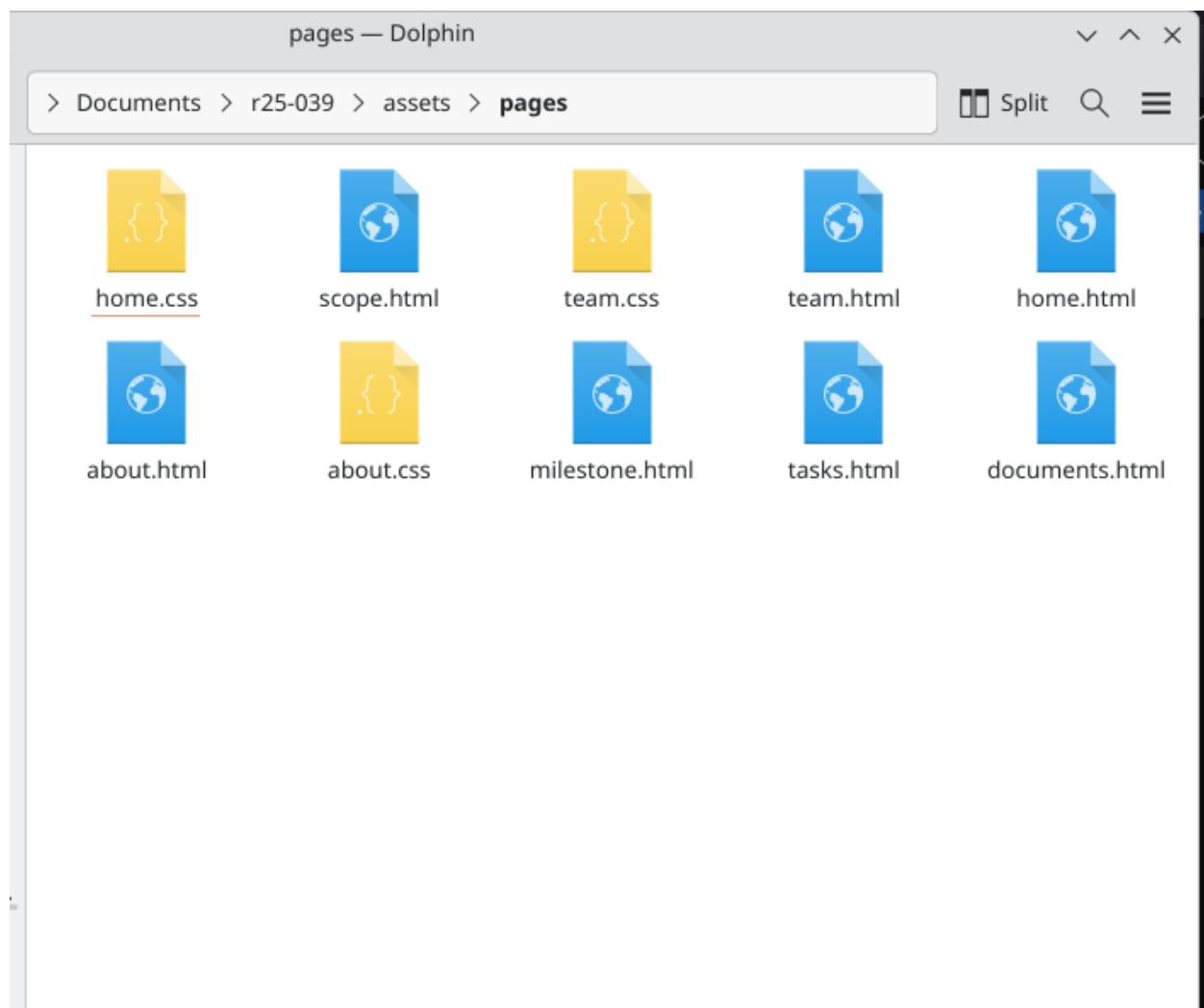


```

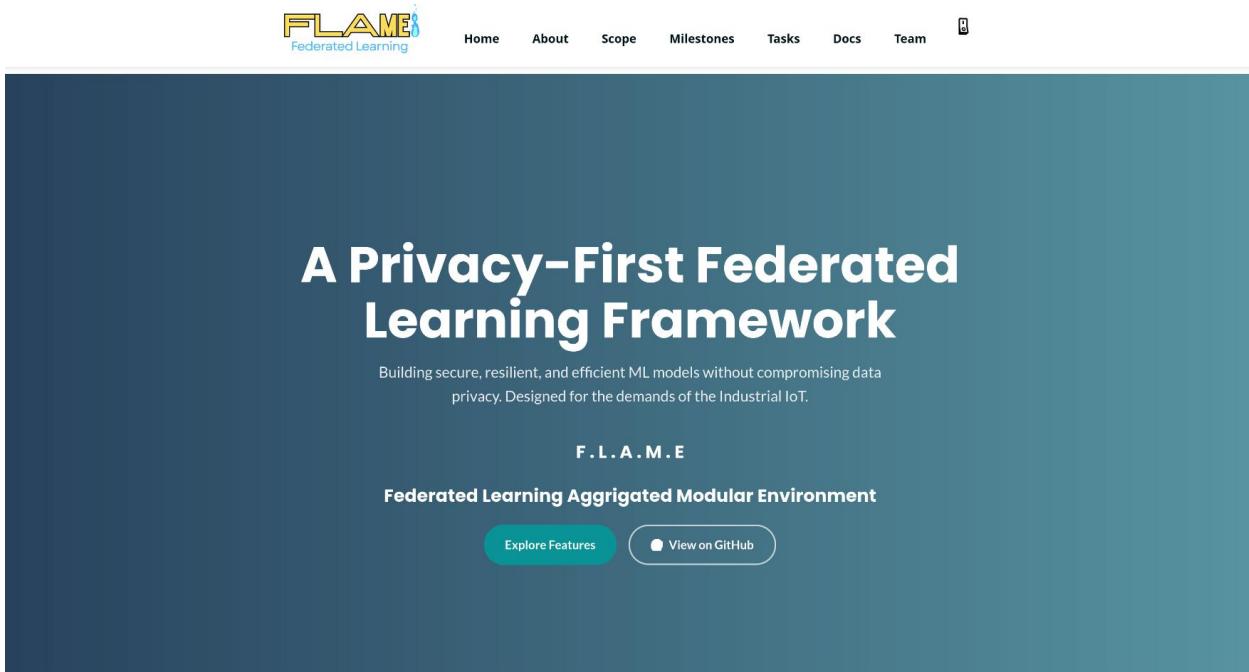
1  /* TEAM.HTML SPECIFIC STYLES
2   */
3
4  :root {
5      --team-primary: #005f73;
6      --team-secondary: #0a9396;
7      --team-light-bg: #f8f9fa;
8      --team-dark-bg: #d1b2e6;
9      --team-dark-surface: #1b263b;
10     --team-light-text: #e6e0d0;
11     --team-dark-text: #001219;
12     --team-soft-text: #e6c75d;
13     --team-radius: 12px;
14     --team-shadow: 0 8px 16px rgba(0, 0, 0, 0.05);
15     --team-shadow-hover: 0 12px 24px rgba(0, 0, 0, 0.1);
16 }
17
18 body.dark-mode {
19     background-color: var(--team-dark-bg);
20 }
21
22 .team-container {
23     max-width: 1100px;
24     margin: 0 auto;
25     padding: 3rem 0;
26 }
27
28 .team-header {
29     text-align: center;
30     margin-bottom: 4rem;
31 }
32
33

```





10.2 Finalize

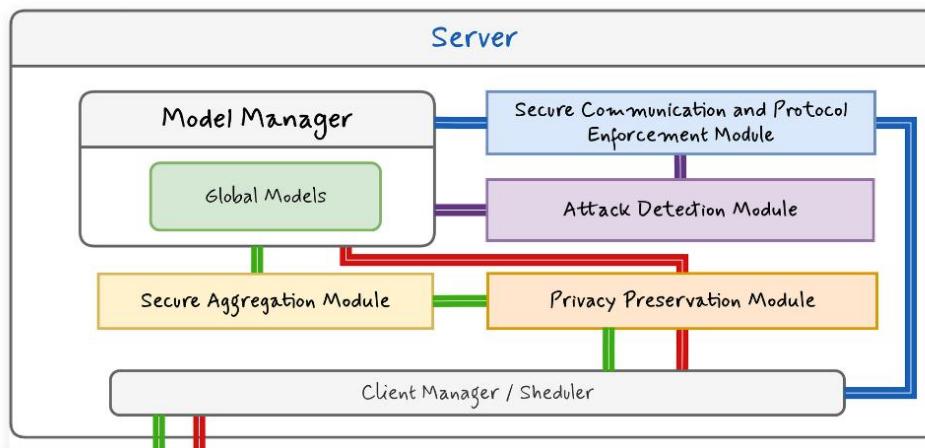


The screenshot shows the FLAME Federated Learning website. At the top, there is a navigation bar with links for Home, About, Scope, Milestones, Tasks, Docs, Team, and a search icon. The main title is "A Privacy-First Federated Learning Framework". Below the title, a subtitle reads: "Building secure, resilient, and efficient ML models without compromising data privacy. Designed for the demands of the Industrial IoT." The acronym "F.L.A.M.E" is displayed above the full name "Federated Learning Aggregated Modular Environment". There are two buttons at the bottom: "Explore Features" and "View on GitHub".

Privacy-Enhanced Federated Learning Framework

Our comprehensive Federated Learning (FL) System Framework is engineered to significantly augment the privacy, security, and operational resilience of machine learning models deployed in decentralized and distributed environments. The framework is composed of four interconnected core modules, collectively guaranteeing data integrity, defense against adversarial attacks, and authenticated inter-component communication.

Framework Overview and Architecture



**Mr. Amila Senerathne**

Supervisor

**Dr. Sanika Wijesekara**

External Supervisor

**Mr. T. Kumaralingam**

Co-Supervisor

