

BAB III

PELAKSANAAN KERJA

3.1 Kedudukan dan Koordinasi

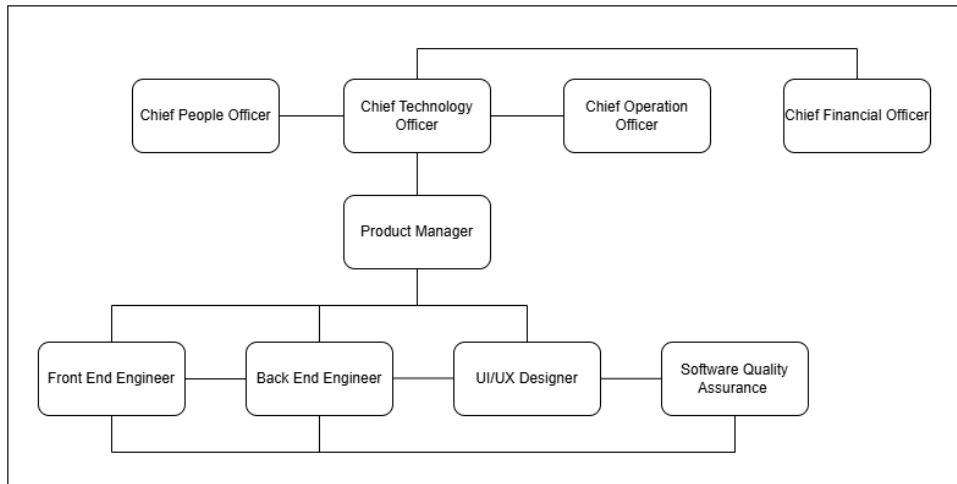
Bagian ini memuat keterangan dan informasi mengenai posisi yang ditempati selama periode praktik kerja serta alur koordinasi yang diterapkan. Dijelaskan pula bagaimana mekanisme koordinasi dilaksanakan dengan pembimbing lapangan pada saat pengerjaan proyek atau penugasan tertentu, termasuk struktur pelaporan dan interaksi tim yang dilakukan.

3.1.1 Kedudukan

Dalam *role* sebagai *QA Intern*, kedudukan yang dimiliki adalah setara dengan rekan-rekan *engineer* dan *designer* lainnya. Hal ini disebabkan tidak adanya senior atau manajer dengan *role* QA yang sama secara langsung dalam tim, sehingga *QA Intern* merupakan satu-satunya yang bertugas. Kondisi ini menuntut tingkat kemandirian yang tinggi dan tanggung jawab penuh terhadap keseluruhan proses pengujian dan kualitas produk. Oleh karena itu, koordinasi dan interaksi dilakukan secara langsung dengan berbagai *role* fungsional lain, seperti *Back-End Engineer*, *Front-End Engineer*, dan *UI/UX Designer*.

3.1.2 Koordinasi

Bagian ini memuat penjelasan terperinci mengenai informasi alur koordinasi pekerjaan yang dilaksanakan selama periode praktik kerja di perusahaan. Dijelaskan pula bagaimana mekanisme alur kerja tersebut divalidasi dan diikuti, yang dapat dilengkapi dengan bagan alur kerja visual untuk memberikan gambaran yang lebih jelas mengenai interaksi dan proses yang dilakukan:



Gambar 3. 1 Alur Koordinasi

Dalam *role* sebagai QA, kedudukan yang dimiliki adalah setara dengan rekan-rekan *engineer* dan *designer* lainnya seperti pada gambar 3.1. Hal ini disebabkan tidak terdapat senior atau manajer dengan *role* QA yang sama secara langsung dalam tim, sehingga QA merupakan satu-satunya yang bertugas. Kondisi ini menuntut tingkat kemandirian yang tinggi dan tanggung jawab penuh terhadap keseluruhan proses pengujian dan kualitas produk.

Struktur dan alur koordinasi pekerjaan dimulai dari tingkat kepemimpinan eksekutif. Kebutuhan spesifik divisi, seperti yang diajukan oleh CPO, COO, dan CFO, dilaporkan kepada CTO. Kebutuhan tersebut kemudian diturunkan kepada *Product Manager* untuk dianalisis dan diterjemahkan menjadi spesifikasi fitur dalam aplikasi. Setelah spesifikasi disepakati, pengembangan dilanjutkan oleh *UI/UX Designer*, *Back-End Engineer*, dan *Front-End Engineer*. *Product Manager* bertanggung jawab untuk memberikan penjelasan komprehensif mengenai keseluruhan konsep produk dan *system flow*. Penjelasan ini kemudian dijadikan dasar utama bagi penyusunan perencanaan pengujian yang akan dilaksanakan.

Apabila terdapat alur (*flow*) yang dinilai cukup rumit, konfirmasi dan pertanyaan dilakukan kepada *UI/UX Designer* untuk meminimalisir salah pemahaman sebelum pengujian dimulai. Selain itu, apabila selama proses pengujian teridentifikasi adanya alur yang janggal, tidak sesuai ekspektasi,

atau membingungkan, diskusi terlebih dahulu dilakukan dengan *Designer*. Tujuan dari langkah ini adalah untuk memverifikasi alur yang seharusnya sesuai dengan spesifikasi desain dan kebutuhan pengguna.

Pelaporan *bug* teknis diarahkan secara langsung kepada tim *Engineer* yang relevan. Jika pengujian menemukan *bug* terkait tampilan (*user interface*) atau fungsionalitas sisi klien, seperti elemen yang tidak berfungsi atau tata letak yang salah, pelaporan ditujukan kepada *Front End Engineer*. Sebaliknya, apabila terdeteksi *bug* pada sisi *server*, seperti *error* kode 500 atau masalah pada integritas data dan logika bisnis, pelaporan diarahkan kepada *Back End Engineer*. Setelah diperoleh konfirmasi alur yang benar dari *Designer* terkait alur yang janggal, pelaporan *bug* atau permintaan perubahan baru akan dilanjutkan kepada *Front End Engineer* untuk implementasi perbaikan.

3.2 Tugas yang Dilakukan

Selama program kerja magang, terdapat beberapa proyek yang dikerjakan, tabel 3.1 merupakan contoh proyek beserta keterangan yang dilakukan dari awal hingga akhir.

Tabel 3. 1 Contoh Proyek Setiap Minggu

| No. | Proyek | Keterangan | Minggu | Tanggal Mulai Aktivitas | Tanggal Akhir Aktivitas |
|-----|------------|--|--------|-------------------------|-------------------------|
| 1 | Perkenalan | Perkenalan Lingkungan Kerja | 1 | 23 Juni 2025 | 27 Juni 2025 |
| 2 | Fibrina | Pemahaman produk Fibrina. | 2 | 30 Juni 2025 | 4 Juli 2025 |
| 3 | Fibrina | Memperbarui dokumentasi <i>flow</i> sistem dan membuat <i>Test Planning</i> dan <i>Skenario</i> modul <i>Request Expense</i> . | 3 | 7 Juli 2025 | 11 Juli 2025 |

| No. | Proyek | Keterangan | Minggu | Tanggal Mulai Aktivitas | Tanggal Akhir Aktivitas |
|-----|---------|---|---------|-------------------------|-------------------------|
| 4 | Fibrina | Membuat <i>Test Planning</i> dan <i>Skenario</i> modul <i>Approval</i> dan <i>Rejection Expense</i> . | 4 | 14 Juli 2025 | 18 Juli 2025 |
| 5 | Fibrina | Membuat <i>Test Planning</i> dan <i>Skenario</i> modul <i>Payment</i> . | 5 - 6 | 21 Juli 2025 | 1 Agustus 2025 |
| 6 | Fibrina | Pelaksanaan <i>Testing</i> manual modul <i>Request</i> dan <i>Approval Expense</i> . | 7 – 8 | 4 Agustus 2025 | 15 Agustus 2025 |
| 7 | Fibrina | Pelaksanaan <i>Testing</i> manual modul <i>Payment</i> dan <i>Automation Testing</i> . | 9 - 10 | 18 Agustus 2025 | 29 Agustus 2025 |
| 8 | Alpha | Pemahaman produk Alpha, membuat <i>Test Planning</i> dan <i>Skenario</i> modul <i>Request</i> dan <i>Approve RO, PO, dan DO</i> . | 11 – 12 | 1 September 2025 | 12 September 2025 |
| 9 | Alpha | Membuat <i>Test Planning</i> dan <i>Skenario</i> modul <i>GR</i> dan <i>Payment</i> . | 13 | 15 September 2025 | 19 September 2025 |
| 10 | Alpha | Pelaksanaan <i>Testing</i> modul <i>Request</i> dan <i>Approve RO, PO, dan DO</i> . | 14 - 15 | 22 September 2025 | 3 Oktober 2025 |
| 11 | Alpha | Pelaksanaan <i>Testing</i> modul <i>GR</i> dan <i>Payment</i> . | 16 - 17 | 6 Oktober 2025 | 22 Oktober 2025 |

3.3 Uraian Pelaksanaan Kerja

Sub-bab ini menyajikan uraian rinci mengenai pelaksanaan kegiatan magang dalam *role QA Intern* selama periode penugasan. Pembahasan mencakup keseluruhan proses, mulai dari pengembangan keterampilan teknis dan non-teknis terkait QA, kolaborasi tim, kontribusi pada pengembangan dan optimalisasi sistem yang diuji, hingga identifikasi kendala yang muncul beserta solusi yang diterapkan. Uraian yang disajikan dilengkapi dengan detail mengenai *role* dan tanggung jawab yang diemban, metode atau alat pengujian yang digunakan, serta hasil konkret yang dicapai pada setiap bagian.

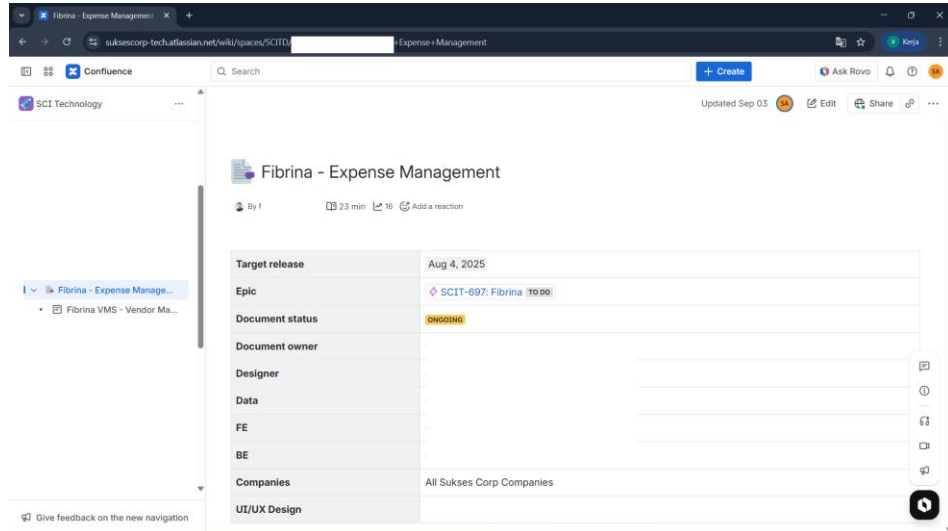
3.3.1 Proses Pelaksanaan

Pada proses ini menjelaskan secara rinci yang telah dilakukan selama kerja magang dari proses awal hingga akhir yang memberikan hasil yang relevan dengan *jobdesc* dan tanggung jawab sebagai *QA Intern* sebagaimana yang dijelaskan pada *point* terinci di bawah ini.

3.3.1.1 Perkenalan Lingkungan Kerja

Hari pertama memasuki program kerja magang dilakukan *onboarding* yang dibawakan oleh HR. Seluruh peserta magang yang tergabung pada hari itu dikumpulkan di salah satu ruang *meeting* yang bernama Ruang Sintia. Pada kesempatan ini, terdapat delapan peserta magang yang terbagi menjadi dalam masing-masing divisi mulai dari divisi *Sourcing*, *HR employee*, *Finance*, *Market Research*, *Legal* dan *QA*. Kegiatan *onboarding* menjadi pintu awal masing-masing peserta magang saling mengenal rekan kerja untuk beberapa bulan kedepan. Sesi ini tidak hanya diisi dengan perkenalan perusahaan, tetapi juga perkenalan masing-masing individu serta *mini games*. Setelah seluruh rangkaian telah dilaksanakan, kemudian peserta magang di ajak *office tour* oleh HR, mulai dari lantai 1 – 7. Setelah itu, masing-masing peserta akan kembali ke meja kerjanya untuk perkenalan lebih lanjut dengan user.

3.3.1.2 Pemahaman produk Fibrina.



Gambar 3. 2 Confluence Fibrina

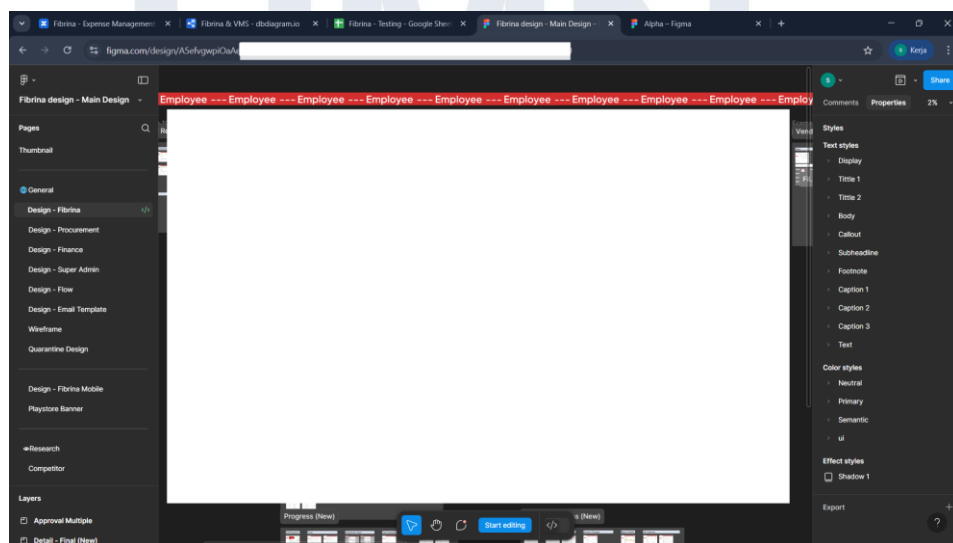
Pada minggu pertama pelaksanaan kegiatan magang, dilakukan upaya pemahaman yang komprehensif terhadap produk Fibrina, khususnya yang berkaitan dengan modul *Expense Management*. Pemahaman ini difokuskan pada analisis fungsionalitas sistem, alur kerja keseluruhan, dan lini masa rilis yang ditargetkan. Proses pemahaman mendalam ini wajib dilakukan sebagai langkah awal dan didasarkan pada tinjauan dokumentasi resmi yang telah tersedia dan tersimpan dalam *tools* Confluence pada gambar 3.2 perusahaan. Dokumentasi yang dipelajari berfungsi sebagai fondasi utama untuk seluruh aktivitas pengujian yang akan dilaksanakan kemudian.

Dokumentasi tersebut memuat *Product Overview* dan *General Requirements*, yang mencakup identifikasi tipe pengguna, aktivitas utama (*key activities*), serta perkiraan total angka pengguna sistem. Alur basis data aplikasi dijelaskan melalui *Entity-Relationship Diagram* (ERD) berdasarkan kebutuhan pengguna, dan alur antarmuka pengguna (*flow UI*) digambarkan melalui diagram

use case serta diagram alir. Selain itu, terdapat penjelasan rinci mengenai *Story Line* yang memetakan fitur dan menu spesifik yang dapat diakses, di mana akses ini disesuaikan berdasarkan perbedaan *role* atau jabatan. Kelengkapan dokumentasi ditunjukkan dengan adanya bagian *Expense Status* yang mendefinisikan arti dari setiap status pengeluaran, serta bagian *Expense Request Fields* yang merinci *fields* wajib pada halaman permintaan pengeluaran, disesuaikan setiap *role* yang ada.

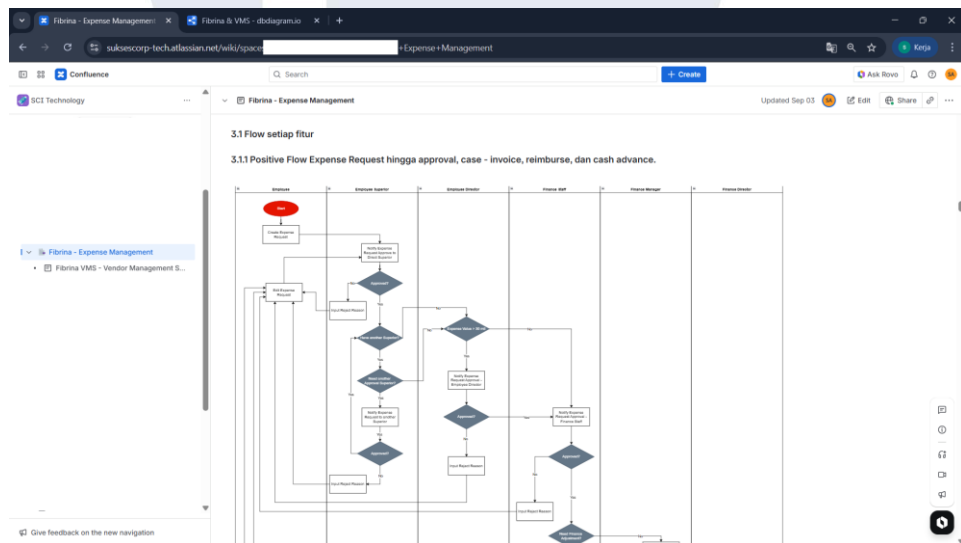
Pemahaman menyeluruh terhadap seluruh dokumentasi teknis dan fungsional yang tersedia ini menjadi landasan yang sangat penting bagi aktivitas *Software Quality Assurance*. Seluruh informasi tersebut digunakan sebagai acuan baku dalam penyusunan perencanaan pengujian, penentuan skenario, dan validasi fungsionalitas sistem Fibrina. Dengan fondasi pemahaman yang kuat, akurasi dan efektivitas pengujian di tahap-tahap selanjutnya dapat terjamin.

3.3.1.3 Memperbarui dokumentasi *flow* sistem dan membuat *Test Planning* dan *Skenario* modul *Request Expense*.



Gambar 3. 3 Figma Fibrina

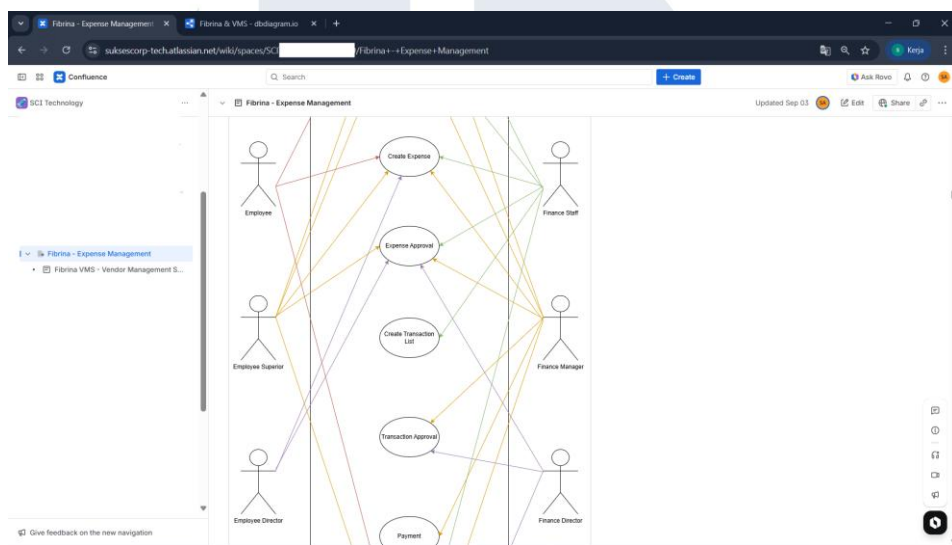
Pada minggu pertama pelaksanaan kegiatan magang, dilakukan upaya pemahaman yang komprehensif terhadap produk Fibrina, khususnya yang berkaitan dengan modul *Expense Management*. Pemahaman ini diawali dengan meninjau *design* antarmuka pengguna (*user interface*) secara terperinci melalui *tools* Figma seperti pada gambar 3.3, yang memuat seluruh rancangan yang telah dibuat oleh *UI/UX Designer*. Selanjutnya, pemahaman juga difokuskan pada analisis fungsionalitas sistem, alur kerja keseluruhan, dan lini masa rilis yang ditargetkan pada tanggal 4 Agustus 2025. Proses pemahaman mendalam ini didasarkan pada tinjauan dokumentasi resmi yang telah tersedia dan tersimpan dalam *tools* Confluence perusahaan.



Gambar 3. 4 Diagram Flow Fibrina

Dokumentasi Confluence tersebut memuat *Product Overview* dan *General Requirements*, yang mencakup identifikasi tipe pengguna, aktivitas utama (*key activities*), serta perkiraan total angka pengguna sistem. Alur basis data aplikasi dijelaskan melalui *Entity-Relationship Diagram* (ERD) gambar 3.4 berdasarkan kebutuhan pengguna, dan alur design interface digambarkan melalui diagram use case serta diagram alur. Selain itu, terdapat penjelasan rinci mengenai

Story Line yang memetakan fitur dan menu spesifik yang dapat diakses, di mana akses ini disesuaikan berdasarkan perbedaan role atau jabatan. Kelengkapan dokumentasi ditunjukkan dengan adanya bagian *Expense Status* yang mendefinisikan arti dari setiap status pengeluaran, serta bagian *Expense Request Fields* yang merinci *fields* wajib pada halaman permintaan pengeluaran, disesuaikan setiap *role* yang ada.



Gambar 3. 5 Use Case Fibrina

Selain pembaruan *flow diagram*, pembuatan *use case* diagram juga dilaksanakan untuk mempermudah pemahaman mengenai fitur dan menu yang dapat diakses oleh setiap pengguna yang terlibat dalam sistem. Dalam aplikasi ini, sistem terdiri dari enam peran utama gambar 3.5, yaitu *Employee*, *Employee Superior*, *Employee Director*, *Finance Staff*, *Finance Manager*, dan *Finance Director*. Setiap aktor yang terlibat memiliki batasan fitur yang spesifik dalam sistem, dan diagram *use case* ini menjadi alat visual yang efektif untuk memetakan batasan akses tersebut. Dengan demikian, dapat dipastikan bahwa setiap peran hanya dapat mengakses fungsionalitas yang sesuai dengan tanggung jawabnya.

Batasan akses fitur yang spesifik diatur untuk setiap peran. Peran *Employee Staff* dibatasi hanya untuk dapat *request expense* dan mengajukan akses (*request access*). Peran *Employee Superior* dan *Employee Director* diberikan wewenang yang sama, yaitu dapat mengajukan *expense*, menyetujui (*approve*) dan menolak (*reject*) *expense*, serta menyetujui dan menolak permintaan akses. Peran *Finance Staff* memiliki wewenang untuk mengajukan *expense*, menyetujui dan menolak *expense*, serta membuat *transaction list*. Sementara itu, *Finance Manager* dan *Finance Director* memiliki wewenang yang serupa, yaitu dapat mengajukan *expense*, menyetujui dan menolak *expense*, serta menyetujui dan menolak *transaction list*.

Alur utama pemrosesan dimulai dari pembuatan *expense* oleh *Employee*, yang kemudian dilanjutkan ke tahap persetujuan oleh seluruh *approver* yang berwenang yaitu *Superior*, *Director*, *Finance Staff*, *Finance Manager*, dan *Finance Director*. Setelah disetujui, selanjutnya akan masuk ke tahap pembayaran dengan *transaction list* yang akan dibuat oleh *Finance Staff*, yang kemudian disetujui oleh *Finance Manager* dan *Finance Director*. Proses dilanjutkan dengan pengunggahan bukti pembayaran (*upload proof of payment*) oleh *Finance Staff* dan diakhiri dengan pengunggahan bukti penggunaan (*upload proof of usage*) oleh *Employee* sebagai *requestor*.

| Finance - Testing | | | | | | | | | | | | | | Share | |
|---|--|--|--|--|--|--|--|--|--|--|--|--|--|-------|--|
| File Edit View Insert Format Data Tools Extensions Help | | | | | | | | | | | | | | | |
| Q Menu 100% 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 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000 1001 1002 1003 1004 1005 1006 1007 1008 1009 1010 1011 1012 1013 1014 1015 1016 1017 1018 1019 1020 1021 1022 1023 1024 1025 1026 1027 1028 1029 1030 1031 1032 1033 1034 1035 1036 1037 1038 1039 1040 1041 1042 1043 1044 1045 1046 1047 1048 1049 1050 1051 1052 1053 1054 1055 1056 1057 1058 1059 1060 1061 1062 1063 1064 1065 1066 1067 1068 1069 1070 1071 1072 1073 1074 1075 1076 1077 1078 1079 1080 1081 1082 1083 1084 1085 1086 1087 1088 1089 1090 1091 1092 1093 1094 1095 1096 1097 1098 1099 1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 1110 1111 1112 1113 1114 1115 1116 1117 1118 1119 1120 1121 1122 1123 1124 1125 1126 1127 1128 1129 1130 1131 1132 1133 1134 1135 1136 1137 1138 1139 1140 1141 1142 1143 1144 1145 1146 1147 1148 1149 1150 1151 1152 1153 1154 1155 1156 1157 1158 1159 1160 1161 1162 1163 1164 1165 1166 1167 1168 1169 1170 1171 1172 1173 1174 1175 1176 1177 1178 1179 1180 1181 1182 1183 1184 1185 1186 1187 1188 1189 1190 1191 1192 1193 1194 1195 1196 1197 1198 1199 1200 1201 1202 1203 1204 1205 1206 1207 1208 1209 1210 1211 1212 1213 1214 1215 1216 1217 1218 1219 1220 1221 1222 1223 1224 1225 1226 1227 1228 1229 1230 1231 1232 1233 1234 1235 1236 1237 1238 1239 1240 1241 1242 1243 1244 1245 1246 1247 1248 1249 1250 1251 1252 1253 1254 1255 1256 1257 1258 1259 1260 1261 1262 1263 1264 1265 1266 1267 1268 1269 1270 1271 1272 1273 1274 1275 1276 1277 1278 1279 1280 1281 1282 1283 1284 1285 1286 1287 1288 1289 1290 1291 1292 1293 1294 1295 1296 1297 1298 1299 1300 1301 1302 1303 1304 1305 1306 1307 1308 1309 1310 1311 1312 1313 1314 1315 1316 1317 1318 1319 1320 1321 1322 1323 1324 1325 1326 1327 1328 1329 1330 1331 1332 1333 1334 1335 1336 1337 1338 1339 1340 1341 1342 1343 1344 1345 1346 1347 1348 1349 1350 1351 1352 1353 1354 1355 1356 1357 1358 1359 1360 1361 1362 1363 1364 1365 1366 1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403 1404 1405 1406 1407 1408 1409 1410 1411 1412 1413 1414 1415 1416 1417 1418 1419 1420 1421 1422 1423 1424 1425 1426 1427 1428 1429 1430 1431 1432 1433 1434 1435 1436 1437 1438 1439 1440 1441 1442 1443 1444 1445 1446 1447 1448 1449 1450 1451 1452 1453 1454 1455 1456 1457 1458 1459 1460 1461 1462 1463 1464 1465 1466 1467 1468 1469 1470 1471 1472 1473 1474 1475 1476 1477 1478 1479 1480 1481 1482 1483 1484 1485 1486 1487 1488 1489 1490 1491 1492 1493 1494 1495 1496 1497 1498 1499 1500 1501 1502 1503 1504 1505 1506 1507 1508 1509 1510 1511 1512 1513 1514 1515 1516 1517 1518 1519 1520 1521 1522 1523 1524 1525 1526 1527 1528 1529 1530 1531 1532 1533 1534 1535 1536 1537 1538 1539 1540 1541 1542 1543 1544 1545 1546 1547 1548 1549 1550 1551 1552 1553 1554 1555 1556 1557 1558 1559 1560 1561 1562 1563 1564 1565 1566 1567 1568 1569 1570 1571 1572 1573 1574 1575 1576 1577 1578 1579 1580 1581 1582 1583 1584 1585 1586 1587 1588 1589 1590 1591 1592 1593 1594 1595 1596 1597 1598 1599 1600 1601 1602 1603 1604 1605 1606 1607 1608 1609 1610 1611 1612 1613 1614 1615 1616 1617 1618 1619 1620 1621 1622 1623 1624 1625 1626 1627 1628 1629 1630 1631 1632 1633 1634 1635 1636 1637 1638 1639 1640 1641 1642 1643 1644 1645 1646 1647 1648 1649 1650 1651 1652 1653 1654 1655 1656 1657 1658 1659 1660 1661 1662 1663 1664 1665 1666 1667 1668 1669 1670 1671 1672 1673 1674 1675 1676 1677 1678 1679 1680 1681 1682 1683 1684 1685 1686 1687 1688 1689 1690 1691 1692 1693 1694 1695 1696 1697 1698 1699 1700 1701 1702 1703 1704 1705 1706 1707 1708 1709 1710 1711 1712 1713 1714 1715 1716 1717 1718 1719 1720 1721 1722 1723 1724 1725 1726 1727 1728 1729 1730 1731 1732 1733 1734 1735 1736 1737 1738 1739 1740 1741 1742 1743 1744 1745 1746 1747 1748 1749 1750 1751 1752 1753 1754 1755 1756 1757 1758 1759 1760 1761 1762 1763 1764 1765 1766 1767 1768 1769 1770 1771 1772 1773 1774 1775 1776 1777 1778 1779 1780 1781 1782 1783 1784 1785 1786 1787 1788 1789 1790 1791 1792 1793 1794 1795 1796 1797 1798 1799 1800 1801 1802 1803 1804 1805 1806 1807 1808 1809 1810 1811 1812 1813 1814 1815 1816 1817 1818 1819 1820 1821 1822 1823 1824 1825 1826 1827 1828 1829 1830 1831 1832 1833 1834 1835 1836 1837 1838 1839 1840 1841 1842 1843 1844 1845 1846 1847 1848 1849 1850 1851 1852 1853 1854 1855 1856 1857 1858 1859 1860 1861 1862 1863 1864 1865 1866 1867 1868 1869 1870 1871 1872 1873 1874 1875 1876 1877 1878 1879 1880 1881 1882 1883 1884 1885 1886 1887 1888 1889 1890 1891 1892 1893 1894 1895 1896 1897 1898 1899 1900 1901 1902 1903 1904 1905 1906 1907 1908 1909 1910 1911 1912 1913 1914 1915 1916 1917 1918 1919 1920 1921 1922 1923 1924 1925 1926 1927 1928 1929 1930 1931 1932 1933 1934 1935 1936 1937 1938 1939 1940 1941 1942 1943 1944 1945 1946 1947 1948 1949 1950 1951 1952 1953 1954 1955 1956 1957 1958 1959 1960 1961 1962 1963 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975 1976 1977 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021 2022 2023 2024 2025 2026 2027 2028 2029 2030 2031 2032 2033 2034 2035 2036 2037 2038 2039 2040 2041 2042 2043 2044 2045 2046 2047 2048 2049 2050 2051 2052 2053 2054 2055 2056 2057 2058 2059 2060 2061 2062 2063 2064 2065 2066 2067 2068 2069 2070 2071 2072 2073 2074 2075 2076 2077 2078 2079 2080 2081 2082 2083 2084 2085 2086 2087 2088 2089 2090 2091 2092 2093 2094 2095 2096 2097 2098 2099 2100 2101 2102 2103 2104 2105 2106 2107 2108 2109 2110 2111 2112 2113 2114 2115 2116 2117 2118 2119 2120 2121 2122 2123 2124 2125 2126 2127 2128 2129 2130 2131 2132 2133 2134 2135 2136 2137 2138 2139 2140 2141 2142 2143 2144 2145 2146 2147 2148 2149 2150 2151 2152 2153 2154 2155 2156 2157 2158 2159 2160 2161 2162 2163 2164 2165 2166 2167 2168 2169 2170 2171 2172 2173 2174 2175 2176 2177 2178 2179 2180 2181 2182 2183 2184 2185 2186 2187 2188 2189 2190 2191 2192 2193 2194 2195 2196 2197 2198 2199 2200 2201 2202 2203 2204 2205 2206 2207 2208 2209 2210 2211 2212 2213 2214 2215 2216 2217 2218 2219 2220 2221 2222 2223 2224 2225 2226 2227 2228 2229 2230 2231 2232 2233 2234 2235 2236 2237 2238 2239 2240 2241 2242 2243 2244 2245 2246 2247 2248 2249 2250 2251 2252 2253 2254 2255 2256 2257 2258 2259 2260 2261 2262 2263 2264 2265 2266 2267 2268 2269 2270 2271 2272 2273 2274 2275 2276 2277 2278 2279 2280 2281 2282 2283 2284 2285 2286 2287 2288 2289 2290 2291 2292 2293 2294 2295 2296 2297 2298 2299 2300 2301 2302 2303 2304 2305 2306 2307 2308 2309 2310 2311 2312 2313 2314 2315 2316 2317 2318 2319 2320 2321 2322 2323 2324 2325 2326 2327 2328 2329 2330 2331 2332 2333 2334 2335 2336 2337 2338 2339 2340 2341 2342 2343 2344 2345 2346 2347 2348 2349 2350 2351 2352 2353 2354 2355 2356 2357 2358 2359 2360 2361 2362 2363 2364 2365 2366 2367 2368 2369 2370 2371 2372 2373 2374 2375 2376 2377 2378 2379 2380 2381 2382 2383 2384 2385 2386 2387 2388 2389 2390 2391 2392 2393 2394 2395 2396 2397 2398 2399 2400 2401 2402 2403 2404 2405 2406 2407 2408 2409 2410 2411 2412 2413 2414 2415 2416 2417 2418 2419 2420 2421 2422 2423 2424 2425 2426 2427 2428 2429 2430 2431 2432 2433 2434 2435 2436 2437 2438 2439 2440 2441 2442 2443 2444 2445 2446 2447 2448 2449 2450 2451 2452 2453 2454 2455 2456 2457 2458 2459 2460 2461 2462 2463 2464 2465 2466 2467 2468 2469 2470 2471 2472 2473 2474 2475 2476 2477 2478 2479 2480 2481 2482 2483 2484 2485 2486 2487 2488 2489 2490 2491 2492 2493 2494 2495 2496 2497 2498 2499 2500 2501 2502 2503 2504 2505 2506 2507 2508 2509 2510 2511 2512 2513 2514 2515 2516 2517 2518 2519 2520 2521 2522 2523 2524 2525 2526 2527 2528 2529 2530 2531 2532 2533 2534 2535 2536 2537 2538 2539 2540 2541 2542 2543 2544 2545 2546 2547 2548 2549 2550 2551 2552 2553 2554 2555 2556 2557 2558 2559 2560 2561 2562 2563 2564 2565 2566 2567 2568 2569 2570 2571 2572 2573 2574 2575 2576 2577 2578 2579 2580 2581 2582 2583 2584 2585 2586 2587 2588 2589 2590 2591 2592 2593 2594 2595 2596 2597 2598 2599 2600 2601 2602 2603 2604 2605 2606 2607 2608 2609 2610 2611 2612 2613 2614 2615 2616 2617 2618 2619 2620 2621 2622 2623 2624 2625 2626 2627 2628 2629 2630 2631 2632 2633 2634 2635 2636 2637 2638 2639 2640 2641 2642 2643 2644 2645 2646 2647 2648 2649 2650 2651 2652 2653 2654 2655 2656 2657 2658 2659 2660 2661 2662 2663 2664 2665 2666 2667 2668 2669 2670 2671 2672 2673 2674 2675 2676 2677 2678 2679 2680 2681 2682 2683 2684 2685 2 | | | | | | | | | | | | | | | |

Gambar 3. 6 Potongan Lembar Kerja Test Planning

Setelah dilakukan pemahaman product, selanjutnya adalah pembuatan lembar kerja untuk *test planning*. Lembar kerja ini digunakan bukan hanya untuk dokumentasi QA dalam melakukan testing seperti gambar 3.6 contoh potongan lembar kerja *test planning*, tapi juga sebagai lembar yang dapat di akses oleh *frontend* dan *backend developer* untuk mengecek apakah seluruh pengujian sudah menyeluruh atau terdapat bagian yang belum di jadikan *planning*.

Dalam penulisan *Test Planning*, sebanyak 14 kolom menjadi persyaratan utama yang harus dipenuhi oleh seorang *Quality Assurance*. Kolom pertama adalah *Naming*, yang berisi *ID* pengujian yang ditulis secara berurutan dengan format *TP – Fibrina – 1* mengacu pada *Test Planning* pertama untuk aplikasi Fibrina. Selanjutnya, terdapat kolom *Category*, yang berfungsi sebagai indikator pengujian dengan dua pilihan yaitu *Functional Test*, yang berfokus pada kegunaan setiap fitur dan alur dan *Regression Test*, yang merupakan pengujian ulang setelah adanya perbaikan temuan atau masalah. Kolom *Automate* digunakan untuk memberikan keterangan mengenai metode pengujian, apakah dibantu dengan alat otomasi atau dilakukan secara manual. Kolom *Role* menjelaskan peran pengguna yang menjalankan skenario, sementara kolom *Apps* memberikan penjelasan mengenai tipe aplikasi yang diuji, apakah berbasis *web* atau *mobile*.

Kolom *Modul* dan *Submodul* berfungsi untuk menjelaskan fungsi umum dan fungsi lebih rinci dari skenario yang dibuat. *Test Skenario* berisi tahapan yang harus dieksekusi dari awal hingga akhir, dan *Duration* mencatat hasil *end point* yang dikeluarkan oleh sistem untuk mengukur kecepatan *backend* dalam memproses data. Kolom *Expected Result* menjelaskan hasil yang seharusnya diberikan pada skenario tersebut, dan *Result* berisi hasil

sesungguhnya dari skenario yang telah dilakukan. Kolom *Status* memberikan keterangan hasil pengujian dengan dua opsi *Pass* apabila pengujian berhasil dan *Not Pass* apabila pengujian gagal. Terakhir, kolom *PC Dev* dan *PC QA* mencatat nama *developer* yang mengerjakan dan *QA* yang melakukan pengujian.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|------------------------|-------------------|-------------------------|---|----------|------------------|----------------------|--------------|---------|------------------------------------|----------|----|----|----|----|----|----|----|----|
| | Test Case ID | Modul | Sub Modul | Log | Log Type | Status | Raise Date | PIC Test | PIC Dev | Attachment | Dev Note | | | | | | | | |
| 5 | TP - Fibrina - 06 - 01 | Superior | Approve Expense Request | Unlink invois | Minor | Done | 02/07/2025 - 3:48 PM | Subrina Azmi | | ■ Bug - TP - Fibrina - 06 - 01.png | | | | | | | | | |
| 6 | TP - Fibrina - 06 - 02 | Superior | Approve Expense Request | Unus Ream-when I use dari or page any notes | Minor | Ready to re-test | 02/07/2025 - 5:00 PM | Subrina Azmi | | ■ Bug - TP - Fibrina - 06 - 02.png | | | | | | | | | |
| 7 | TP - Fibrina - 07 - 01 | Employee Director | Approve Expense Request | Unus Ream-Empli use dari a Director I did not | Minor | Pending | 02/07/2025 - 5:15 PM | Subrina Azmi | | ■ Bug - TP - Fibrina - 07 - 01.png | | | | | | | | | |
| 8 | TP - Fibrina - 10 - 01 | Financa Director | Approve Expense Request | Unus Ream- Finca use dari a Director did not | Minor | Done | 02/07/2025 - 5:55 PM | Subrina Azmi | | ■ Bug - TP - Fibrina - 10 - 01.png | | | | | | | | | |
| 9 | TP - Fibrina - 11 - 01 | Financa Staff | Transaction List | Date fill any Name | Minor | Done | 02/07/2025 - 6:01 PM | Subrina Azmi | | ■ Bug - TP - Fibrina - 11 - 01.png | | | | | | | | | |

Gambar 3. 7 Potongan lembar kerja *Test Log*

Setelah *Test Planning* disusun, tahap selanjutnya adalah penulisan *Test Log* seperti gambar 3.7 contoh lembar kerja *Test Log*. Lembar kerja *Test Log* ini terdiri dari 12 kolom yang berfungsi untuk mendokumentasikan setiap temuan *bug*. Kolom pertama, *Test Case ID*, memiliki penamaan unik setiap barisnya, seperti *TP – Fibrina – 06 – 01*. Format penamaan ini menunjukkan bahwa permasalahan ditemukan dari *Test Planning* aplikasi *Fibrina*, berasal dari urutan skenario nomor 06, dengan dua angka terakhir mengindikasikan jumlah permasalahan yang muncul pada skenario tersebut. Kolom *Modul* dan *Submodul* digunakan untuk mengidentifikasi fungsi yang terpengaruh. Kolom *Log* berisi penjelasan rinci mengenai permasalahan atau *error* yang ditemukan. *Log Type* mengklasifikasikan tipe permasalahan, apakah termasuk dalam kategori minor (*non blocking*) atau permasalahan major (*blocking*). Kolom *Status* memiliki empat kategori yaitu *To Do*, yang berarti temuan telah dilaporkan oleh *QA*, *Pending*, yang mengindikasikan bahwa perbaikan *bug* harus ditunda karena adanya kepentingan lain dari *developer*, *Ready to re-test*, yang berarti *bug* sudah diperbaiki oleh *developer* dan siap untuk diuji kembali oleh *QA*, dan *Done*, yang mengartikan bahwa *bug* sudah selesai diuji kembali. *Raise Date* mencatat tanggal dilaporkannya *bug* oleh *QA*. Kolom *PIC QA*

mencantumkan penanggung jawab yang melaporkan *bug*, sementara *PIC Dev* menunjukkan kepada siapa *bug* harus diperbaiki, apakah *front end* atau *back end*. Terakhir, kolom *Attachment* digunakan untuk menyimpan bukti saat *QA* menemukan *bug* selama pengujian.

| No | Testing | Category | Autotest | Role | App | Module | SubModule | Pre-Request | Test Scenario | Duration (Seconds) | Expected Result | Result |
|----|------------------|-----------------|----------|----------|-----|---------|---|--|---|--------------------|----------------------|--------|
| 3 | TP - Fibrina - 3 | Functional Test | No | Employee | Web | Expense | Create Expense Request Invoice Single | 5. C1 1. C1 2. C2 3. P1 4. C3 5. C4 6. C5 7. C6 | User can create and move to Create Expense Invoice page | 00 ms | TP - Fibrina - 3.png | Pass |
| 4 | TP - Fibrina - 4 | Functional Test | No | Employee | Web | Expense | Create Expense Request Invoice Multiple | 1. C1 2. C2 3. P1 4. C3 5. C4 6. C5 7. C6 | User can create request expense reimburse for single | 01 ms | TP - Fibrina - 4.png | Pass |
| 5 | TP - Fibrina - 5 | Functional Test | No | Employee | Web | Expense | Create Expense Request Reimburse Single | 1. C1 2. C2 3. P1 4. C3 5. C4 6. C5 7. C6 | User can create request expense reimburse for single | | | Pass |

Gambar 3. 8 Potongan Test Planning Fibrina Request Expense

Pada minggu yang sama dengan kegiatan pemahaman produk, penyusunan *Test Planning* dan *Test Skenario* secara rinci mulai dilaksanakan, berfokus pada modul request expense seperti pada gambar 3.8. Modul ini diidentifikasi sebagai alur utama dari aplikasi Fibrina. Perencanaan pengujian dirancang untuk mencakup seluruh skenario pengguna, di mana pengguna diberikan kapabilitas untuk mengajukan pembayaran dengan tiga tipe pengeluaran utama, yaitu tipe *Invoice*, *Reimburse*, dan *Cash Advance*.

Skenario *request expense* dibagi menjadi dua jenis utama, yaitu *positive flow* dan *negative flow*. Alur positif merupakan kondisi ketika skenario pengujian dirancang berjalan maju, dimulai dari menekan tombol *create* hingga pengajuan berhasil disubmit. Sementara itu, alur negatif merupakan kondisi pengujian yang dibuat berjalan maju lalu mundur kembali, misalnya ketika pengajuan dibuat sebagai *draft* alih-alih langsung di-*submit*. Kemudian, dilakukan pengujian pada proses pengeditan *draft* pengeluaran yang telah disimpan sebelumnya.

Kompleksitas pengujian ditingkatkan karena setiap tipe pengeluaran memiliki dua variasi kategori, yaitu *single* dan *multiple*.

Kategori *single* digunakan ketika permintaan pengeluaran hanya untuk satu jenis beban biaya, sedangkan kategori *multiple* digunakan jika pengajuan melibatkan lebih dari satu tipe beban biaya. Oleh karena itu, *test planning* yang dilaksanakan menjadi sangat beragam.

Secara rinci, alur positif diuji untuk enam skenario yang meliputi *create* hingga *submit* pada kombinasi *invoice single* dan *multiple*, *reimburse single* dan *multiple*, serta *cash advance single* dan *multiple*. Selain itu, alur negatif diuji dengan dua variasi utama mulai dari *create* hingga *draft*, dan *edit draft* hingga *submit*. Karena setiap variasi alur negatif diterapkan pada enam kombinasi tipe dan kategori, total terdapat 12 skenario pengujian negatif. Dengan demikian, total keseluruhan *test plan* yang dibuat untuk modul *request expense* ini mencapai 18 skenario.

3.3.1.4 Membuat *Test Planning* dan *Skenario* modul *Approval* dan *Rejection Expense*.

| ID | No | Scenario | Category | Role | Appr | Method | SubMethod | Pre-Requisite | Test Scenario | Duration (Seconds) | Expected Result | Result |
|----|-------------------|-----------------|----------|------------------|------|---------|-----------------------------------|---------------|---------------|--------------------|--|-----------------------|
| 5 | TP - Fibrina - 5 | Test | No | Employee | Web | Expense | (Single) | | | | | |
| 6 | TP - Fibrina - 6 | Functional Test | No | Superior | Web | Expense | Expense Approval Invoice (Single) | | | 77 ms | User can approve expense request single from Employee | TP - Fibrina - 6.png |
| 7 | TP - Fibrina - 7 | Functional Test | No | Director | Web | Expense | Expense Approval Invoice (Single) | | | 64 ms | User can approve expense request single from Superior | TP - Fibrina - 7.png |
| 8 | TP - Fibrina - 8 | Functional Test | No | Finance Staff | Web | Expense | Expense Approval Invoice (Single) | | | 96 ms | User can approve expense request single from Employee Director | TP - Fibrina - 8.png |
| 9 | TP - Fibrina - 9 | Functional Test | No | Finance Manager | Web | Expense | Expense Approval Invoice (Single) | | | 99 ms | User can approve expense request single from Finance Staff | TP - Fibrina - 9.png |
| 10 | TP - Fibrina - 10 | Functional Test | No | Finance Director | Web | Expense | Expense Approval Invoice (Single) | | | 53 ms | User can approve expense request single from Finance Manager | TP - Fibrina - 10.png |

Gambar 3. 9 Potongan Test Planning Approval dan Rejection Fibrina

Pada minggu ini, penyusunan *Test Planning* dan *Test Skenario* difokuskan pada modul persetujuan (*approval flow*) seperti pada gambar 3.9. Proses persetujuan ini melibatkan lima *role* utama, yaitu *Employee Superior*, *Employee Director*, *Finance Staff*, *Finance Manager*, dan *Finance Director*. Skenario pengujian dirancang untuk mencakup alur positif (*approve*) dan alur negatif (*rejection*) dari setiap tipe pengeluaran.

Untuk alur *approve*, pengujian dilakukan pada enam skenario yang meliputi *approve expense* dengan tipe *invoice single* dan *multiple*, *reimburse single* dan *multiple*, serta *cash advance single* dan *multiple*. Sementara itu, alur *rejection* melibatkan pengujian pada enam skenario yang serupa, namun berfokus pada *rejection* pengeluaran. Dengan demikian, setiap *role approver* membutuhkan total 12 skenario pengujian untuk modul *approval*. Karena terdapat lima *role approver* yang berbeda, maka total keseluruhan *test planning* yang dibuat dan didokumentasikan mencapai 60 skenario.

Total keseluruhan test planning yang dibuat untuk pengujian aplikasi Fibrina adalah 135 skenario mulai dari flow positive hingga flow negative. Test planning ini mencakup dari role *employee*, *employee superior*, *employee director*, *finance staff*, *finance manager* dan *finance director*. Dengan total tiga modul utama modul *expense*, *approve expense* dan *payment*.

3.3.1.5 Membuat Test Planning dan Skenario modul *Payment*.

| No | Naming | Category | Automate | Role | Apps | Modul | SubModul | Pre-Request | Test Scenario |
|----|-------------------|-----------------|----------|------------------|------|---------|--|-------------|---------------|
| 29 | TP - Fibrina - 29 | Functional Test | No | Finance Director | Web | Expense | SubModule Single | | |
| 30 | TP - Fibrina - 30 | Functional Test | No | Finance Staff | Web | Payment | Create Transaction List for Multiple Expense Request Reimburse | | |
| 31 | TP - Fibrina - 31 | Functional Test | No | Finance Manager | Web | Payment | Approve Transaction List | | |
| 32 | TP - Fibrina - 32 | Functional Test | No | Finance Director | Web | Payment | Approve Transaction List | | |
| 33 | TP - Fibrina - 33 | Functional Test | No | Finance Manager | Web | Payment | Payment Transaction List | | |
| 34 | TP - Fibrina - 34 | Functional Test | No | Employee | Web | Expense | Need to Complete | | |
| 35 | TP - Fibrina - 35 | Functional Test | No | Employee | Web | Expense | Create Expense Request Reimburse Multiple | | |

Gambar 3. 10 Potongan *Test Planning Payment* Fibrina

Pada modul pembayaran (*Payment*), terdapat dua jenis alur utama yang diuji, yaitu alur positif dan alur negatif. Alur positif

didefinisikan sebagai kondisi ketika seluruh pengeluaran yang terdaftar dalam satu daftar transaksi (*transaction list*) seperti gambar 3.10 disetujui oleh kedua *approver* transaksi, yaitu *Finance Manager* dan *Finance Director*. Sebaliknya, alur negatif akan terjadi jika terjadi penolakan (*rejection*) oleh salah satu dari kedua *approver* tersebut.

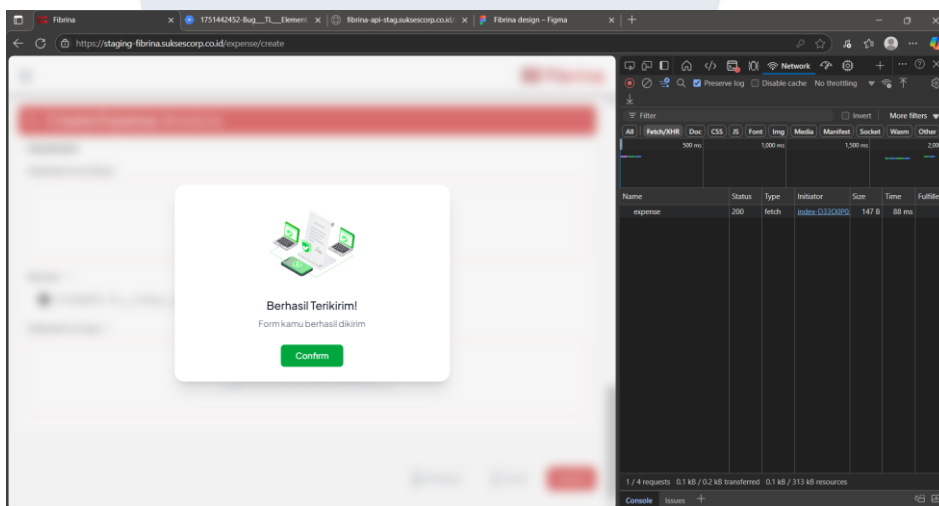
Penyusunan *Test Planning* untuk modul ini terbagi menjadi beberapa tahapan pengujian yang melibatkan berbagai *role*. Tahap awal, yaitu pembuatan daftar transaksi, dilakukan oleh *Finance Staff* untuk pengajuan pembayaran. Tahap *create* ini mencakup sembilan skenario pengujian, enam skenario untuk pembuatan daftar transaksi tunggal (*single transaction list*) yang memvalidasi *expense* bertipe *invoice single*, *invoice multiple*, *reimburse single*, *reimburse multiple*, *cash advance single*, dan *cash advance multiple*. Tiga skenario tambahan dibuat untuk skema *bulk transaction list* yang menguji penggabungan *expense single* dan *multiple* untuk tipe *invoice*, *reimburse*, dan *cash advance*. Tahap pembuatan ini menghasilkan subtotal sembilan *test planning*.

Setelah daftar transaksi dibuat, proses akan dilanjutkan ke tahap persetujuan oleh *Finance Manager*. Terdapat sembilan *test planning* yang serupa yang dirancang untuk tahap persetujuan ini. *Finance Manager* bertanggung jawab untuk menyetujui setiap daftar transaksi yang masuk, baik yang berisi satu *expense* (tunggal) maupun yang berisi banyak *expense* (*bulk*), sehingga menghasilkan subtotal sembilan *test planning* untuk persetujuan *Finance Manager*. Selanjutnya, daftar transaksi akan masuk ke tahap persetujuan *Finance Director*, yang juga memiliki subtotal sembilan *test planning* yang sama dengan *Finance Manager*.

Setelah disetujui, daftar transaksi akan kembali kepada *Finance Staff* untuk aktivitas unggah bukti pembayaran (*upload*

proof of payment). Bukti pembayaran wajib diinput per setiap *expense* secara individual, tidak dapat dilakukan secara *bulk*. Pengujian untuk tahap ini mencakup seluruh kombinasi dari transaksi tunggal dan *bulk*, sehingga total terdapat 24 *test planning*. Terakhir, *expense* akan kembali kepada *Requestor* untuk aktivitas unggah bukti penggunaan (*upload proof of usage*), dengan total 12 *test planning* yang telah disusun sehingga keseluruhan *test planning* yang dibuat untuk modul transaksi (*transaction module*) adalah 63 skenario .

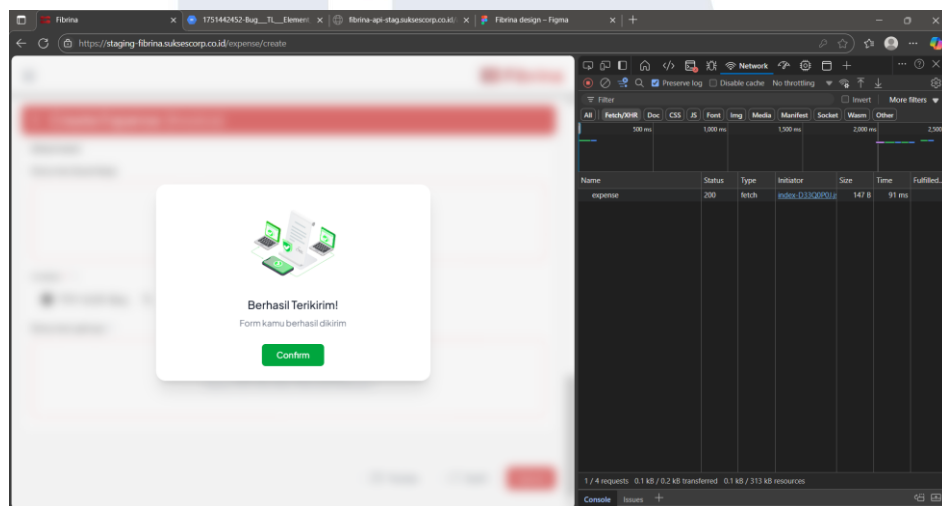
3.3.1.6 Pelaksanaan *Testing* manual modul *Request* dan *Approval Expense*.



Gambar 3. 11 Contoh Hasil *Testing Request Expense Invoice Single*

Pengujian fungsional modul *Request* dan *Approval Expense* dilakukan secara manual seperti pada gambar 3.11 untuk memvalidasi alur pembuatan (*create*) hingga pengiriman data. Salah satu hasil pengujian ditampilkan pada proses pembuatan *expense* dengan tipe *invoice single*. Hasil pengujian tersebut menunjukkan bahwa ketika tombol *submit* di antarmuka diklik, *end-point* yang dituju merespons dengan Status Code 200 (OK). Kode status ini mengindikasikan bahwa permintaan *request* yang dikirimkan oleh

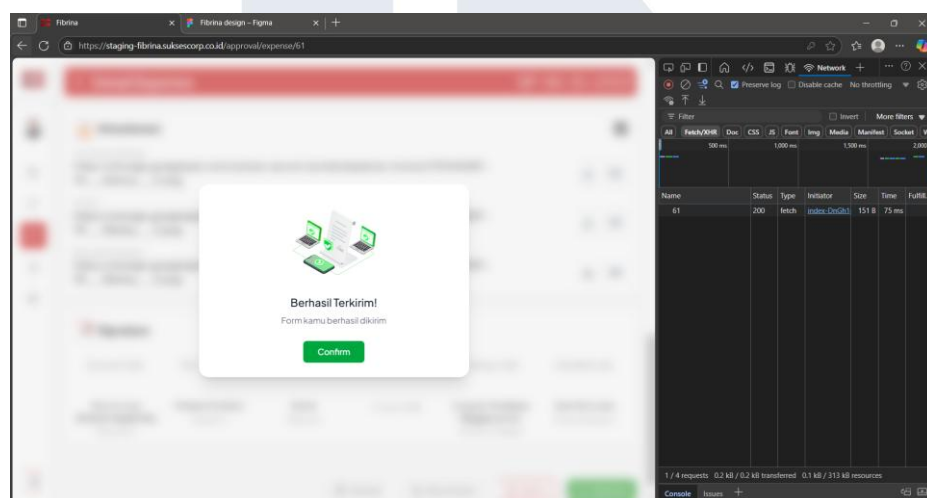
sistem (berisi data *expense* dalam format JSON) berhasil diterima dan diproses oleh server tanpa adanya kesalahan (*error*). Waktu respons yang tercatat adalah 88 milidetik (ms), yang menunjukkan bahwa pengiriman data dan respons dari server terjadi dalam durasi yang sangat cepat. Durasi respons yang singkat ini mengindikasikan kinerja *end-point* yang optimal saat memproses data *expense* tunggal. Dengan demikian, data JSON *expense* telah berhasil dikirim dan tersimpan di basis data, dan notifikasi "*Berhasil Terkirim!*" ditampilkan sebagai konfirmasi kepada pengguna.



Gambar 3. 12 Contoh Hasil *Testing Request Expense Invoice Multiple*

Pengujian fungsional modul *Request* dan *Approval Expense* dilanjutkan dengan skenario pembuatan *expense* berjenis Invoice Multiple seperti pada gambar 3.12. Pengujian ini bertujuan untuk memvalidasi kemampuan sistem dalam memproses pengiriman data lebih dari satu *expense* dalam satu kali proses *submit*. Hasil pengujian yang diamati menunjukkan bahwa ketika data *multiple expense* dikirimkan dari antarmuka, *end-point* yang dituju merespons dengan Status Code 200 (OK). Kode status ini menegaskan bahwa permintaan *request* yang membawa *payload* data *multiple expense* berhasil diterima dan diproses oleh server tanpa adanya hambatan teknis. Waktu respons yang tercatat adalah

91 milidetik (ms), yang mengindikasikan bahwa kinerja *end-point* tetap berada dalam kondisi yang sangat baik meskipun memproses data *expense* dalam jumlah jamak. Kecepatan respons yang singkat ini sangat penting untuk mendukung pengalaman pengguna. Setelah proses selesai, notifikasi "*Berhasil Terkirim!*" ditampilkan sebagai konfirmasi bahwa data *multiple expense* telah berhasil disimpan ke basis data.

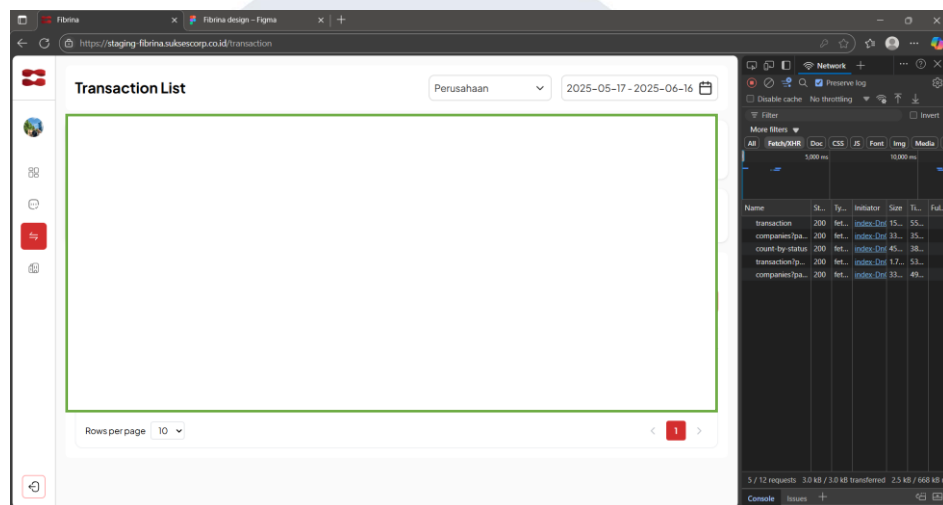


Gambar 3. 13 Contoh Hasil *Testing Approval Expense Invoice Multiple*

Pengujian fungsional modul *Request* dan *Approval Expense* dilanjutkan dengan skenario persetujuan (*approval*) *expense* yang berjenis Invoice Multiple seperti pada gambar 3.13. Pengujian ini memvalidasi kemampuan *approver* untuk menyetujui sekaligus beberapa *expense* yang terkait dalam satu permintaan. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani persetujuan (*approval*) merespons dengan Status Code 200 (OK). Kode status tersebut mengindikasikan bahwa permintaan *request* persetujuan *multiple expense* berhasil diterima dan diproses oleh server tanpa adanya kesalahan teknis. Waktu respons yang tercatat adalah 75 milidetik (ms), menunjukkan bahwa kinerja *end-point* sangat baik dan cepat dalam memproses persetujuan untuk data *expense* jamak. Kecepatan respons yang efisien ini penting untuk

memastikan kelancaran alur kerja persetujuan. Setelah proses selesai, notifikasi "*Berhasil Terkirim!*" ditampilkan sebagai konfirmasi bahwa status *multiple expense* telah berhasil diperbarui menjadi *approved* di basis data

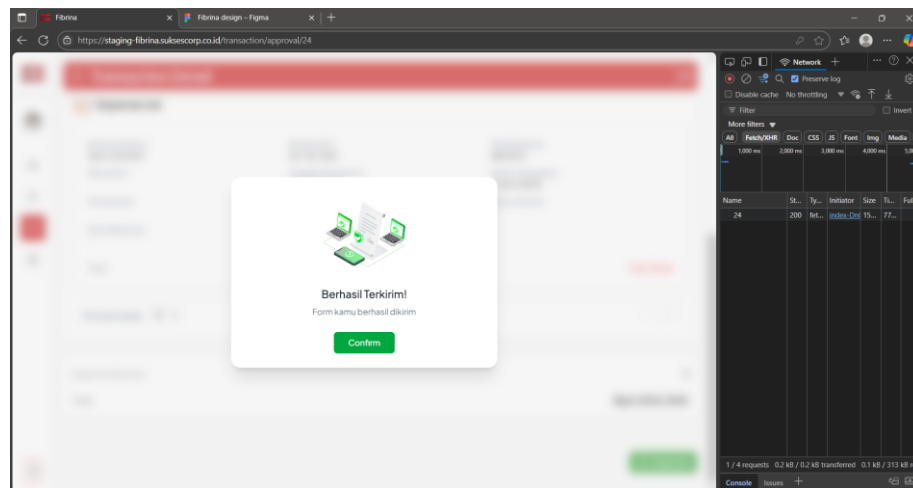
3.3.1.7 Pelaksanaan *Testing* manual modul *Transaction List* dan *Paid Expense* dan *Automation Testing*.



Gambar 3. 14 Contoh Hasil *Testing Create Transaction List*

Pengujian fungsional modul *Transaction List* dilakukan dengan skenario pembuatan daftar transaksi oleh *Finance Staff* seperti pada gambar 3.14. Modul ini bertujuan untuk mengelompokkan *expense* yang sudah *approved* menjadi satu daftar yang siap untuk proses pembayaran lebih lanjut. Hasil pengujian menunjukkan bahwa ketika perintah *Create Transaction List* dijalankan, *end-point* yang menangani pembuatan daftar tersebut merespons dengan Status Code 200 (OK). Kode status ini memverifikasi bahwa *request* untuk membuat *transaction list* telah berhasil diproses oleh server tanpa kegagalan. Waktu respons yang tercatat adalah 53 milidetik (ms), mengindikasikan bahwa kinerja *end-point* sangat cepat dalam memproses *request* pembuatan data. Kecepatan ini memastikan efisiensi dalam alur kerja Finance. Setelah *transaction list* berhasil dibuat, data yang baru seperti pada

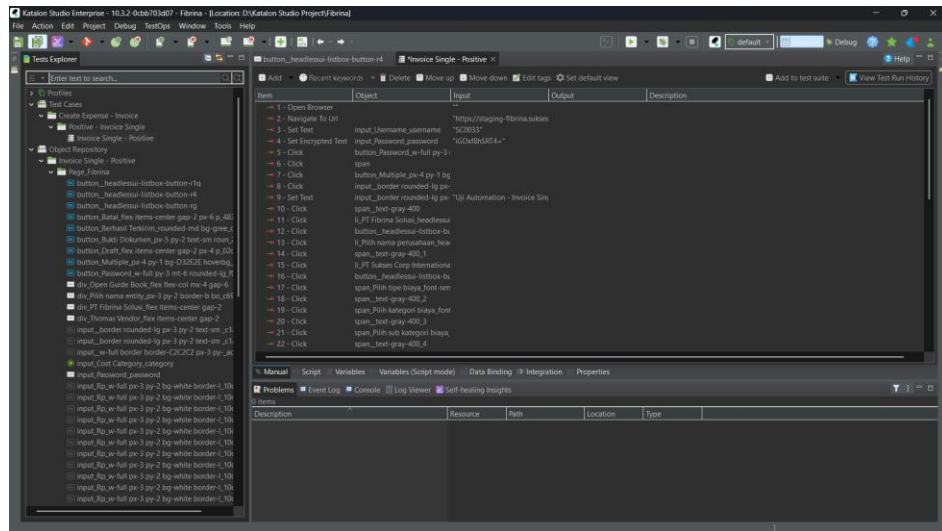
gambar 3.12 contohnya *Transaction ID* 24 pada gambar akan diperbarui pada *dashboard* dengan status awal *Need Finance Manager Approval*, menandakan bahwa alur persetujuan telah dimulai.



Gambar 3. 15 Contoh Hasil *Testing Approve Transaction List*

Pengujian fungsional modul *Transaction List* dilanjutkan dengan skenario persetujuan (*approval*) seperti pada gambar 3.15 daftar transaksi oleh *approver* yang berwenang merujuk pada *Finance Manager* atau *Finance Director* dalam alur. Pengujian ini memvalidasi kemampuan sistem untuk mengubah status *transaction list* dari *pending* menjadi *approved* melalui antarmuka persetujuan. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani aksi persetujuan merespons dengan Status Code 200 (OK). Kode status ini memverifikasi bahwa *request* persetujuan yang dikirimkan oleh sistem telah berhasil diproses oleh server tanpa adanya kesalahan teknis. Waktu respons yang tercatat adalah 77 milidetik (ms), mengindikasikan bahwa kinerja *end-point* efisien dan responsif dalam memproses pembaruan status data. Kecepatan respons yang singkat ini memastikan kelancaran dan efektivitas alur persetujuan. Setelah proses selesai, notifikasi "*Berhasil Terkirim!*" ditampilkan, mengonfirmasi bahwa status

transaction list telah berhasil diperbarui dan dialihkan ke tahap proses berikutnya di basis data.



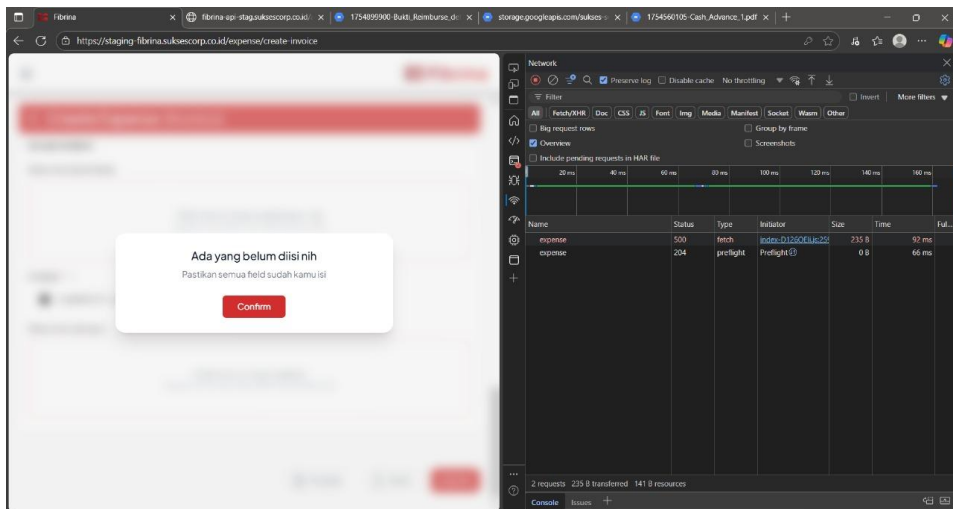
Gambar 3. 16 Hasil Testing Automation Katalon Studio

Pelaksanaan *testing automation* dilakukan menggunakan Katalon Studio Enterprise seperti pada gambar 3.16. Metode otomasi ini dipilih untuk mempermudah proses eksekusi *test case* yang bersifat pasti dan berulang. Pembuatan *test case* otomasi diawali dengan pemanfaatan fitur Web Recorder; fitur ini merekam setiap interaksi pengguna pada *browser*, yang kemudian secara otomatis menghasilkan skrip pengujian. Pada skenario ini, *testing automation* digunakan secara spesifik untuk percobaan *create expense* karena elemen antarmuka (*button* atau *element*) yang terlibat tidak banyak mengalami perubahan posisi. Katalon Studio bekerja dengan menangkap setiap item pada aplikasi sebagai objek yang unik, di mana sistem akan mendeteksi *input* yang dimasukkan oleh pengguna. Dengan demikian, serangkaian langkah *test case* yang kompleks dapat dieksekusi secara otomatis dan akurat, mengurangi waktu yang dibutuhkan untuk pengujian manual berulang.

| Test Case ID | Modul | Sub Modul | Log | Log Type | Status | Raise Date | PIC Test | PIC Dev | Attachment | Dev Note |
|------------------------|-------------------|--|---|----------|--------|-----------------------|--------------|---------|------------|----------------------------------|
| TP - Fibrina - 03 - 01 | Employee | Create Expense - Invoice Single | Test box for Description expense cant fill with long text | Minor | Pass | 03/07/2025 - 10:30 AM | Sabrina Azmi | | | |
| TP - Fibrina - 03 - 02 | Employee | Create Expense - Invoice Single | Test box for Description of Edit Expense Detail cant fill with long text | Minor | Pass | 03/07/2025 - 10:35 AM | Sabrina Azmi | | | |
| TL - Element - 1 | Employee | Create Expense - Invoice Single | The input box in the Category form is required, but no asterisk (*) or required indicator is shown for section Type, Kategori, Sub Kategori and Name Staff | Minor | Pass | 03/07/2025 - 2:40 PM | Sabrina Azmi | | | |
| TP - Fibrina - 06 - 01 | Superior | Approve Expense Request | Unable to approve expense request for multiple invoices because the approval page does not display | Minor | Pass | 03/07/2025 - 3:45 PM | Sabrina Azmi | | | Bug - TL - Element - 1.png |
| TP - Fibrina - 06 - 02 | Superior | Approve Expense Request | Unexpected section "Unexpected expense detail finance" appears on the Detail Superior page when the Finance staff did not provide any notes | Minor | Pass | 03/07/2025 - 5:00 PM | Sabrina Azmi | | | Bug - TP - Fibrina - 06 - 02.png |
| TP - Fibrina - 07 - 01 | Employee Director | Approve Expense Request | Unexpected section "Unexpected expense detail finance" appears on the Detail Expense Director Employee page when the Finance staff did not provide any notes | Minor | Pass | 03/07/2025 - 5:15 PM | Sabrina Azmi | | | Bug - TP - Fibrina - 07 - 01.png |
| TP - Fibrina - 10 - 01 | Finance Director | Approve Expense Request | Unexpected section "Unexpected expense detail finance" appears on the Detail Expense Director Finance page when the Finance staff did not provide any notes (duplicate) | Minor | Pass | 03/07/2025 - 5:50 PM | Sabrina Azmi | | | |
| TP - Fibrina - 11 - 01 | Finance Staff | Transaction List | Date filter is still visible when the Company Name dropdown is open | Minor | Pass | 03/07/2025 - 5:50 PM | Sabrina Azmi | | | Bug - TP - Fibrina - 11 - 01.png |
| TP - Fibrina - 11 - 02 | Finance Staff | Transaction List | Transaction list cant be created even though all forms have been approve | Minor | Pass | 03/07/2025 - 9:30 AM | Sabrina Azmi | | | Bug - TP - Fibrina - 11 - 02.png |
| TP - Fibrina - 04 - 01 | Employee | Create Expense - Invoice Multiple | Alert "Bulan diatas tahun" appears even though the expense can still be submitted | Minor | Pass | 03/07/2025 - 4:30 PM | Sabrina Azmi | | | Bug - TP - Fibrina - 04 - 01.png |
| TP - Fibrina - 04 - 02 | Employee | Create Expense - Invoice Multiple | Unable to create expense even when all form fields are filled | Minor | Pass | 03/07/2025 - 4:40 PM | Sabrina Azmi | | | Bug - TP - Fibrina - 04 - 02.png |
| TP - Fibrina - 16 - 01 | Employee Director | Approve Expense Request - Invoice Multiple | In the attachment form, attachments from Dokumen Surat Kerja and Dokumen Screenshot are displayed even though the requestor detached it from them | Minor | Pass | 04/07/2025 - 11:35 AM | Sabrina Azmi | | | Bug - TP - Fibrina - 16 - 01.png |
| TP - Fibrina - 17 - 01 | Employee Staff | Transaction List | User cant upload proof of usage because the page doesnt display | Minor | Pass | 04/07/2025 - 1:35 PM | Sabrina Azmi | | | Bug - TP - Fibrina - 17 - 01.png |
| TP - Fibrina - 04 - 03 | Employee Staff | Create Expense - Invoice Multiple | Test box for Description of Edit Expense Detail cant fill with long text and error at Expense Detail after change Description to short text | Minor | Pass | 04/07/2025 - 2:45 PM | Sabrina Azmi | | | Bug - TP - Fibrina - 04 - 03.png |
| TP - Fibrina - 06 - 03 | Superior | Approve Expense Request - Invoice Multiple | Description Expense is missing on the approval page, although it was provided by the requestor | Minor | Pass | 04/07/2024 - 3:35 PM | Sabrina Azmi | | | Bug - TP - Fibrina - 06 - 03.png |
| TP - Fibrina - 06 - 04 | Superior | Approve Expense Request - Invoice Multiple | The system shows Dokumen Surat Kerja and Dokumen Screenshot attachments even when the requestor left them empty | Minor | Pass | 04/07/2024 - 3:35 PM | Sabrina Azmi | | | Bug - TP - Fibrina - 06 - 04.png |

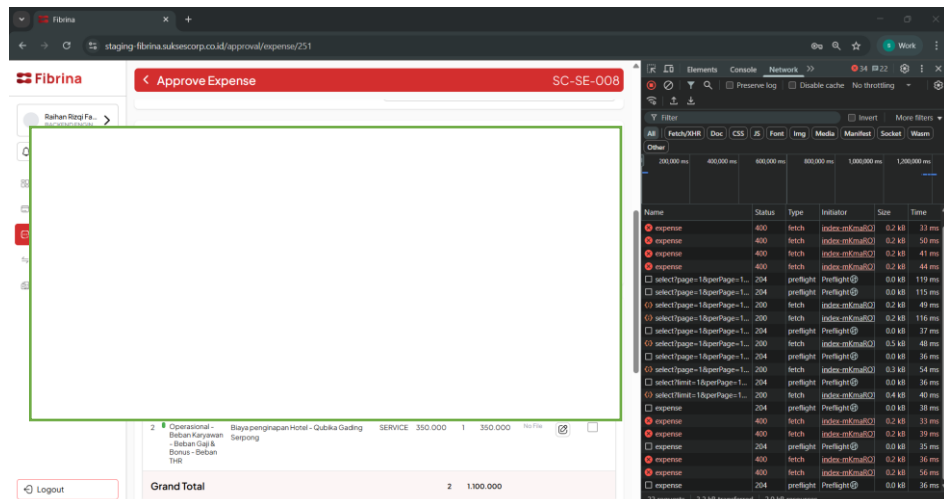
Gambar 3. 17 Hasil *Test Log* Fibrina

Dalam pengujian yang telah dilakukan mulai dari modul *request* hingga *payment* yang melibatkan seluruh role. Terdapat banyak temuan *error* atau *bug* yang muncul seperti pada gambar 3.17. Hal ini ditemukan bukan hanya dari pengujian berdasarkan interface tetapi juga pengujian data. Pengujian *interface* dilihat dengan acuan dari design yang telah dibuat oleh *UI/UX Designer* pada Figma seperti Gambar 3.2, menguji apakah setiap komponen sudah sesuai, tidak ada komponen yang hilang atau delay dalam penampilan data. Sedangkan pengujian data mengacu pada pelemparan data yang di input dan hasil yang diberikan, apakah data yang dimasukan dan di tampilkan sesuai, pengujian ini dilihat dari hasil response pada *end-point* yang diberikan. Pada proyek Fibrina terdapat 79 bug yang ditemukan, mulai dari *bug minor* hingga *major*.



Gambar 3. 18 Contoh Bug Major

Pada gambar 3.18 merupakan contoh bug major yang ditemukan ketika ingin menguji *create draft request expense* menyebabkan *blocking* sehingga pengujian tidak dapat dilanjutkan ke tahap berikutnya. Hal ini dapat dilihat dari status yang dikeluarkan oleh *end-point* berupa 500 dengan permasalahan pada *backend* yang belum membuat development untuk *case draft*.



Gambar 3. 19 Contoh Bug Minor

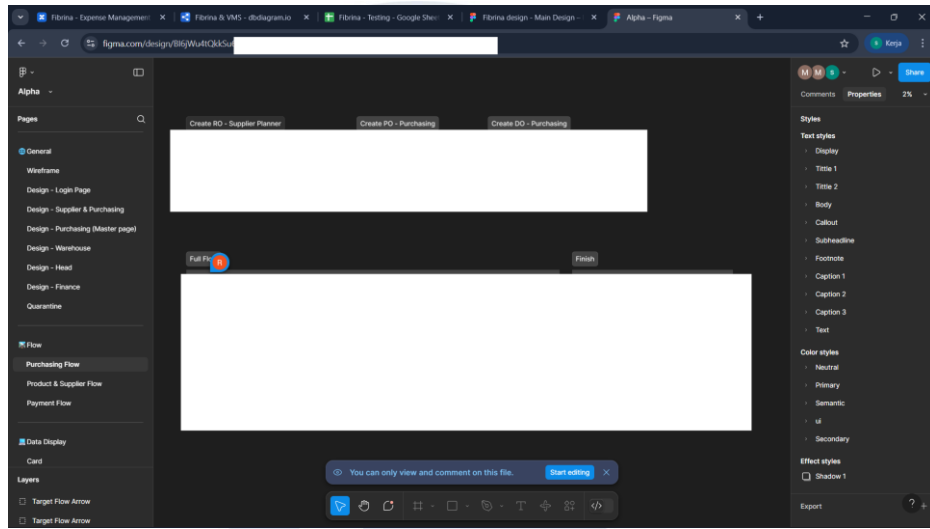
Selain *bug major* yang ditemukan sebelumnya, terdapat pula temuan bug minor, seperti gambar 3.19. *Bug* jenis ini biasanya muncul tanpa mengakibatkan *blocking* pada alur kerja utama pengguna.

3.3.1.8 Pemahaman produk Alpha, membuat *Test Planning* dan *Skenario* modul *Request* dan *Approve* RO, PO, dan DO.



55

pembuatan dokumen awal hingga proses persetujuan dan pembayaran yang memiliki kompleksitas bersyarat. Berdasarkan pemahaman tersebut, *Test Planning* dan *Test Skenario* mulai disusun untuk memvalidasi seluruh skenario bisnis dan batasan *role* di setiap modul.



Gambar 3. 21 Figma Alpha

Selain peninjauan dokumentasi di Confluence, penjelasan lebih rinci terkait aplikasi juga dipelajari melalui *tools* Figma, seperti yang ditunjukkan pada Gambar 3.21. Pemahaman desain ini sangat penting karena di dalamnya dimuat seluruh detail visual *user interface*. Hal ini meliputi perancangan tata letak halaman, peletakan tombol (*button*), penggunaan komponen, serta konsistensi visual di seluruh modul. Pemahaman desain di Figma ini berfungsi sebagai acuan utama bagi pengujian *UI* dan *UX*, memastikan bahwa implementasi yang dilakukan oleh tim *Engineer* sesuai dengan spesifikasi visual yang telah disepakati oleh *UI/UX Designer*.

| No | Naming | Category | Automate | Role | Apps | Modul | Submodul | Total Scenario | Duration (Seconds) | Expected Result | Result | Status | PC Dev |
|----|----------------|-----------------|----------|----------|------|---------------|----------------------------------|--|--------------------|--|-------------------------|--------|-----------|
| 4 | TP - Alpha - 4 | Functional Test | No | Supplier | Web | Request Order | Create Request Order | 1. User can create request order by 10 all data 2. User can create request order by 10 all data 3. User can create request order by 10 all data 4. User can create request order by 10 all data 5. User can create request order by 10 all data 6. User can create request order by 10 all data 7. User can create request order by 10 all data 8. User can create request order by 10 all data | 17 ms | User can create request order by 10 all data | TP - Alpha - 04 - 8.jpg | PASS | Pass Test |
| 5 | TP - Alpha - 5 | Functional Test | No | Supplier | Web | Request Order | Draft Request Order | 1. User can draft request order by 10 all data 2. User can draft request order by 10 all data 3. User can draft request order by 10 all data 4. User can draft request order by 10 all data 5. User can draft request order by 10 all data 6. User can draft request order by 10 all data 7. User can draft request order by 10 all data 8. User can draft request order by 10 all data | 22 ms | User can draft request order by 10 all data | TP - Alpha - 05.jpg | PASS | Pass Test |
| 6 | TP - Alpha - 6 | Functional Test | No | Supplier | Web | Request Order | Edit Request Order | 1. User can edit request order by 10 all data 2. User can edit request order by 10 all data 3. User can edit request order by 10 all data 4. User can edit request order by 10 all data 5. User can edit request order by 10 all data 6. User can edit request order by 10 all data 7. User can edit request order by 10 all data 8. User can edit request order by 10 all data | 79 ms | User can edit request order by 10 all data | TP - Alpha - 06 (1).jpg | PASS | Pass Test |
| 7 | TP - Alpha - 7 | Functional Test | No | Supplier | Web | Request Order | Cancel Request Order | 1. User can cancel request order by 10 all data 2. User can cancel request order by 10 all data 3. User can cancel request order by 10 all data 4. User can cancel request order by 10 all data 5. User can cancel request order by 10 all data 6. User can cancel request order by 10 all data 7. User can cancel request order by 10 all data 8. User can cancel request order by 10 all data | 86 ms | User can cancel request order by 10 all data | TP - Alpha - 07.jpg | PASS | Pass Test |
| 8 | TP - Alpha - 8 | Functional Test | No | Supplier | Web | Request Order | Delete Request Order - Cancelled | 1. User can delete request order by 10 all data 2. User can delete request order by 10 all data 3. User can delete request order by 10 all data 4. User can delete request order by 10 all data 5. User can delete request order by 10 all data 6. User can delete request order by 10 all data 7. User can delete request order by 10 all data 8. User can delete request order by 10 all data | 1.00 ms | User can delete request order by 10 all data | TP - Alpha - 08.jpg | FAIL | Fail Test |

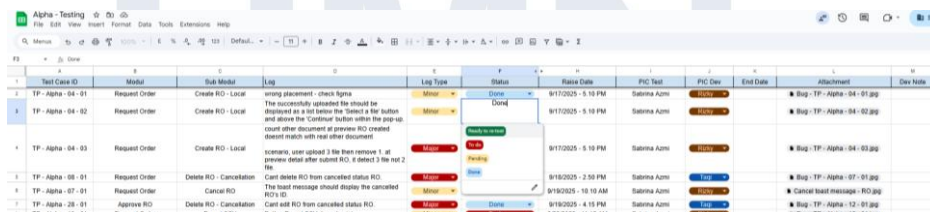
Gambar 3. 22 Potongan Lembar Kerja *Test Planning*.

Setelah dilakukan pemahaman produk, selanjutnya dilaksanakan pembuatan lembar kerja untuk *Test Planning* seperti pada gambar 3.22. Lembar kerja ini digunakan sebagai dokumentasi strategis bagi *QAdalam* melaksanakan *testing*, serta berfungsi sebagai sumber informasi yang dapat diakses oleh *front end* dan *back end developer* untuk memverifikasi kelengkapan cakupan pengujian.

Dalam penulisan *Test Planning*, sebanyak 14 kolom menjadi persyaratan utama yang harus dipenuhi. Kolom pertama adalah *Naming*, yang berisi *ID* pengujian yang ditulis secara berurutan dengan format *TP – Alpha – 4* mengacu pada *Test Planning* untuk aplikasi Alpha dan angka terakhir menunjukkan bahwa skenario tersebut ada dalam urutan beberapa. Selanjutnya, terdapat kolom *Category*, yang berfungsi sebagai indikator pengujian dengan dua pilihan yaitu *Functional Test*, yang berfokus pada kegunaan setiap fitur dan alur dan *Regression Test*, yang merupakan pengujian ulang setelah adanya perbaikan temuan atau masalah. Kolom *Automate* digunakan untuk memberikan keterangan mengenai metode pengujian, apakah dibantu dengan alat otomatisasi atau dilakukan secara manual. Kolom *Role* menjelaskan peran pengguna yang menjalankan skenario, sementara kolom *Apps* memberikan

penjelasan mengenai tipe aplikasi yang diuji, apakah berbasis *web* atau *mobile*.

Kolom *Modul* dan *Submodul* berfungsi untuk menjelaskan fungsi umum dan fungsi lebih rinci dari skenario yang dibuat. *Test Skenario* berisi tahapan yang harus dieksekusi dari awal hingga akhir, dan *Duration* mencatat hasil *end point* yang dikeluarkan oleh sistem untuk mengukur kecepatan *back end* dalam memproses data. Kolom *Expected Result* menjelaskan hasil yang seharusnya diberikan pada skenario tersebut, dan *Result* berisi hasil sesungguhnya dari skenario yang telah dilakukan. Kolom Status memberikan keterangan hasil pengujian yang terdiri dari empat opsi: *Pass* memiliki arti bahwa pengujian berhasil, *Not Pass* memiliki arti bahwa pengujian gagal, *Skip-Remove* memiliki arti bahwa pengujian dilewati karena skenario dihapus atau tidak relevan, dan *Skip-Double* memiliki arti bahwa pengujian dilewati karena merupakan skenario duplikasi. Terakhir, kolom *PC Dev* dan *PC QA* mencatat nama *developer* yang mengerjakan dan *QA* yang melakukan pengujian.



| Test Case ID | Modul | Sub Modul | Log | Log Type | Status | Start Date | End Date | PC Dev | PC QA | Attachment | Dev Note |
|----------------|---------------|--------------------------|--|----------|---------|----------------------|----------|--------------|-------|---------------------------------|----------|
| TP-Alpha-04-01 | Request Order | Create RO - Local | wrong placement - check signs | Minor | Done | 9/10/2025 - 5:10 PM | | Sabrina Azmi | | Bug - TP-Alpha-04-01.jpg | |
| TP-Alpha-04-02 | Request Order | Create RO - Local | The successfully uploaded file should be displayed as a file below the 'Send' button and above the 'Continue' button within the pop-up. Could other document at previous RO created doesn't match with real other document | Minor | Done | 9/10/2025 - 5:10 PM | | Sabrina Azmi | | Bug - TP-Alpha-04-02.jpg | |
| TP-Alpha-04-03 | Request Order | Create RO - Local | scenario, user upload 3 file then remove 1, at previous detail after upload RO, it select 3 file not 2 file | Major | Pending | 9/10/2025 - 5:10 PM | | Sabrina Azmi | | Bug - TP-Alpha-04-03.jpg | |
| TP-Alpha-08-01 | Request Order | Delete RO - Cancellation | Can't delete RO from cancelled status RO | Minor | Done | 9/10/2025 - 2:50 PM | | Sabrina Azmi | | Bug - TP-Alpha-08-01.jpg | |
| TP-Alpha-08-01 | Request Order | Cancel RO | The load message should display the cancelled RO ID | Minor | Done | 9/10/2025 - 10:10 AM | | Sabrina Azmi | | Cancel load message - RO ID.jpg | |
| TP-Alpha-08-01 | Approve RO | Delete RO - Cancellation | Can't edit RO from cancelled status RO | Minor | Done | 9/10/2025 - 4:10 PM | | Sabrina Azmi | | Bug - TP-Alpha-08-01.jpg | |

Gambar 3. 23 Potongan Lembar Kerja *Test Log*

Seluruh temuan *bug* dan *error* didokumentasikan dalam lembar kerja *Test Log*. Lembar kerja ini memiliki 12 kolom yang berfungsi untuk memonitor siklus hidup setiap *bug* seperti pada gambar 3.23. Kolom *Test Case ID* memiliki penamaan unik yang mengacu pada *ID* dari *Test Planning*. Sebagai contoh, *TP – Alpha – 08 – 01* mengindikasikan bahwa itu adalah *bug* pertama yang ditemukan saat pelaksanaan skenario kedelapan dari *Test Plan*

Alpha. Selanjutnya, kolom *Modul* dan *Submodul* digunakan untuk mengidentifikasi fungsionalitas yang terpengaruh. Kolom *Log* berisi deskripsi terperinci mengenai *error* yang ditemukan, termasuk penjelasan tentang kondisi *bug* saat ini dan bagaimana hasil yang seharusnya ditampilkan. Kolom *Log Type* mengklasifikasikan tingkat keparahan permasalahan, apakah termasuk kategori *Minor* atau *Major*.

Kolom Status pada *Test Log* memiliki empat kategori yang menunjukkan progres perbaikan. Status *To Do* mengindikasikan bahwa *bug* telah dilaporkan oleh *QA* dan sedang menunggu perbaikan oleh *developer*. Status *Pending* berarti perbaikan *bug* akan ditunda oleh *developer* karena adanya prioritas atau kepentingan lain. Status *Ready to re-test* mengartikan bahwa *bug* sudah selesai diperbaiki oleh *developer* dan siap untuk diuji kembali oleh *QA*. Terakhir, status *Done* menunjukkan bahwa *QA* telah selesai memverifikasi perbaikan *bug*. Kolom *Raise Date* mencatat tanggal *bug* ditemukan dan dilaporkan oleh *QA*. *PIC Test* mencantumkan penanggung jawab yang melakukan pengujian, dan *PIC Dev* mengarahkan *bug* tersebut kepada tim yang bertanggung jawab untuk perbaikan, apakah *front end* atau *back end*. Kolom Attachment digunakan untuk melampirkan bukti visual saat *QA* menemukan *bug* selama pengujian, dan kolom Dev Note berfungsi untuk mencatat catatan atau komentar dari *developer* terkait proses perbaikan.

| No | Name | Category | Automate | Role | Apps | Modul | Submodul | Test Scenarios | Duration (Second) | Expected Result |
|----|-----------------|-----------------|----------|----------|------|---------------|----------------------------------|--|-------------------|---|
| 3 | TP - Alpha - 3 | Functional Test | No | Supplier | Web | Request Order | Create Request Order - Import | 1. User can create request order by import data | 117 ms | User can create request order by import data |
| 4 | TP - Alpha - 4 | Functional Test | No | Supplier | Web | Request Order | Create Request Order | 1. User can create request order by manual | 117 ms | User can create request order by manual |
| 5 | TP - Alpha - 5 | Functional Test | No | Supplier | Web | Request Order | Draft Request Order | 1. User can draft request order | 133 ms | User can draft request order |
| 6 | TP - Alpha - 6 | Functional Test | No | Supplier | Web | Request Order | Edit Request Order | 1. User can edit RO from draft data | 779 ms | User can edit RO from draft data |
| 7 | TP - Alpha - 7 | Functional Test | No | Supplier | Web | Request Order | Cancel Request Order | 1. User can cancel RO when no one created | 96 ms | User can cancel RO when no one created |
| 8 | TP - Alpha - 8 | Functional Test | No | Supplier | Web | Request Order | Delete Request Order - Cancelled | 1. User can delete Request Order if it submitted before with status 'Cancel' | | User can delete Request Order if it submitted before with status 'Cancel' |
| 9 | TP - Alpha - 9 | Functional Test | No | Supplier | Web | Request Order | Delete Request Order - Draft | 1. User can delete Request Order from draft | 87 ms | User can delete Request Order from draft |
| 10 | TP - Alpha - 10 | Functional Test | No | Supplier | Web | List Overview | Export CSV | 1. User can export data to CSV file | | User can export data to CSV file |

Gambar 3. 24 Potongan Test Planning RO, PO, dan DO

Penyusunan *Test Planning* dan *Test Skenario* seperti pada gambar 3.24 untuk modul RO, PO, dan DO dilaksanakan setelah pemahaman terhadap *product requirement* selesai dilakukan. Skenario pengujian dirancang untuk mencakup *positive case* dan *negative case* guna memvalidasi seluruh fungsionalitas dan integrasi logis antarmodul. Secara keseluruhan, 29 *Test Planning* telah disusun untuk ketiga modul ini.

Modul RO diuji melalui total 12 skenario. Alur pembuatan dan manajemen RO diuji melalui delapan skenario, yang terdiri dari *create local* dan *import*, *cancel RO local* dan *import*, *draft RO*, dan *edit RO*. Alur negatif pembatalan RO diatur dengan syarat ketat yaitu apabila RO telah dibuatkan PO oleh purchasing staff, PO terkait wajib dibatalkan terlebih dahulu, memastikan tidak ada PO yang berjalan saat RO dibatalkan. Selain itu, empat *Test Plan* disiapkan untuk tahap persetujuan oleh *Procurement Head*, yang meliputi pengujian *approve* dan *reject* untuk RO *local* dan *import*.

Pengujian untuk PO juga disusun menjadi 12 skenario. Sebanyak delapan *Test Plan* dialokasikan untuk pembuatan dan manajemen PO, yang terdiri dari *create local* dan *import*, *cancel PO local* dan *import*, *draft PO*, dan *edit PO*. Pembatalan PO dapat

dilakukan dengan prosedur berjenjang: PO hanya dapat dibatalkan apabila belum dibuatkan DO, dan apabila DO telah ada, maka DO tersebut harus dibatalkan terlebih dahulu. Empat *Test Plan* disiapkan untuk tahap persetujuan oleh *Purchasing Head*, meliputi *approve* dan *reject* untuk PO *local* dan *import*.

Terakhir, modul DO diuji menggunakan lima *Test Plan* yang difokuskan pada variasi tipe pembuatan DO. Lima skenario *positive flow* telah disusun berdasarkan jenis pengiriman, yaitu tiga skenario *local* dengan pemilihan menggunakan ekspedisi, *pick-up*, dan *supplier driver* dan dua skenario *import* dengan pemilihan menggunakan *sea* dan *air*. Mekanisme pembatalan juga diuji secara berjenjang. Pembatalan DO hanya dapat dilaksanakan apabila tidak terdapat GR yang melekat. Apabila GR telah dibuat, pembatalan GR wajib dilakukan terlebih dahulu sebelum pembatalan DO dapat diproses. Seluruh skenario ini dirancang untuk memvalidasi batasan *role* dan integrasi logis antarmodul agar sistem dapat beroperasi sesuai dengan kebutuhan bisnis.

3.3.1.9 Membuat Test Planning dan Skenario modul GR dan Payment.

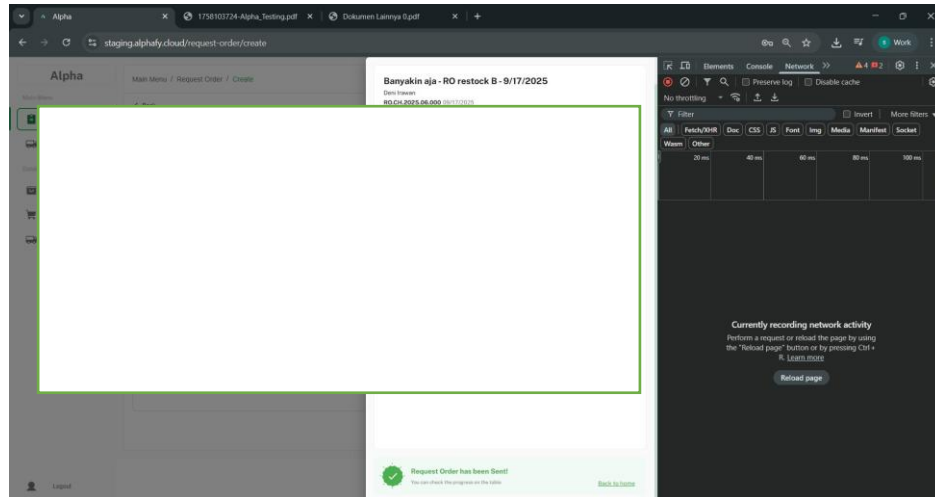
| No | Testing | Category | Automate | Role | Agent | Modul | Submodul | Test Scenario | Duration (Second) | Expected Result |
|----|-----------------|-----------------|----------|------------|-------|---------|---|--|-------------------|---|
| 55 | TP - Alpha - 55 | Functional Test | No | Purchasing | Web | Payment | Create Payment - Purchase Order - IC | 1. User can create Payment with Purchase Order Voucher with Hard Copy type | 45 ms | User can create Payment with Purchase Order Voucher with Hard Copy type |
| 56 | TP - Alpha - 56 | Functional Test | No | Purchasing | Web | Payment | Create Payment - Purchase Order - IC | 1. User can create Payment with Purchase Order Voucher with Soft Copy type | 54 ms | User can create Payment with Purchase Order Voucher with Soft Copy type |
| 57 | TP - Alpha - 57 | Functional Test | No | Purchasing | Web | Payment | Create Payment - Purchase Order - Local | 1. User can create Payment for Purchase Order Local type | | User can create Payment for Purchase Order Local type |
| 58 | TP - Alpha - 58 | Functional Test | No | Purchasing | Web | Payment | Create Payment - Good Receipt - IC | 1. User can create payment with Good Receipt and Hard copy type | 55 ms | User can create payment with Good Receipt and Hard copy type |
| 59 | TP - Alpha - 59 | Functional Test | No | Purchasing | Web | Payment | Create Payment - Custom - IC | 1. User can create Payment with Custom Hard copy type | 38 ms | User can create Payment with Custom Hard copy type |

Gambar 3. 25 Potongan *Test Planning Payment*

Modul *Payment* seperti pada gambar 3.25 diuji melalui alur berjenjang yang melibatkan beberapa *role*. Proses *positive case* dimulai dari *Purchasing Staff* yang membuat *payment voucher*, dilanjutkan dengan persetujuan oleh *Purchasing Head*. Setelah disetujui, *Purchasing Staff* membuat *handover payment* yang kemudian disetujui oleh *Finance Staff*. *Payment voucher* tersebut lalu masuk ke *Finance Manager* untuk dibuatkan *transaction list*, dan *transaction list* ini disetujui oleh *Finance Director*. Proses diakhiri dengan *Finance Staff* yang mengunggah *proof of payment* dari *transaction list* tersebut, yang kemudian diterima oleh *Purchasing Staff*. Jika terdapat *voucher type* yang menggunakan dokumen *hard copy*, *Purchasing Staff* wajib membuat *additional handover* yang disetujui oleh *Finance Staff*.

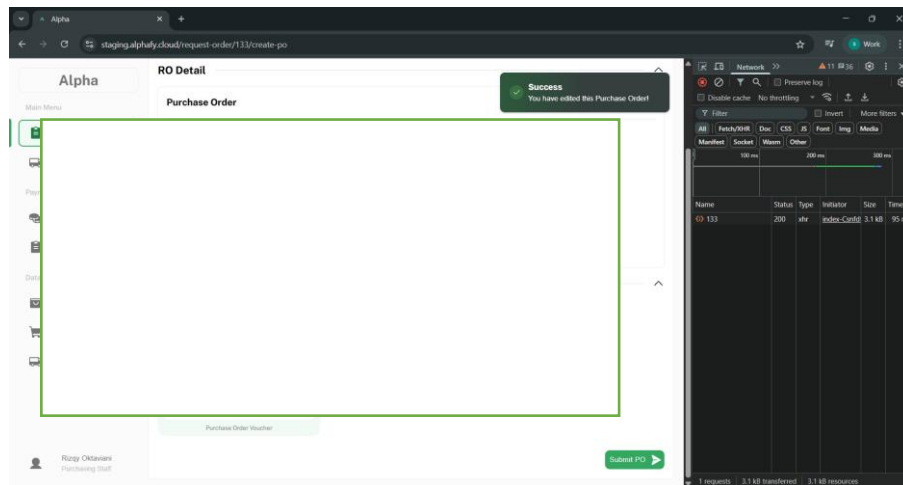
Secara keseluruhan, 24 *Test Planning* telah disusun untuk seluruh alur modul *Payment*. Tahap pembuatan *payment voucher* diuji melalui empat *test planning* untuk kondisi *PO local* dan *import*. Tahap persetujuan oleh *Purchasing Head* diuji dengan empat skenario *approve* dan empat skenario *reject*. Selanjutnya, satu *test planning* disiapkan untuk pembuatan *handover*, satu untuk *approve handover* oleh *Finance Staff*, dan satu untuk *reject handover* oleh *Finance Staff*. Alur *transaction list* diuji dengan satu skenario *create* oleh *Finance Manager*, satu skenario *approve* oleh *Finance Director*, dan satu skenario *reject* oleh *Finance Director*. Empat *test case* dibuat untuk proses *upload proof of payment* oleh *Finance Staff*. Terakhir, proses *additional handover* diuji dengan satu skenario *submit* oleh *Purchasing Staff*, satu skenario *approve*, dan satu skenario *reject* oleh *Finance Staff*.

3.3.1.10 Pelaksanaan Testing modul *Request* dan *Approve* RO, PO, dan DO.



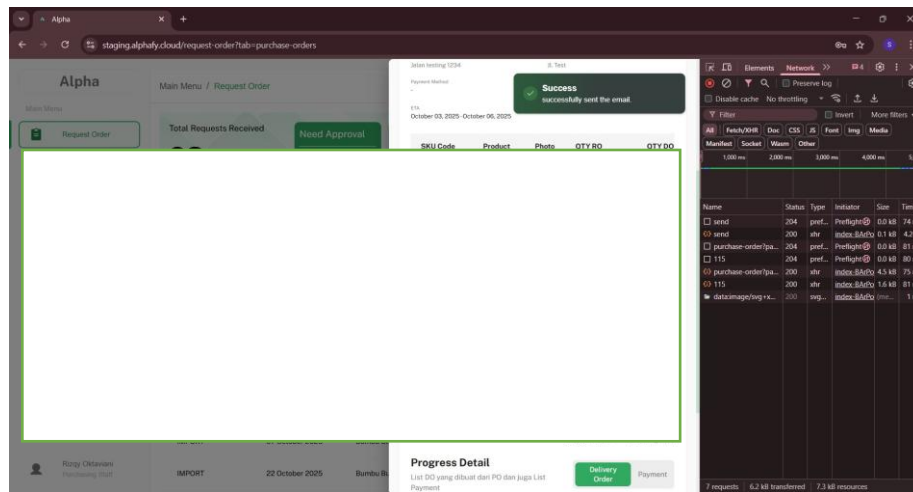
Gambar 3. 27 Contoh Hasil *Testing* RO

Pengujian fungsional modul RO dimulai dengan skenario pembuatan RO oleh *Supplier Planner* seperti pada gambar 3.27. Pengujian ini dilakukan untuk memvalidasi alur pengiriman data RO yang mencakup detail produk (*SKU Code*), jumlah pesanan, dan dokumen pendukung yang diunggah. Hasil pengujian yang diamati menunjukkan bahwa ketika RO telah diajukan, *end-point* yang menangani pembuatan RO merespons dengan kode status yang ideal. Status ini mengindikasikan bahwa permintaan *request* yang berisi seluruh detail RO berhasil diterima dan diproses oleh server tanpa adanya kegagalan teknis. Notifikasi “*Request Order has been sent!*” ditampilkan sebagai konfirmasi bahwa data RO telah berhasil tersimpan dan dialihkan ke tahap proses berikutnya. Validasi ini memastikan bahwa *Supplier Planner* dapat mengajukan permintaan order secara efisien sesuai alur yang ditetapkan.



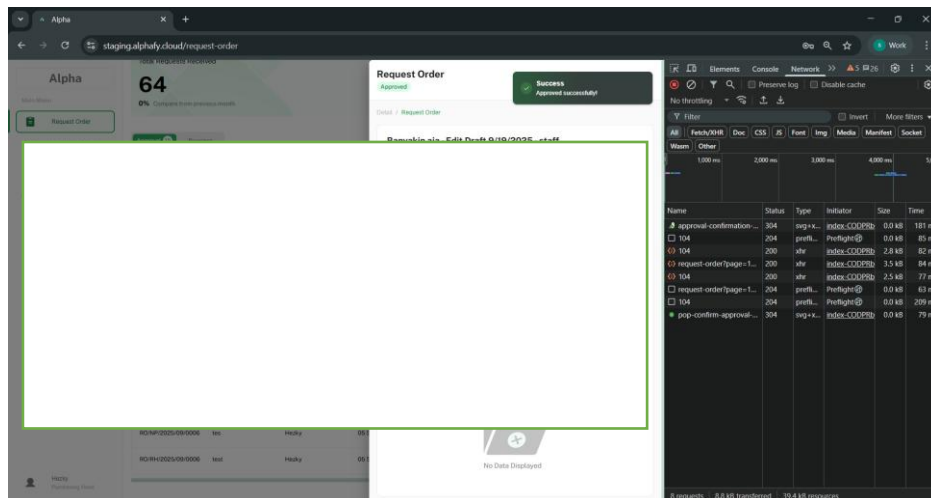
Gambar 3. 28 Contoh Hasil *Testing* PO

Pengujian fungsional modul PO dilanjutkan dengan skenario pembuatan PO oleh *Purchasing Staff*. Pembuatan PO dilakukan berdasarkan RO yang telah disetujui, seperti yang terlihat dari detail RO *New Order 2 (Import)* yang ditampilkan. Pengujian ini bertujuan untuk memvalidasi alur pembuatan PO seperti pada gambar 3.28, termasuk pemrosesan data *Supplier*, nominal transaksi, dan status PO baru. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani pembuatan PO merespons dengan Status Code 200 (OK). Kode status ini memverifikasi bahwa *request* untuk membuat PO telah berhasil diproses oleh server tanpa kegagalan. Waktu respons yang tercatat adalah 95 milidetik (ms), mengindikasikan bahwa kinerja *end-point* sangat efisien dan responsif dalam memproses *request* yang kompleks. Setelah PO berhasil dibuat, notifikasi “*Success. You have edited the Purchase Order!*” ditampilkan, mengonfirmasi bahwa PO telah tersimpan dan siap dialihkan ke tahap persetujuan berikutnya.



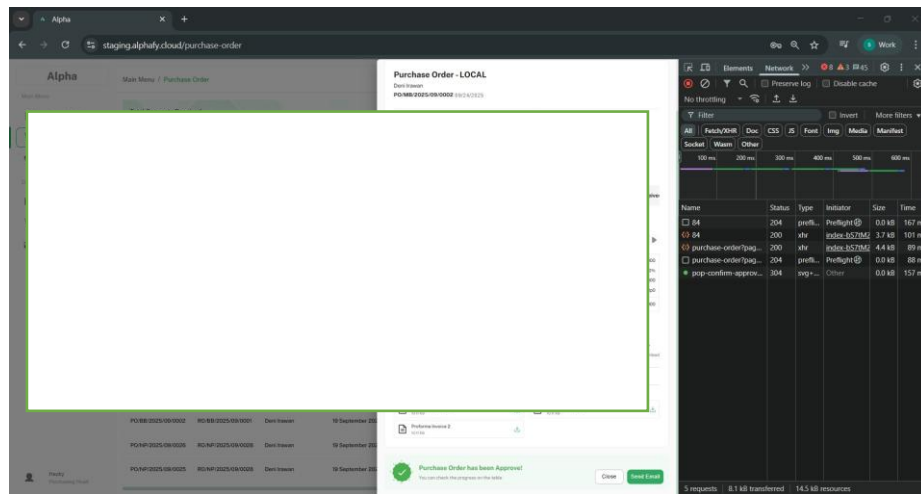
Gambar 3. 29 Contoh Hasil *Testing* DO

Pengujian fungsional modul DO dilanjutkan dengan skenario pembuatan DO oleh *Purchasing Staff*. Pembuatan DO ini seperti pada gambar 3.29 merupakan tindak lanjut dari PO yang telah disetujui, dan bertujuan untuk memproses pengiriman barang dari pemasok ke gudang. Hasil pengujian yang diamati menunjukkan bahwa *end-point* utama yang menangani pengiriman data DO merespons dengan Status Code 200 (OK). Kode status tersebut memverifikasi bahwa *request* untuk membuat DO telah berhasil diproses oleh server tanpa adanya kesalahan teknis. Waktu respons yang tercatat bervariasi antara 45 hingga 74 milidetik (ms) untuk berbagai *end-point* terkait (*purchase-order*, *send email*). Kinerja yang cepat ini mengindikasikan bahwa sistem mampu memproses pembuatan dokumen DO dan mengirimkan notifikasi terkait secara efisien. Setelah proses selesai, notifikasi “*Success. successfully sent the email.*” ditampilkan, mengonfirmasi bahwa DO telah tersimpan di basis data dan notifikasi pengiriman telah berhasil dikirimkan.



Gambar 3. 30 Contoh Hasil *Testing Approve RO*

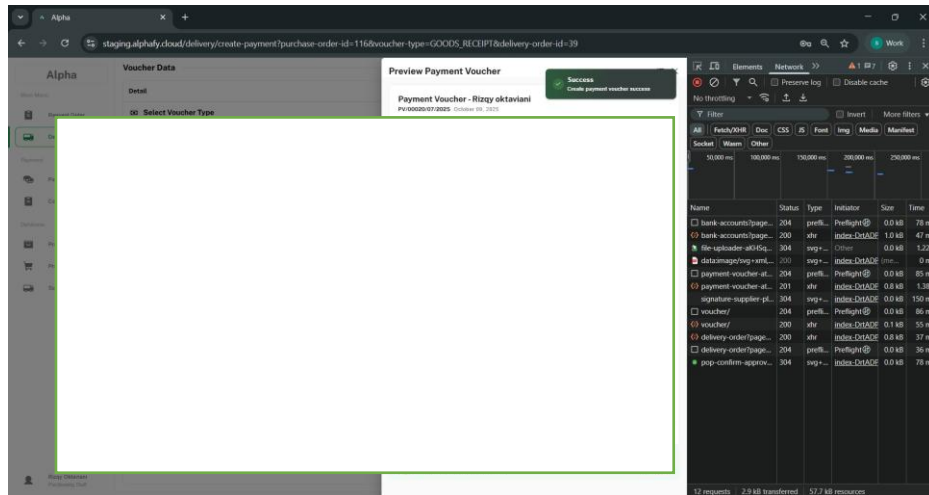
Pengujian fungsional modul RO dilanjutkan dengan skenario *approval RO* oleh *Procurement Head*. Pengujian ini bertujuan untuk memvalidasi alur persetujuan akhir RO seperti pada gambar 3.30, yang akan mengubah status dokumen menjadi *Approved* dan memungkinkan proses dilanjutkan ke pembuatan PO. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani aksi *approval* merespons dengan Status Code 200 (OK). Kode status ini memverifikasi bahwa *request* persetujuan RO telah berhasil diproses oleh server tanpa adanya kesalahan teknis. Waktu respons yang tercatat bervariasi antara 84 milidetik (ms) untuk *end-point* yang terkait dengan pembaruan status dan konfirmasi persetujuan. Kinerja yang efisien ini mengindikasikan bahwa sistem mampu memproses pembaruan status RO dengan cepat dan responsif. Setelah persetujuan berhasil dilakukan, notifikasi “*Success. Approved successfully!*” ditampilkan, mengonfirmasi bahwa status RO telah berhasil diperbarui dan dialihkan ke tahap proses berikutnya di basis data.



Gambar 3. 31 Contoh Hasil *Testing Approve PO*

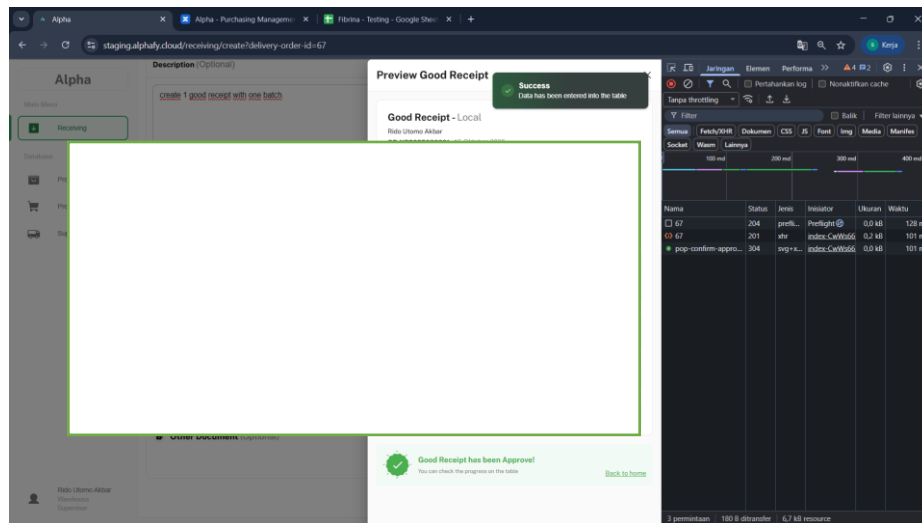
Pengujian fungsional modul PO dilanjutkan dengan skenario *approval PO* oleh *Procurement Head*. Pengujian ini bertujuan untuk memvalidasi alur persetujuan akhir PO, yang akan mengubah status dokumen menjadi *Approved* dan memungkinkan proses dilanjutkan ke pembuatan DO. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani aksi *approval* merespons dengan Status Code 200 (OK). Kode status tersebut memverifikasi bahwa *request* persetujuan PO telah berhasil diproses oleh server tanpa adanya kesalahan teknis seperti pada gambar 3.31. Waktu respons yang tercatat bervariasi antara 89 milidetik (ms) untuk *end-point* yang terkait dengan pembaruan status dan pengiriman notifikasi, menunjukkan kinerja *end-point* yang efisien dan responsif dalam memproses pembaruan status data. Setelah persetujuan berhasil dilakukan, notifikasi “*Purchase Order has been Approved!*” ditampilkan, mengonfirmasi bahwa status PO telah berhasil diperbarui dan notifikasi *email* telah berhasil dikirimkan.

3.3.1.11 Pelaksanaan Testing modul *GR* dan *Payment*



Gambar 3. 32 Contoh Hasil Testing Payment

Pengujian fungsional modul Payment dimulai dengan skenario pembuatan *Payment Voucher* oleh *Purchasing Staff*. Pengujian ini bertujuan untuk memvalidasi alur pembuatan dokumen pembayaran, termasuk pemrosesan *Voucher Type*, detail *Beneficiary*, dan dokumen faktur yang terkait. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani pembuatan *Payment Voucher* merespons dengan Status Code 200 (OK) seperti pada gambar 3.32. Kode status tersebut memverifikasi bahwa *request* untuk membuat *voucher* telah berhasil diproses oleh server tanpa adanya kesalahan teknis. Waktu respons yang tercatat menunjukkan durasi yang efisien, dengan *end-point* utama *payment-voucher* merespons dalam 78 milidetik (ms), sementara *end-point* konfirmasi persetujuan merespons dalam 37 milidetik (ms). Kinerja yang cepat ini mengindikasikan bahwa sistem mampu memproses pembuatan dokumen pembayaran yang kompleks secara responsif. Setelah proses selesai, notifikasi “*Success. Create payment voucher success*” dan “*Payment voucher has been sent!*” ditampilkan, mengonfirmasi bahwa *voucher* telah tersimpan dan berhasil dikirimkan ke alur persetujuan berikutnya.



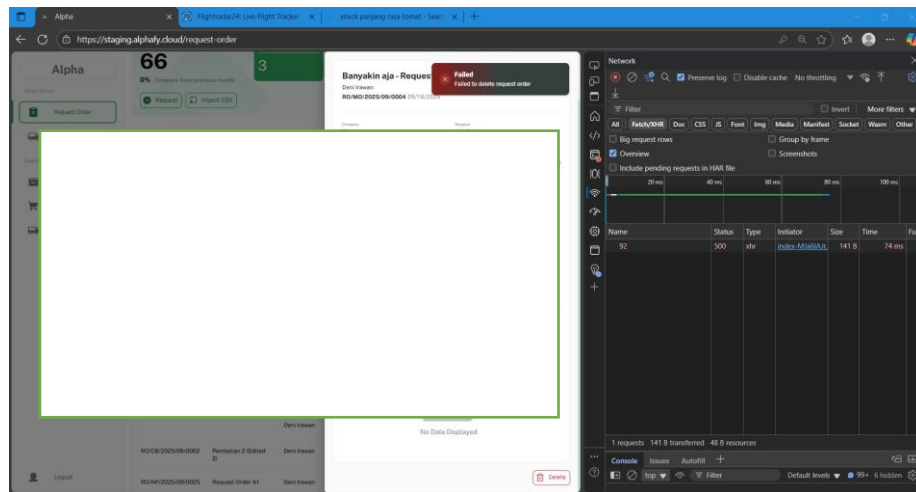
Gambar 3. 33 Contoh Hasil *Testing Good Receipt*

Pengujian fungsional modul GR dilaksanakan oleh peran *Warehouse* setelah DO tersedia. Pengujian ini bertujuan untuk memvalidasi kemampuan sistem dalam memproses penerimaan barang ke gudang dan pembaruan inventaris. Skenario yang diuji, seperti terlihat pada gambar, adalah pembuatan *good receipt* dengan satu *batch* (*create 1 good receipt with one batch*) seperti pada gambar 3.33. Hasil pengujian yang diamati menunjukkan bahwa *end-point* yang menangani pembuatan GR merespons dengan Status Code 200 (OK). Kode status tersebut memverifikasi bahwa *request* untuk membuat GR telah berhasil diproses oleh *server* tanpa adanya kegagalan teknis. Waktu respons yang tercatat menunjukkan durasi yang efisien, dengan *end-point* utama *receiving/create* merespons dalam waktu sekitar 101 milidetik (ms). Kinerja yang cepat ini memastikan efisiensi dalam alur kerja *Warehouse*. Setelah proses selesai, notifikasi “Data has been entered into the table” dan “Good Receipt has been Approved!” ditampilkan, mengonfirmasi bahwa data penerimaan barang telah tersimpan dan telah melalui alur persetujuan.

| Test Case ID | Module | Sub-Module | Log | Log Type | Status | Run Date | PIC Test | PIC Dev | End Date | Attachment | Dev Note |
|--------------|----------------------|----------------|-----------------------------------|--|--------|----------|----------------------|--------------|-----------|------------|----------------------------------|
| 1 | TP - Alpha - 01 | Request Order | Create RO - Local | wrong placement - check figna | Minor | Done | 9/17/2025 - 5:10 PM | Sabrina Azmi | 9/17/2025 | | ■ Bug - TP - Alpha - 04 - 01.jpg |
| 2 | TP - Alpha - 04 - 02 | Request Order | Create RO - Local | The successfully uploaded file should be displayed as a list below the "Select a file" button and above the "Continue" button within the pop-up count other document at previous RO created doesn't match with real other document | Minor | Done | 9/17/2025 - 5:10 PM | Sabrina Azmi | 9/17/2025 | | ■ Bug - TP - Alpha - 04 - 02.jpg |
| 4 | TP - Alpha - 04 - 03 | Request Order | Create RO - Local | scenario, user upload 3 file then remove 1. at preview detail after submit RO, it detect 3 file not 2 file | Major | Done | 9/17/2025 - 5:10 PM | Sabrina Azmi | 9/17/2025 | | ■ Bug - TP - Alpha - 04 - 03.jpg |
| 5 | TP - Alpha - 08 - 01 | Request Order | Delete RO - Cancellation | Can't delete RO from cancelled status RO. The toast message should display the cancelled RO's ID | Major | Done | 9/18/2025 - 2:50 PM | Sabrina Azmi | 9/18/2025 | | ■ Bug - TP - Alpha - 07 - 01.jpg |
| 6 | TP - Alpha - 07 - 01 | Request Order | Cancel RO | Can't delete RO from cancelled status RO | Minor | Done | 9/19/2025 - 10:10 AM | Sabrina Azmi | 9/19/2025 | | ■ Cancel toast message - RO.jpg |
| 7 | TP - Alpha - 28 - 01 | Approve RO | Delete RO - Cancellation | Can't edit RO from cancelled status RO | Minor | Done | 9/19/2025 - 4:10 PM | Sabrina Azmi | 9/19/2025 | | ■ Bug - TP - Alpha - 12 - 01.jpg |
| 8 | TP - Alpha - 10 - 01 | Request Order | Export CSV | Button Export CSV doesn't exist | Minor | Done | 9/22/2025 - 11:35 AM | Sabrina Azmi | 9/22/2025 | | ■ Bug - TP - Alpha - 10 - 01.jpg |
| 9 | TP - Alpha - 12 - 01 | Request Order | Sort by company | The result of filter is not correct | Minor | Done | 9/22/2025 - 11:35 AM | Sabrina Azmi | 9/22/2025 | | ■ Bug - TP - Alpha - 12 - 01.jpg |
| 10 | TP - Alpha - 22 - 01 | Request Order | Selected by name | after select one of requester name, the result is not correct | Minor | Done | 9/22/2025 - 1:45 PM | Sabrina Azmi | 9/22/2025 | | ■ Bug - TP - Alpha - 22 - 01.jpg |
| 11 | TP - Alpha - 26 - 01 | Approve RO | Rejection notes | add reason before rejection, but at RO preview detail after rejection, the reason doesn't show (empty field) | Minor | Done | 9/22/2025 - 3:45 PM | Sabrina Azmi | 9/22/2025 | | ■ Bug - TP - Alpha - 26 - 01.jpg |
| 12 | TP - Alpha - 23 - 01 | List Overview | Selected by status | add "cancelled" status | Minor | Done | 9/23/2025 - 2:40 PM | Sabrina Azmi | 9/23/2025 | | ■ Bug - TP - Alpha - 23 - 01.jpg |
| 13 | TP - Alpha - 03 - 01 | Request Order | Import CSV | Should be Hyperion "CSV Template" | Minor | Done | 9/23/2025 - 2:40 PM | Sabrina Azmi | 9/23/2025 | | ■ Bug - TP - Alpha - 03 - 01.jpg |
| 14 | TP - Alpha - 04 - 01 | Request Order | Delete RO - Rejection | Can't delete RO from rejected status RO, error 400, response only can delete RO from draft | Major | Done | 9/23/2025 - 3:05 PM | Sabrina Azmi | 9/23/2025 | | ■ Bug - TP - Alpha - 04 - 01.jpg |
| 15 | TP - Alpha - 08 - 02 | Request Order | Delete RO - Cancellation | Can't delete RO from cancelled status RO, 400 status | Major | Done | 9/23/2025 - 4:20 PM | Sabrina Azmi | 9/23/2025 | | ■ Bug - TP - Alpha - 08 - 02.jpg |
| 16 | TP - Alpha - 11 - 01 | List Overview | Sort by date - range, menu upper | Data show 1-1 from date selected | Minor | Done | 9/23/2025 - 4:30 PM | Sabrina Azmi | 9/23/2025 | | ■ Bug - TP - Alpha - 11 - 01.jpg |
| 17 | TP - Alpha - 11 - 01 | List Overview | Sort by date - Object to filtered | When the date filter is applied and saved, the date displayed in the table does not change | Minor | Done | 9/23/2025 - 4:50 PM | Sabrina Azmi | 9/23/2025 | | ■ Bug - TP - Alpha - 11 - 01.jpg |
| 18 | TP - Alpha - 16 - 01 | List Overview | Sort by branch | When the date filter is applied and saved, the date displayed in the table does not change | Minor | Done | 9/23/2025 - 4:55 PM | Sabrina Azmi | 9/23/2025 | | ■ Bug - TP - Alpha - 16 - 01.jpg |
| 19 | TP - Alpha - 26 - 02 | Request Order | Rejection detail preview | no need delete and revise button after RO rejected | Minor | Done | 9/24/2025 - 1:35 PM | Sabrina Azmi | 9/24/2025 | | ■ Bug - TP - Alpha - 26 - 02.jpg |
| 20 | TP - Alpha - 04 - 03 | Request Order | Create RO - Local | At preview RO, column product should be name of product, not number | Minor | Done | 9/24/2025 - 3:00 PM | Sabrina Azmi | 9/24/2025 | | ■ Bug - TP - Alpha - 04 - 03.jpg |
| 21 | TP - Alpha - 40 - 01 | Purchase Order | Create PO - Local | Discount section, label label, can't create PO if quantity more than RO (warning should be show) num of quantity kaba mas rdn selection - check figna | Minor | Done | 9/24/2025 - 3:05 PM | Sabrina Azmi | 9/24/2025 | | ■ Bug - TP - Alpha - 40 - 01.jpg |
| 22 | TP - Alpha - 40 - 02 | Purchase Order | Create PO - Local | At preview PO error SKU Code, Product, and discount, minus document attachment and signature of purchasing staff | Minor | Done | 9/24/2025 - 3:30 PM | Sabrina Azmi | 9/24/2025 | | ■ Bug - TP - Alpha - 40 - 02.jpg |
| 23 | TP - Alpha - 40 - 03 | Purchase Order | Create PO - Local | error design at create PO - check figna, submit | Minor | Done | 9/24/2025 - 3:45 PM | Sabrina Azmi | 9/24/2025 | | ■ Bug - TP - Alpha - 40 - 03.jpg |

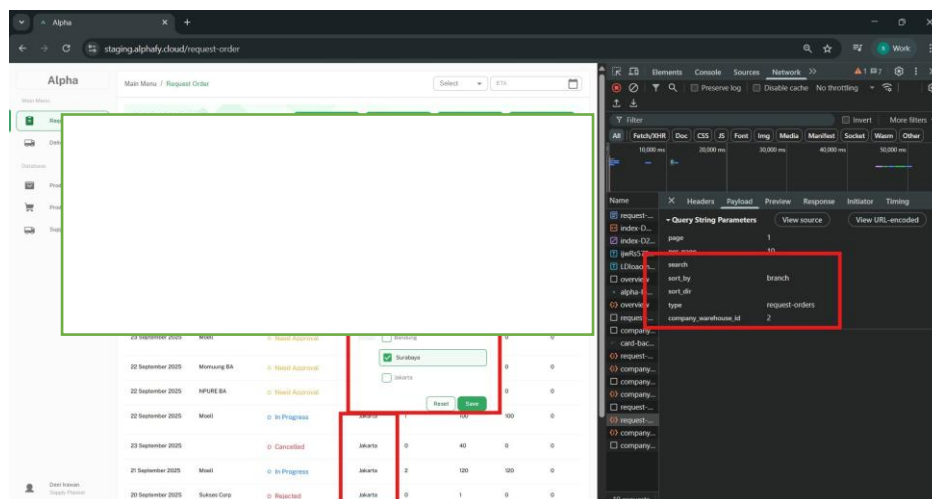
Gambar 3. 34 Hasil Test Log Alpha

Dalam pengujian yang telah dilakukan mulai dari modul RO hingga *payment* yang melibatkan seluruh role. Terdapat banyak temuan *error* atau *bug* yang muncul. Hal ini ditemukan bukan hanya dari pengujian berdasarkan interface tetapi juga pengujian data seperti pada gambar 3.34. Pengujian *interface* dilihat dengan acuan dari design yang telah dibuat oleh *UI/UX Designer* pada Figma seperti Gambar 3.19, menguji apakah setiap komponen sudah sesuai, tidak ada komponen yang hilang atau delay dalam penampilan data. Sedangkan pengujian data mengacu pada pelemparan data yang di input dan hasil yang diberikan, apakah data yang dimasukan dan di tampilkan sesuai, pengujian ini dilihat dari hasil response pada *end-point* yang diberikan. Pada proyek Alpha terdapat 64 bug yang ditemukan, mulai dari *bug minor* hingga *major*.



Gambar 3. 35 Contoh *Bug Major*

Pada gambar 3.35 merupakan contoh *bug major* yang ditemukan ketika ingin menguji *cancel RO* menyebabkan *blocking* sehingga pengujian tidak dapat dilanjutkan ke tahap berikutnya. Hal ini dapat dilihat dari status yang dikeluarkan oleh end-point berupa 500 dengan permasalahan pada backend yang belum membuat *development* untuk *cancel RO*.



Gambar 3. 36 Contoh *Bug Minor*

Selain bug major yang ditemukan sebelumnya, terdapat pula temuan bug minor, seperti yang gambar 3.36. Bug jenis ini biasanya muncul tanpa mengakibatkan blocking pada alur kerja utama

pengguna. Pada gambar 3.36 menunjukkan adanya *error* saat pelemparan data, *action* yang dilakukan tidak memberikan hasil yang sesuai meskipun response 200. Jika hal ini terjadi, perlu diperhatikan *payload* dari end-point yang ada untuk mengidentifikasi permasalahan yang ada.

3.3.2 Kendala yang Ditemukan

Selama proses praktik kerja magang sebagai QA *Intern* di PT. Sukses Corp International, ditemukan beberapa tantangan dan kendala yang menjadi bagian penting dari proses adaptasi dan pembelajaran. Kendala-kendala ini diatasi melalui inisiatif pribadi, komunikasi aktif, dan adaptasi cepat terhadap lingkungan kerja yang dinamis.

1. Tantangan Struktur Organisasi dan Pembimbingan Fungsional

Kendala utama teridentifikasi dari struktur tim yang unik, di mana posisi yang diemban adalah satu-satunya peran QA *Intern*. Karena tidak tersedianya atasan atau senior langsung dengan peran QA yang serupa dalam tim, acuan pembimbingan fungsional harian yang spesifik dari seorang QA senior tidak dimiliki. Konsekuensi dari situasi ini adalah diharuskannya pembentukan standar kerja secara mandiri. Hal tersebut menuntut adanya inisiatif yang tinggi dan kemampuan pembelajaran mandiri yang kuat untuk setiap pengambilan keputusan teknis yang berkaitan dengan metodologi pengujian, prosedur pelaporan *bug*, dan dokumentasi.

2. Keterbatasan Dokumentasi dan Standar Pengujian

Kesulitan signifikan dihadapi pada fase awal proses pengujian karena ketiadaan dokumentasi pengujian (*testing documentation*) yang terstruktur dari proyek-proyek terdahulu. Selain itu, standar pengujian baku yang jelas untuk QA, seperti format penulisan *test case* atau kriteria penentuan tingkat keparahan *bug*, belum ditetapkan secara formal. Kondisi ini memperpanjang waktu adaptasi yang diperlukan pada proses pengujian

di tahap awal. Oleh sebab itu, serangkaian diskusi intensif perlu dilakukan bersama *UI/UX Designer* dan tim *Engineer* untuk mendefinisikan serta menyepakati mekanisme pengujian yang paling efektif untuk aplikasi Fibrina dan Alpha.

3. Adaptasi Awal dan Kurva Pembelajaran Teknis

Kurva pembelajaran yang signifikan teridentifikasi selama periode magang, mengingat bahwa peran ini merupakan pengalaman pertama sebagai QA profesional. Proses adaptasi yang diperlukan mencakup pemahaman mendalam mengenai alur bisnis perusahaan yang kompleks, khususnya terkait produk Fibrina dan Alpha, dan penguasaan alat bantu pengujian. Secara spesifik, penguasaan alat ini meliputi keterampilan dalam *scripting* dan implementasi otomatisasi pengujian menggunakan Katalon Studio Enterprise. Kondisi tersebut menuntut adanya upaya proaktif dalam pencarian sumber daya belajar dan penyediaan waktu tambahan untuk menguasai keterampilan teknis yang dibutuhkan secara mandiri.

4. Isu Koordinasi Lintas Divisi

Dalam kolaborasi tim yang intensif, komunikasi yang kurang efektif terjadi beberapa kali di antara pihak *UI/UX Designer*, *Back-End Developer*, dan *Front-End Developer*. Ketidakselarasan komunikasi ini umumnya berkaitan dengan perbedaan interpretasi terhadap spesifikasi fungsional atau alur desain yang telah ditetapkan. Hal ini kemudian berakibat pada terhambatnya alur pengujian yang sedang dilaksanakan. Untuk memitigasi isu tersebut, inisiatif diambil untuk memverifikasi ulang alur yang terindikasi janggal dengan *Designer* sebelum setiap *bug* dilaporkan kepada tim *Developer*. Upaya verifikasi ini berfungsi untuk memastikan bahwa masalah yang disampaikan telah divalidasi berdasarkan spesifikasi desain yang akurat.

3.3.3 Solusi atas Kendala yang Ditemukan

Setiap kendala yang dihadapi selama praktik kerja magang dijadikan peluang untuk menunjukkan inisiatif, adaptasi, dan tanggung jawab. Berikut adalah solusi proaktif yang diterapkan untuk mengatasi tantangan yang muncul:

1. Mengambil Alih Pembelajaran Mandiri

Pelaksanaan pembelajaran mandiri dilakukan secara aktif terkait sistematika dan *testing flow*, serta penyesuaiannya dengan kebutuhan aplikasi internal perusahaan. Kegiatan yang dijalankan mencakup pendalaman penguasaan alat bantu otomatisasi pengujian. Upaya ini dilaksanakan untuk mengatasi tantangan yang timbul akibat tidak adanya atasan langsung dengan *role* yang serupa di dalam tim, sehingga berkontribusi pada percepatan proses adaptasi teknis yang dibutuhkan.

2. Menciptakan Dokumentasi Standar Perusahaan

Inisiatif penyusunan dokumentasi yang terstruktur dilakukan guna mengatasi ketiadaan dokumentasi pengujian sebelumnya. Dokumentasi yang dihasilkan mencakup *Test Plan*, *Test Skenario*, dan *Test Log* yang rinci untuk setiap modul yang diuji. Dokumentasi yang telah dibuat tersebut kemudian berfungsi sebagai acuan standar pengujian resmi yang dapat digunakan oleh QA di masa mendatang.

3. Mengoptimalkan Komunikasi Fungsional

Perbaikan dilakukan pada mekanisme komunikasi dengan tim *Designer* dan *Developer* dengan tujuan meminimalkan potensi miskomunikasi. Perbaikan tersebut diimplementasikan melalui prosedur verifikasi ulang alur yang terindikasi janggal dengan *Designer* sebelum setiap *bug* dilaporkan kepada *Developer*. Prosedur ini dilaksanakan untuk memastikan bahwa setiap masalah yang dilaporkan telah divalidasi dan sesuai dengan spesifikasi yang benar.

4. Memperbaiki Efektivitas Pelaporan

Perbaikan kualitas pada pelaporan *bug* menjadi fokus utama. Hal ini diwujudkan dengan memastikan bahwa setiap laporan yang disampaikan kepada *Back-End Developer* atau *Front-End Developer* memiliki sasaran yang tepat, serta dilengkapi dengan langkah reproduksi dan bukti pendukung yang jelas. Peningkatan kualitas pelaporan tersebut ditujukan untuk mempercepat proses *debugging* dan *deployment* yang dilaksanakan oleh tim *Engineer*.

