

Snapshot Week 12 of Group PG02

Project of ATSYS

No-Code Solution for InfluxDB

LeStartUP

Zilin Song - a1833935

Jen-Hao Liu - a1893169

Dang Quy Duong - a1893592

Baojing Li - a1894836

Shih-Han Lin - a1900715

Feinan Guo - a1903270

Xiaoqing Zhao - a1904344

Hao Jiang - a1907177

Ziqi Zhang – a1909438

Product Backlog and Task Board

INFLUXUI-ATYSYS / Projects / Project_Board_PG02

Project_Board_PG02

Board Roadmap Sprint 2 Sprint 3 Sprint 4 View 12 + New view

Filter by keyword or by field

Discard Save

No Status 34

InfluxUI-PG02 #24 chore(frontend): update frontend linting, formatting, git commit co...

InfluxUI-PG02 #25 feat(frontend): add login ui

InfluxUI-PG02 #26 feat(frontend): add auth state management with react query & react hook form

InfluxUI-PG02 #28 feat(backend): add basic api function using influxdb client

InfluxUI-PG02 #29 feat(backend): add basic retrieve measurements

InfluxUI-PG02 #30 Fix/13 fix login state persistence

InfluxUI-PG02 #27 Feature/6 api authentication

InfluxUI-PG02 #42 docs(docs): update server readme for api testing

+ Add Item

Backlog 0 / 5

This is your project Backlog

+ Add Item

Planning 0

This is your planning stage

+ Add Item

In Progress 0

This is actively being worked on

+ Add Item

Testing 6

This is your testing phase

InfluxUI-PG02 #6 [Server] Develop API to authenticate with InfluxDB

InfluxUI-PG02 #7 [Server] Develop API to retrieve buckets of InfluxDB

InfluxUI-PG02 #8 [Server] Develop API to retrieve measurements of InfluxDB

InfluxUI-PG02 #9 [Server] Develop API to retrieve fields of InfluxDB

InfluxUI-PG02 #10 [Server] Develop API to execute InfluxDB queries

InfluxUI-PG02 #49 [Server] Implement Current User Endpoint using InfluxDB API

+ Add Item

INFLUXUI-ATYSYS / Projects / Project_Board_PG02

Project_Board_PG02

Board Roadmap Sprint 2 Sprint 3 Sprint 4 View 12 + New view

Filter by keyword or by field

Discard Save

Backlog 0 / 5

This is your project Backlog

+ Add Item

Planning 0

This is your planning stage

+ Add Item

In Progress 0

This is actively being worked on

+ Add Item

Testing 6

This is your testing phase

InfluxUI-PG02 #6 [Server] Develop API to authenticate with InfluxDB

InfluxUI-PG02 #7 [Server] Develop API to retrieve buckets of InfluxDB

InfluxUI-PG02 #8 [Server] Develop API to retrieve measurements of InfluxDB

InfluxUI-PG02 #9 [Server] Develop API to retrieve fields of InfluxDB

InfluxUI-PG02 #10 [Server] Develop API to execute InfluxDB queries

InfluxUI-PG02 #49 [Server] Implement Current User Endpoint using InfluxDB API

+ Add Item

Done 21

This has been completed

InfluxUI-PG02 #4 Project Initialization

InfluxUI-PG02 #12 [Front End] Create login UI component

InfluxUI-PG02 #11 [Infrastructure] Develop container environments

InfluxUI-PG02 #13 [Front End] Implement authentication flow using React Query

InfluxUI-PG02 #15 [Front End] Create components for buckets, measurements, and fields display

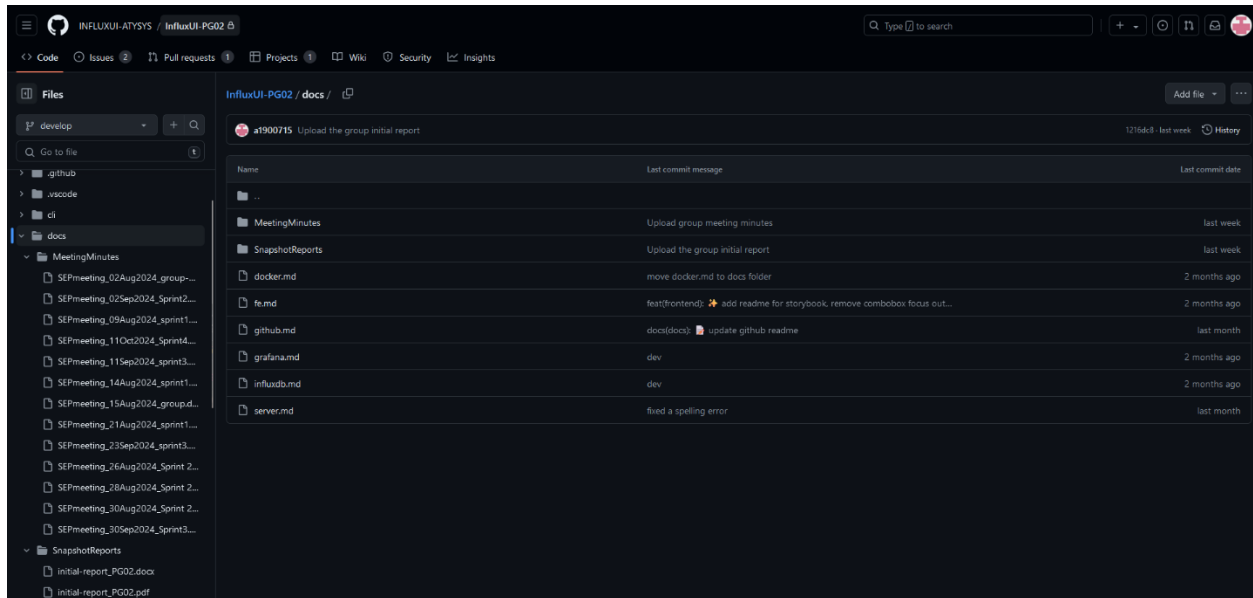
InfluxUI-PG02 #14 [Front End] Design and implement drag-and-drop interface using @xyflow/react

InfluxUI-PG02 #17 [Front End] Implement drag-and-drop between selection list and query builder using @dnd-kit/core

InfluxUI-PG02 #40

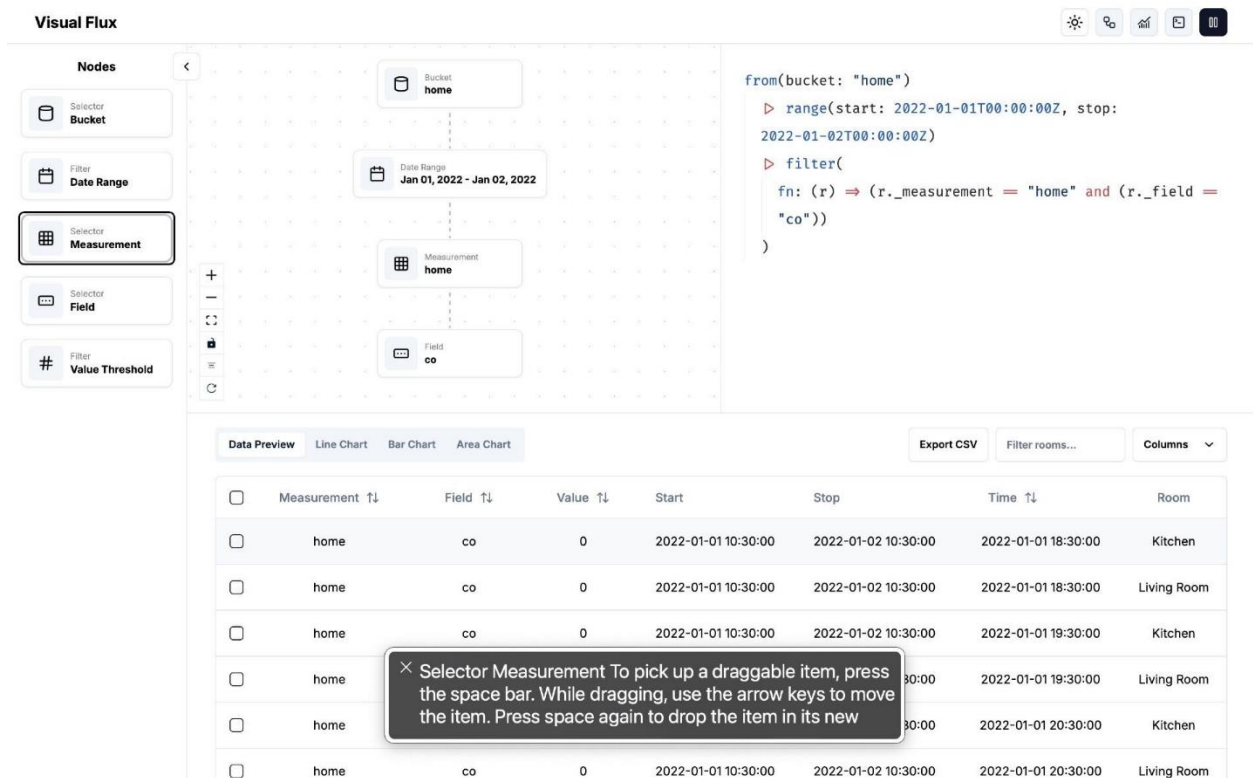
+ Add Item

Documentation for meeting record

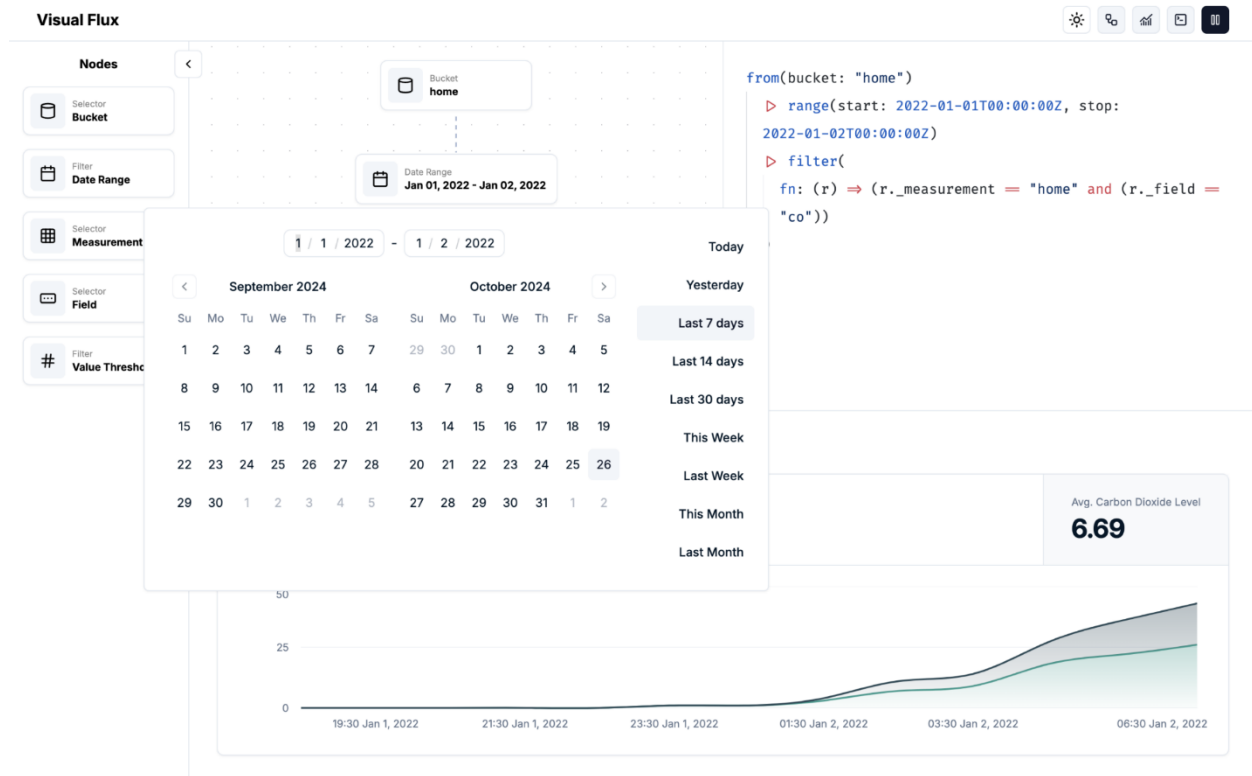


Sprint Backlog and User Stories

- Accessibility mode: screen reader



● UI: data filtering with customisable labels



● Testing: log in

PLAYWRIGHT

> Filter (e.g. text, @tag)
Status: all Projects: chromium

1/1 passed (100%)

- ✓ code-generation.test.ts
 - ✓ Code Generation Test
 - ✓ generates correct Flux query code 20.2s
- ✓ dnd.test.ts
 - ✓ Drag and Drop flow
 - ✓ Drag nodes from sidebar and connect them 20.2s
- ✓ login.test.ts
 - ✓ Authentication Flow
 - ✓ logged in user stays on editor page when going to ... 23.6s
 - ✓ logged in user gets redirected to editor page when ... 22.1s
 - ✓ successful login redirects to editor page 13.9s
 - ✓ failed login shows error message 11.6s
 - ✓ empty form submission shows validation errors
- ✓ visualization.test.ts
 - ✓ Visualization
 - ✓ Visualization should be rendered correctly 27.0s

Actions Metadata

- ✓ Passed 8.5s
- > Before Hooks 1.0s
- page.click locator("button[type='submit']") 7.0s
- expect.toBeVisible getByText("Username is required") 48ms
- expect.toBeVisible getByText("Password is required") 4ms
- > After Hooks 396ms

Action Before After

Locator Source Call Log Errors Console Network Attachments Annotations

login.test.ts

```
99 expect(hasExpectedError).toBe(true);
100
101 await expect(page).toHaveURL("/login");
102 });
103
104 test("empty form submission shows validation errors", async ({ page }) => {
105   await page.click('button[type="submit"]');
106   await expect(page.getByText("Username is required")).toBeVisible();
107   await expect(page.getByText("Password is required")).toBeVisible();
108 });
109
110
111 }
```

● Testing: drag and drop flow

The screenshot shows the Playwright test runner interface. On the left, a list of tests is shown, with 'drag and drop flow' selected. The main panel displays the test results for 'drag and drop flow', showing a list of actions and their durations. The actions include clicking and dragging elements, and a screenshot. The test passed.

Test Results:

- ✓ Passed 20.5s
- locator.click getByRole('button', { name: 'Align' }) 72ms
- locator.dragTo 154ms
- locator.click getByRole('button', { name: 'Align' }) 53ms
- locator.click getByRole('button', { name: 'Pick a t...' 85ms
- locator.fill getByRole('spinbutton') 29ms
- locator.click getByRole('combobox') 370ms
- locator.click getByText('Greater than') 324ms
- locator.dragTo 178ms
- locator.click getByRole('button', { name: 'Align' }) 79ms
- locator.dragTo 138ms
- locator.click getByRole('button', { name: 'Pick a t...' 62ms
- locator.fill getByRole('spinbutton') 26ms
- locator.click getByRole('combobox') 329ms
- locator.click getByText('Smaller than') 327ms
- page.screenshot 108ms
- > After Hooks 2.1s

Test Code (dnd.test.ts):

```
153 await page
154   .getByRole("button", { name: "Pick a threshold" })
155   .first()
156   .click();
157 await page.getByRole("spinbutton").fill("5");
158 await page.getByRole("combobox").click();
159 await page.getByText("Smaller than").click();
160
161 // take a screenshot
162 await page.screenshot({ path: "drag-and-connect-nodes.png" });
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
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

● Testing: visualisation of queries

The screenshot shows the Playwright test runner interface. On the left, a list of tests is shown, with 'visualisation of queries' selected. The main panel displays the test results for 'visualisation of queries', showing a list of actions and their durations. The actions include clicking and waiting for elements to be visible. The test passed.

Test Results:

- ✓ Passed 27.0s
- expect.toBeVisible getByText('Kitchen').first() 118ms
- expect.toBeVisible getByText('Living Room').first() 9ms
- expect.toBeVisible getByText('co').first() 7ms
- locator.click getByText('Line Chart') 118ms
- page.waitForTimeout 3.0s
- expect.toBeVisible locator('.recharts-surface') 7ms
- expect.toBeVisible locator('.recharts-line').first() 6ms
- locator.click getByText('Bar Chart') 81ms
- page.waitForTimeout 3.0s
- expect.toBeVisible locator('.recharts-surface') 6ms
- expect.toBeVisible locator('.recharts-bar').first() 6ms
- locator.click getByText('Area Chart') 74ms
- page.waitForTimeout 3.0s
- expect.toBeVisible locator('.recharts-surface') 7ms
- expect.toBeVisible locator('.recharts-area').first() 7ms
- > After Hooks 1.6s

Test Code (visualization.test.ts):

```
132 await page.getByText("Bar Chart").click();
133 await page.waitForTimeout(3000);
134 await expect(page.locator("css=.recharts-surface")).toBeVisible();
135 await expect(page.locator("css=.recharts-bar").first()).toBeVisible();
136
137 // Switch to Area Chart
138 await page.getByText("Area Chart").click();
139 await page.waitForTimeout(3000);
140 await expect(page.locator("css=.recharts-surface")).toBeVisible();
141 await expect(page.locator("css=.recharts-area").first()).toBeVisible();
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
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

- Testing: code generation

[illegible]

Our team successfully completed multiple-selection of drag-and-drop node in the main user interface with a selection of light and dark mode for user experience. Meanwhile, we also added an additional accessibility mode for visually impaired users.

Definition of Done

- Testing plans are thoroughly followed to ensure the workflow is seamless without critical bugs
- Addition of accessibility function to take different users' experiences into consideration

Summary of Changes

- Additional accessibility mode for visually impaired users
- Possibility of integration of Grafana for more seamless visualisation experience
- Code editing function for more advanced user dynamically interacting with generated code
- Disable selection for data-absent date range to ensure accuracy and performance