

Nama: Andhika Pratama
NIM: 09021182328025
Kelas: TI L3 23

Pemrograman Web II - Progress Report 2

Link Project: <https://github.com/aristelcrowley/mYkasir>

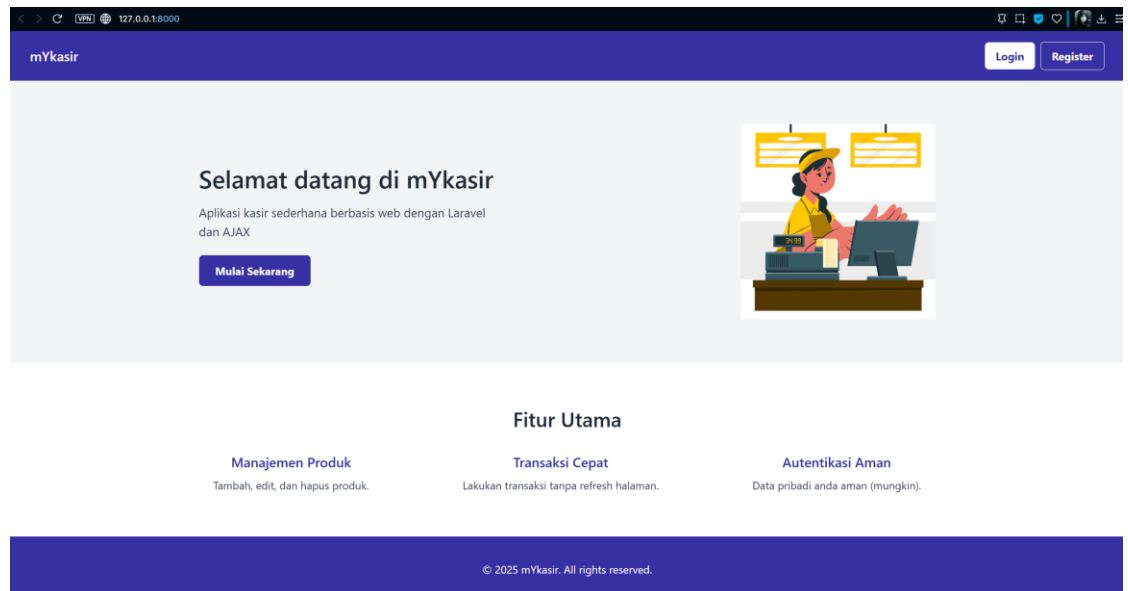
Link Dokumentasi Postman: <https://documenter.getpostman.com/view/40551639/2sB2qcE1tp>

1. Implementasi project yang telah di lakukan (View/Blade, Javascript, CSS, Controller, atau Desain Model / DB)

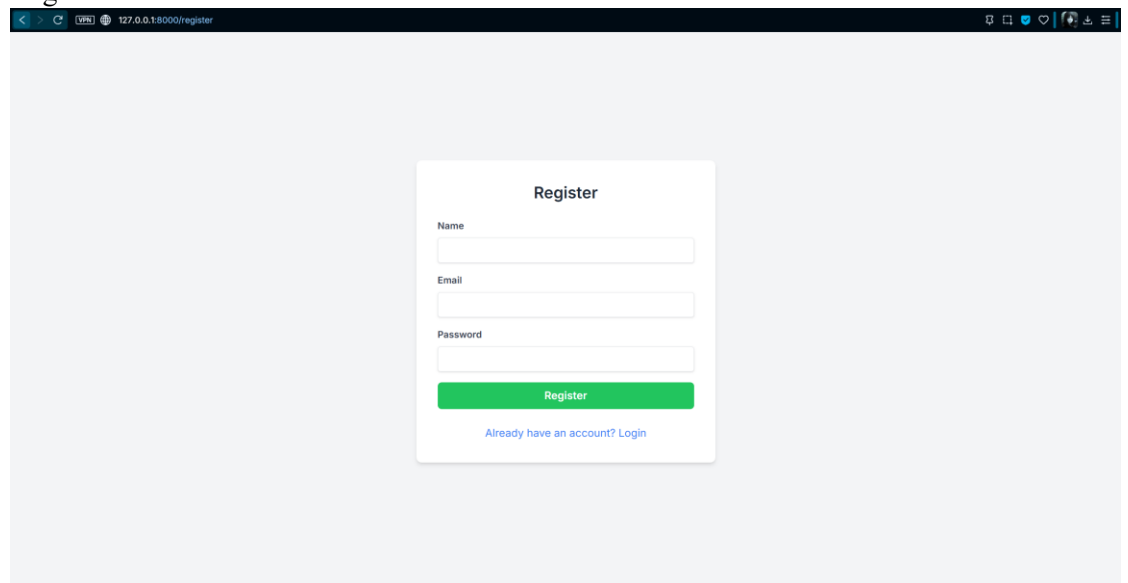
note: untuk kodingannya, hanya saya ss sebagian saja pak/bu, selebihnya ada di

- a. Implementasi Project

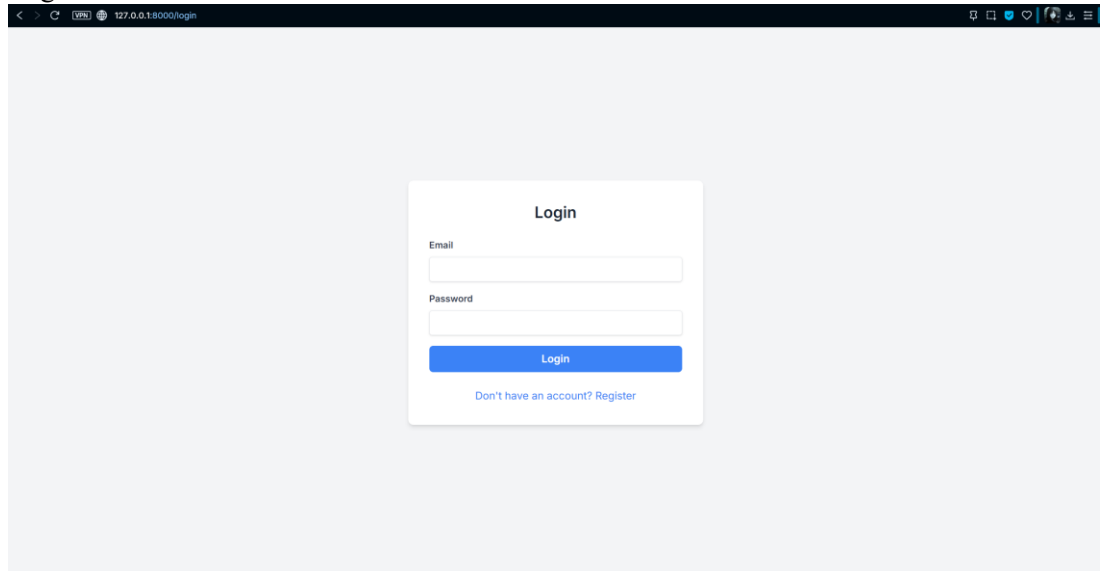
- 1) Home



- 2) Register



3) Login



127.0.0.1:8000/login

Login

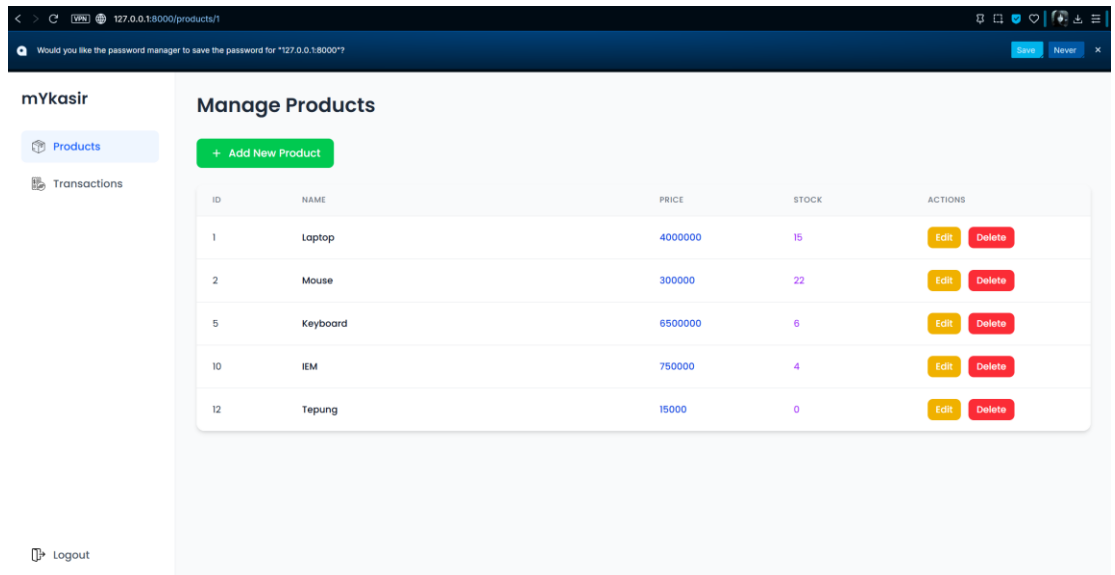
Email

Password

Login

[Don't have an account? Register](#)

4) Products



127.0.0.1:8000/products/1

Would you like the password manager to save the password for "127.0.0.1:8000"? [Save](#) [Never](#)

mYkasir

- Products
- Transactions

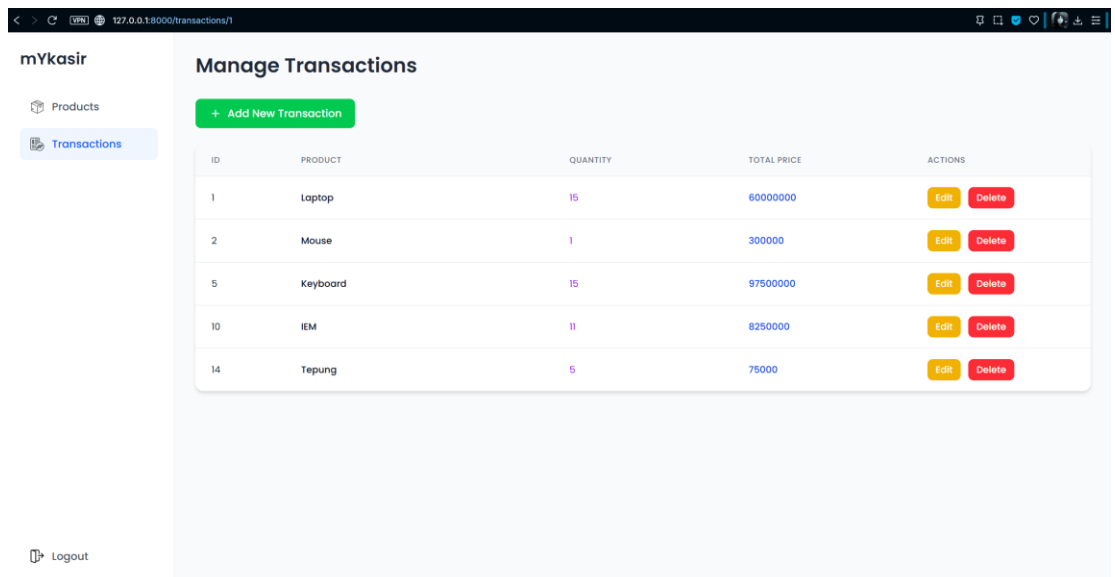
Logout

Manage Products

+ Add New Product

ID	NAME	PRICE	STOCK	ACTIONS
1	Laptop	4000000	15	Edit Delete
2	Mouse	300000	22	Edit Delete
5	Keyboard	6500000	6	Edit Delete
10	IEM	750000	4	Edit Delete
12	Tepung	15000	0	Edit Delete

5) Transactions



127.0.0.1:8000/transactions/1

mYkasir

- Products
- Transactions

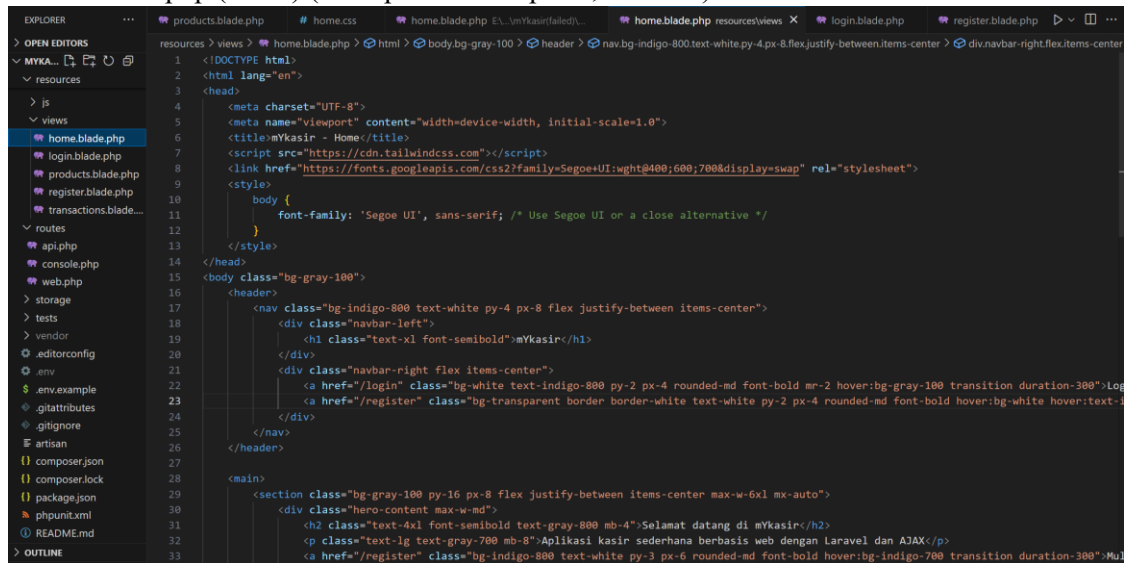
Logout

Manage Transactions

+ Add New Transaction

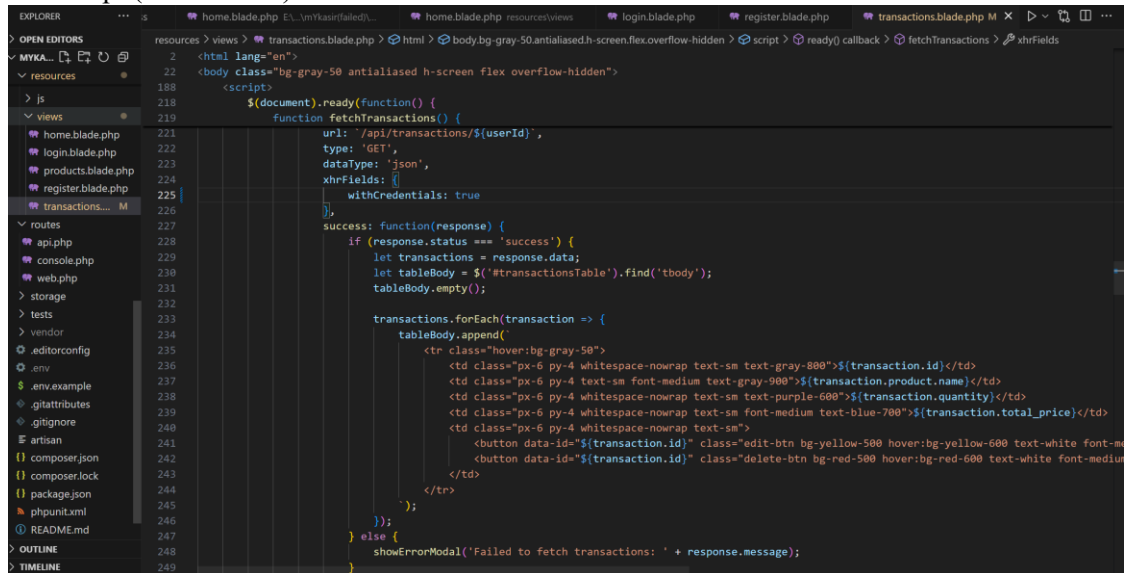
ID	PRODUCT	QUANTITY	TOTAL PRICE	ACTIONS
1	Laptop	15	60000000	Edit Delete
2	Mouse	1	300000	Edit Delete
5	Keyboard	15	97500000	Edit Delete
10	IEM	11	8250000	Edit Delete
14	Tepung	5	75000	Edit Delete

b. Views/Blade.php (home) (tidak pakai css terpisah, tailwind)



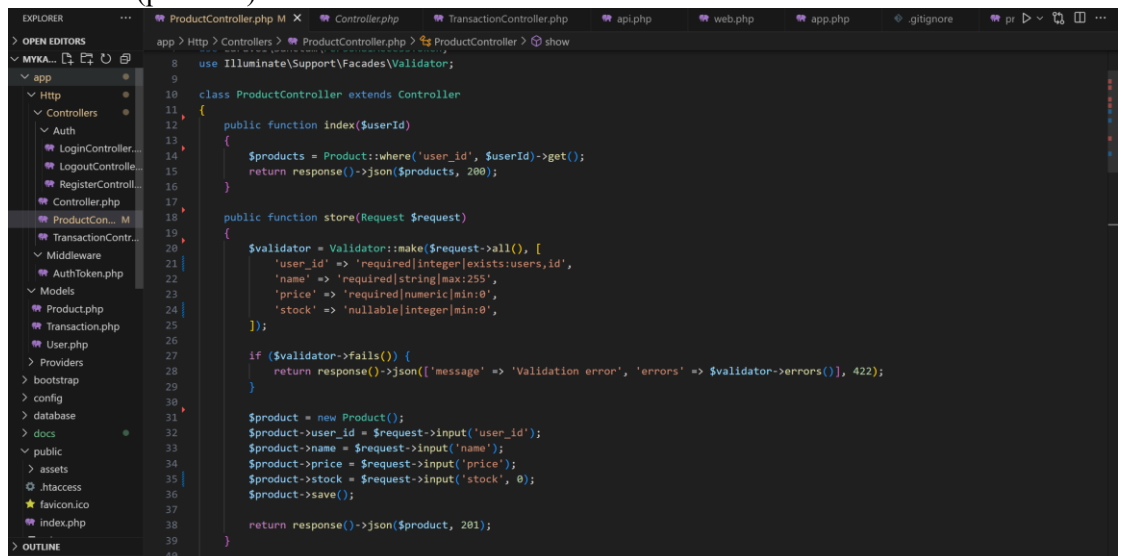
```
1 <DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>mKasir - Home</title>
7 <script src="https://cdn.tailwindcss.com"></script>
8 <link href="https://fonts.googleapis.com/css2?family=Segoe+UI:wght@400;600;700&display=swap" rel="stylesheet">
9 <style>
10 body {
11 font-family: 'Segoe UI', sans-serif; /* Use Segoe UI or a close alternative */
12 }
13 </style>
14 </head>
15 <body class="bg-gray-100">
16 <header>
17 <nav class="bg-indigo-800 text-white py-4 px-8 flex justify-between items-center">
18 <div class="navbar-left">
19 <h1 class="text-xl font-semibold">mKasir</h1>
20 </div>
21 <div class="navbar-right flex items-center">
22 <a href="/login" class="bg-white text-indigo-800 py-2 px-4 rounded-md font-bold mr-2 hover:bg-gray-100 transition duration-300">Log
23 <a href="/register" class="bg-transparent border border-white text-white py-2 px-4 rounded-md font-bold hover:bg-white hover:text-
24 </div>
25 </nav>
26 </header>
27
28 <main>
29 <section class="bg-gray-100 py-16 px-8 flex justify-between items-center max-w-6xl mx-auto">
30 <div class="hero-content max-w-md">
31 <h2 class="text-4xl font-semibold text-gray-800 mb-4">Selamat datang di mKasir</h2>
32 <p class="text-lg text-gray-700 mb-8">Aplikasi kasir sederhana berbasis web dengan Laravel dan AJAX</p>
33 <a href="/register" class="bg-indigo-800 text-white py-3 px-6 rounded-md font-bold hover:bg-indigo-700 transition duration-300">Mul
```

c. Javascript (transactions)



```
22 <html lang="en">
23 <body class="bg-gray-50 antialiased h-screen flex overflow-hidden">
24 <script>
25 $(document).ready(function() {
26 function fetchTransactions() {
27 url: '/api/transactions/${userId}',
28 type: 'GET',
29 dataType: 'json',
30 xhrFields: {
31 withCredentials: true
32 },
33 success: function(response) {
34 if (response.status === 'success') {
35 let transactions = response.data;
36 let tableBody = $('#transactionsTable').find('tbody');
37 tableBody.empty();
38 transactions.forEach(transaction => {
39 tableBody.append(
40 <tr class="hover:bg-gray-50">
41 <td class="px-6 py-4 whitespace-nowrap text-sm text-gray-800">${transaction.id}</td>
42 <td class="px-6 py-4 text-sm font-medium text-gray-900">${transaction.product.name}</td>
43 <td class="px-6 py-4 whitespace-nowrap text-sm text-purple-600">${transaction.quantity}</td>
44 <td class="px-6 py-4 whitespace-nowrap text-sm font-medium text-blue-700">${transaction.total_price}</td>
45 <td class="px-6 py-4 whitespace-nowrap text-sm">
46 <button data-id="${transaction.id}" class="edit-btn bg-yellow-500 hover:bg-yellow-600 text-white font-medium
47 <button data-id="${transaction.id}" class="delete-btn bg-red-500 hover:bg-red-600 text-white font-medium
48 </td>
49 </tr>
50 </td>
51 </tr>
52 </td>
53 </td>
54 </td>
55 </td>
56 </td>
57 </td>
58 </td>
59 </td>
60 </td>
61 </td>
62 </td>
63 </td>
64 </td>
65 </td>
66 </td>
67 </td>
68 </td>
69 </td>
70 </td>
71 </td>
72 </td>
73 </td>
74 </td>
75 </td>
76 </td>
77 </td>
78 </td>
79 </td>
80 </td>
81 </td>
82 </td>
83 </td>
84 </td>
85 </td>
86 </td>
87 </td>
88 </td>
89 </td>
90 </td>
91 </td>
92 </td>
93 </td>
94 </td>
95 </td>
96 </td>
97 </td>
98 </td>
99 </td>
100 </td>
101 </td>
102 </td>
103 </td>
104 </td>
105 </td>
106 </td>
107 </td>
108 </td>
109 </td>
110 </td>
111 </td>
112 </td>
113 </td>
114 </td>
115 </td>
116 </td>
117 </td>
118 </td>
119 </td>
120 </td>
121 </td>
122 </td>
123 </td>
124 </td>
125 </td>
126 </td>
127 </td>
128 </td>
129 </td>
130 </td>
131 </td>
132 </td>
133 </td>
134 </td>
135 </td>
136 </td>
137 </td>
138 </td>
139 </td>
140 </td>
141 </td>
142 </td>
143 </td>
144 </td>
145 </td>
146 </td>
147 </td>
148 </td>
149 </td>
150 </td>
151 </td>
152 </td>
153 </td>
154 </td>
155 </td>
156 </td>
157 </td>
158 </td>
159 </td>
160 </td>
161 </td>
162 </td>
163 </td>
164 </td>
165 </td>
166 </td>
167 </td>
168 </td>
169 </td>
170 </td>
171 </td>
172 </td>
173 </td>
174 </td>
175 </td>
176 </td>
177 </td>
178 </td>
179 </td>
180 </td>
181 </td>
182 </td>
183 </td>
184 </td>
185 </td>
186 </td>
187 </td>
188 </td>
189 </td>
190 </td>
191 </td>
192 </td>
193 </td>
194 </td>
195 </td>
196 </td>
197 </td>
198 </td>
199 </td>
200 </td>
201 </td>
202 </td>
203 </td>
204 </td>
205 </td>
206 </td>
207 </td>
208 </td>
209 </td>
210 </td>
211 </td>
212 </td>
213 </td>
214 </td>
215 </td>
216 </td>
217 </td>
218 </td>
219 </td>
220 </td>
221 </td>
222 </td>
223 </td>
224 </td>
225 </td>
226 </td>
227 </td>
228 </td>
229 </td>
230 </td>
231 </td>
232 </td>
233 </td>
234 </td>
235 </td>
236 </td>
237 </td>
238 </td>
239 </td>
240 </td>
241 </td>
242 </td>
243 </td>
244 </td>
245 </td>
246 </td>
247 </td>
248 </td>
249 </td>
250 </td>
251 </td>
252 </td>
253 </td>
254 </td>
255 </td>
256 </td>
257 </td>
258 </td>
259 </td>
260 </td>
261 </td>
262 </td>
263 </td>
264 </td>
265 </td>
266 </td>
267 </td>
268 </td>
269 </td>
270 </td>
271 </td>
272 </td>
273 </td>
274 </td>
275 </td>
276 </td>
277 </td>
278 </td>
279 </td>
280 </td>
281 </td>
282 </td>
283 </td>
284 </td>
285 </td>
286 </td>
287 </td>
288 </td>
289 </td>
290 </td>
291 </td>
292 </td>
293 </td>
294 </td>
295 </td>
296 </td>
297 </td>
298 </td>
299 </td>
300 </td>
301 </td>
302 </td>
303 </td>
304 </td>
305 </td>
306 </td>
307 </td>
308 </td>
309 </td>
310 </td>
311 </td>
312 </td>
313 </td>
314 </td>
315 </td>
316 </td>
317 </td>
318 </td>
319 </td>
320 </td>
321 </td>
322 </td>
323 </td>
324 </td>
325 </td>
326 </td>
327 </td>
328 </td>
329 </td>
330 </td>
331 </td>
332 </td>
333 </td>
334 </td>
335 </td>
336 </td>
337 </td>
338 </td>
339 </td>
340 </td>
341 </td>
342 </td>
343 </td>
344 </td>
345 </td>
346 </td>
347 </td>
348 </td>
349 </td>
350 </td>
351 </td>
352 </td>
353 </td>
354 </td>
355 </td>
356 </td>
357 </td>
358 </td>
359 </td>
360 </td>
361 </td>
362 </td>
363 </td>
364 </td>
365 </td>
366 </td>
367 </td>
368 </td>
369 </td>
370 </td>
371 </td>
372 </td>
373 </td>
374 </td>
375 </td>
376 </td>
377 </td>
378 </td>
379 </td>
380 </td>
381 </td>
382 </td>
383 </td>
384 </td>
385 </td>
386 </td>
387 </td>
388 </td>
389 </td>
390 </td>
391 </td>
392 </td>
393 </td>
394 </td>
395 </td>
396 </td>
397 </td>
398 </td>
399 </td>
400 </td>
401 </td>
402 </td>
403 </td>
404 </td>
405 </td>
406 </td>
407 </td>
408 </td>
409 </td>
410 </td>
411 </td>
412 </td>
413 </td>
414 </td>
415 </td>
416 </td>
417 </td>
418 </td>
419 </td>
420 </td>
421 </td>
422 </td>
423 </td>
424 </td>
425 </td>
426 </td>
427 </td>
428 </td>
429 </td>
430 </td>
431 </td>
432 </td>
433 </td>
434 </td>
435 </td>
436 </td>
437 </td>
438 </td>
439 </td>
440 </td>
441 </td>
442 </td>
443 </td>
444 </td>
445 </td>
446 </td>
447 </td>
448 </td>
449 </td>
450 </td>
451 </td>
452 </td>
453 </td>
454 </td>
455 </td>
456 </td>
457 </td>
458 </td>
459 </td>
460 </td>
461 </td>
462 </td>
463 </td>
464 </td>
465 </td>
466 </td>
467 </td>
468 </td>
469 </td>
470 </td>
471 </td>
472 </td>
473 </td>
474 </td>
475 </td>
476 </td>
477 </td>
478 </td>
479 </td>
480 </td>
481 </td>
482 </td>
483 </td>
484 </td>
485 </td>
486 </td>
487 </td>
488 </td>
489 </td>
490 </td>
491 </td>
492 </td>
493 </td>
494 </td>
495 </td>
496 </td>
497 </td>
498 </td>
499 </td>
500 </td>
501 </td>
502 </td>
503 </td>
504 </td>
505 </td>
506 </td>
507 </td>
508 </td>
509 </td>
510 </td>
511 </td>
512 </td>
513 </td>
514 </td>
515 </td>
516 </td>
517 </td>
518 </td>
519 </td>
520 </td>
521 </td>
522 </td>
523 </td>
524 </td>
525 </td>
526 </td>
527 </td>
528 </td>
529 </td>
530 </td>
531 </td>
532 </td>
533 </td>
534 </td>
535 </td>
536 </td>
537 </td>
538 </td>
539 </td>
540 </td>
541 </td>
542 </td>
543 </td>
544 </td>
545 </td>
546 </td>
547 </td>
548 </td>
549 </td>
550 </td>
551 </td>
552 </td>
553 </td>
554 </td>
555 </td>
556 </td>
557 </td>
558 </td>
559 </td>
560 </td>
561 </td>
562 </td>
563 </td>
564 </td>
565 </td>
566 </td>
567 </td>
568 </td>
569 </td>
570 </td>
571 </td>
572 </td>
573 </td>
574 </td>
575 </td>
576 </td>
577 </td>
578 </td>
579 </td>
580 </td>
581 </td>
582 </td>
583 </td>
584 </td>
585 </td>
586 </td>
587 </td>
588 </td>
589 </td>
590 </td>
591 </td>
592 </td>
593 </td>
594 </td>
595 </td>
596 </td>
597 </td>
598 </td>
599 </td>
600 </td>
601 </td>
602 </td>
603 </td>
604 </td>
605 </td>
606 </td>
607 </td>
608 </td>
609 </td>
610 </td>
611 </td>
612 </td>
613 </td>
614 </td>
615 </td>
616 </td>
617 </td>
618 </td>
619 </td>
620 </td>
621 </td>
622 </td>
623 </td>
624 </td>
625 </td>
626 </td>
627 </td>
628 </td>
629 </td>
630 </td>
631 </td>
632 </td>
633 </td>
634 </td>
635 </td>
636 </td>
637 </td>
638 </td>
639 </td>
640 </td>
641 </td>
642 </td>
643 </td>
644 </td>
645 </td>
646 </td>
647 </td>
648 </td>
649 </td>
650 </td>
651 </td>
652 </td>
653 </td>
654 </td>
655 </td>
656 </td>
657 </td>
658 </td>
659 </td>
660 </td>
661 </td>
662 </td>
663 </td>
664 </td>
665 </td>
666 </td>
667 </td>
668 </td>
669 </td>
670 </td>
671 </td>
672 </td>
673 </td>
674 </td>
675 </td>
676 </td>
677 </td>
678 </td>
679 </td>
680 </td>
681 </td>
682 </td>
683 </td>
684 </td>
685 </td>
686 </td>
687 </td>
688 </td>
689 </td>
690 </td>
691 </td>
692 </td>
693 </td>
694 </td>
695 </td>
696 </td>
697 </td>
698 </td>
699 </td>
700 </td>
701 </td>
702 </td>
703 </td>
704 </td>
705 </td>
706 </td>
707 </td>
708 </td>
709 </td>
710 </td>
711 </td>
712 </td>
713 </td>
714 </td>
715 </td>
716 </td>
717 </td>
718 </td>
719 </td>
720 </td>
721 </td>
722 </td>
723 </td>
724 </td>
725 </td>
726 </td>
727 </td>
728 </td>
729 </td>
730 </td>
731 </td>
732 </td>
733 </td>
734 </td>
735 </td>
736 </td>
737 </td>
738 </td>
739 </td>
740 </td>
741 </td>
742 </td>
743 </td>
744 </td>
745 </td>
746 </td>
747 </td>
748 </td>
749 </td>
750 </td>
751 </td>
752 </td>
753 </td>
754 </td>
755 </td>
756 </td>
757 </td>
758 </td>
759 </td>
760 </td>
761 </td>
762 </td>
763 </td>
764 </td>
765 </td>
766 </td>
767 </td>
768 </td>
769 </td>
770 </td>
771 </td>
772 </td>
773 </td>
774 </td>
775 </td>
776 </td>
777 </td>
778 </td>
779 </td>
780 </td>
781 </td>
782 </td>
783 </td>
784 </td>
785 </td>
786 </td>
787 </td>
788 </td>
789 </td>
790 </td>
791 </td>
792 </td>
793 </td>
794 </td>
795 </td>
796 </td>
797 </td>
798 </td>
799 </td>
800 </td>
801 </td>
802 </td>
803 </td>
804 </td>
805 </td>
806 </td>
807 </td>
808 </td>
809 </td>
810 </td>
811 </td>
812 </td>
813 </td>
814 </td>
815 </td>
816 </td>
817 </td>
818 </td>
819 </td>
820 </td>
821 </td>
822 </td>
823 </td>
824 </td>
825 </td>
826 </td>
827 </td>
828 </td>
829 </td>
830 </td>
831 </td>
832 </td>
833 </td>
834 </td>
835 </td>
836 </td>
837 </td>
838 </td>
839 </td>
840 </td>
841 </td>
842 </td>
843 </td>
844 </td>
845 </td>
846 </td>
847 </td>
848 </td>
849 </td>
850 </td>
851 </td>
852 </td>
853 </td>
854 </td>
855 </td>
856 </td>
857 </td>
858 </td>
859 </td>
860 </td>
861 </td>
862 </td>
863 </td>
864 </td>
865 </td>
866 </td>
867 </td>
868 </td>
869 </td>
870 </td>
871 </td>
872 </td>
873 </td>
874 </td>
875 </td>
876 </td>
877 </td>
878 </td>
879 </td>
880 </td>
881 </td>
882 </td>
883 </td>
884 </td>
885 </td>
886 </td>
887 </td>
888 </td>
889 </td>
890 </td>
891 </td>
892 </td>
893 </td>
894 </td>
895 </td>
896 </td>
897 </td>
898 </td>
899 </td>
900 </td>
901 </td>
902 </td>
903 </td>
904 </td>
905 </td>
906 </td>
907 </td>
908 </td>
909 </td>
910 </td>
911 </td>
912 </td>
913 </td>
914 </td>
915 </td>
916 </td>
917 </td>
918 </td>
919 </td>
920 </td>
921 </td>
922 </td>
923 </td>
924 </td>
925 </td>
926 </td>
927 </td>
928 </td>
929 </td>
930 </td>
931 </td>
932 </td>
933 </td>
934 </td>
935 </td>
936 </td>
937 </td>
938 </td>
939 </td>
940 </td>
941 </td>
942 </td>
943 </td>
944 </td>
945 </td>
946 </td>
947 </td>
948 </td>
949 </td>
950 </td>
951 </td>
952 </td>
953 </td>
954 </td>
955 </td>
956 </td>
957 </td>
958 </td>
959 </td>
960 </td>
961 </td>
962 </td>
963 </td>
964 </td>
965 </td>
966 </td>
967 </td>
968 </td>
969 </td>
970 </td>
971 </td>
972 </td>
973 </td>
974 </td>
975 </td>
976 </td>
977 </td>
978 </td>
979 </td>
980 </td>
981 </td>
982 </td>
983 </td>
984 </td>
985 </td>
986 </td>
987 </td>
988 </td>
989 </td>
990 </td>
991 </td>
992 </td>
993 </td>
994 </td>
995 </td>
996 </td>
997 </td>
998 </td>
999 </td>
1000 </td>
```

d. Controller (products)



```
8 use Illuminate\Support\Facades\Validator;
9
10 class ProductController extends Controller
11 {
12 public function index($userId)
13 {
14 $products = Product::where('user_id', $userId)->get();
15 return response()->json($products, 200);
16 }
17
18 public function store(Request $request)
19 {
20 $validator = Validator::make($request->all(), [
21 'user_id' => 'required|integer|exists:users,id',
22 'name' => 'required|string|max:255',
23 'price' => 'required|numeric|min:0',
24 'stock' => 'nullable|integer|min:0',
25 ]);
26
27 if ($validator->fails()) {
28 return response()->json(['message' => 'Validation error', 'errors' => $validator->errors()], 422);
29 }
30
31 $product = new Product();
32 $product->user_id = $request->input('user_id');
33 $product->name = $request->input('name');
34 $product->price = $request->input('price');
35 $product->stock = $request->input('stock', 0);
36 $product->save();
37
38 return response()->json($product, 201);
39 }
40 }
```

e. Model (products)

```

EXPLORER
  > OPEN EDITORS
  > MYKASIR
    > app
      > Http
      > Middleware
      > Models
        > Product.php
        > Transaction.php
        > User.php
      > Providers
      > bootstrap
      > config
      > database
      > docs
      > public
      > assets
      > .htaccess
      > favicon.ico
      > index.php
      > robots.txt
      > resources
      > css
      > js
      > views
      > home.blade.php
      > login.blade.php
      > products.blade.php
      > register.blade.php
      > transactions.blade.php
    > OUTLINE
    > TIMELINE

Transaction.php
  1 <?php
  2
  3 namespace App\Models;
  4
  5 use Illuminate\Database\Eloquent\Factories\HasFactory;
  6 use Illuminate\Database\Eloquent\Model;
  7
  8 class Transaction extends Model
  9 {
 10     use HasFactory;
 11
 12     protected $fillable = [
 13         'product_id',
 14         'quantity',
 15         'total_price',
 16         'user_id'
 17     ];
 18
 19     public function product()
 20     {
 21         return $this->belongsTo(Product::class);
 22     }
 23
 24     public function user()
 25     {
 26         return $this->belongsTo(User::class);
 27     }
 28
 29 }

```

f. DB

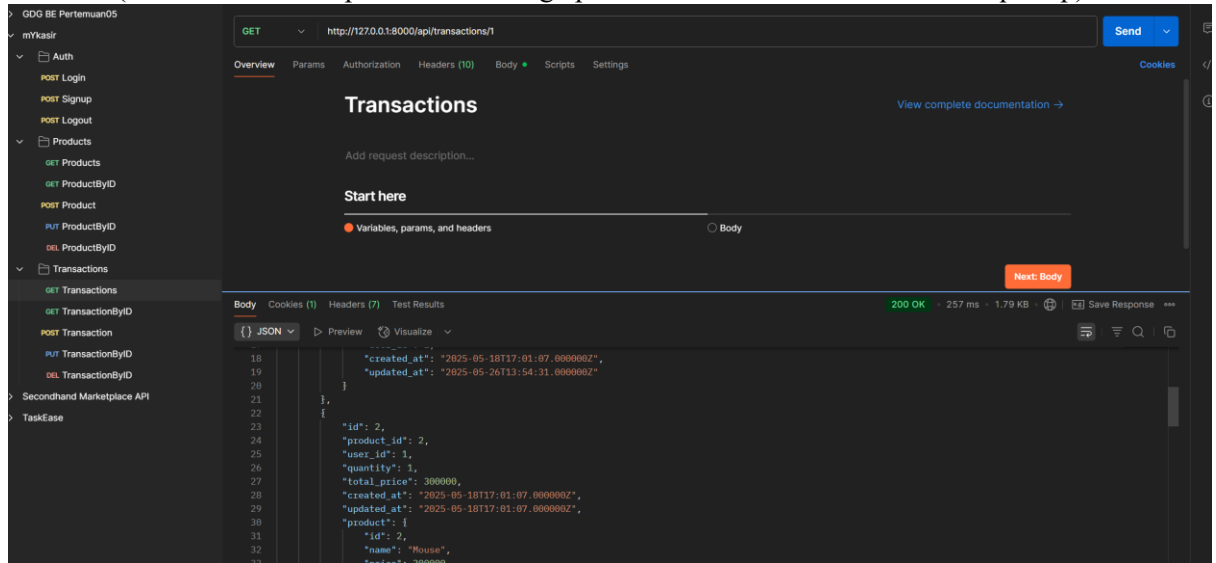


2. Permasalahan yang dihadapi selama fase implementasi (Tuliskan solusi jika sudah ditemukan)

Permasalahan pertama yang saya hadapi selama mengimplementasikan ide saya dalam kodingan ini adalah menerapkan autentikasi. Saya kesulitan untuk menggunakan auth:sanctum, yang menggunakan authorization header sebagai autentikasi, ataupun tools autentikasi bawaan lain dari laravel yang membuat user bisa mengakses seluruh halaman terproteksi tanpa login. Saya akhirnya memilih untuk membuat middleware sendiri berupa autentikasi berdasarkan auth_token, yang akan dicek validasinya dengan table dari auth:sanctum, tapi dicocokkan dengan auth_token yang di set (plain-textnya) di cookies.

Permasalahan saya selanjutnya adalah berusaha melindungi products/transactions users berdasarkan web routes, bukan api routes. Misalnya web url /products/13 akan menampilkan seluruh produk yang dimiliki oleh user dengan id 13. Saya tidak berhasil melindungi products yang dimiliki masing masing user (user bisa pergi ke url dengan user id lain dan melihat items mereka). Saya menemukan cara mengatasinya dengan melindungi api routes nya saja (di masing masing controller). Jadi sistem melakukan check user_id dari PersonalAccessToken user yang sedang ter-login, dan melakukan validasi tersebut di fetchproducts/fetchtransactions (controller index). Jadi, jika user berusaha mengakses item yang bukan miliknya, akan dilempar error 403:forbidden sebelum user tersebut dapat mengecek item apapun yang ada di halaman tersebut. Cara ini masih bisa di improve dengan cara “membungkus” api routesnya itu sendiri dengan middleware baru saja (sekalian agar bisa redirect ke item miliknya sendiri) ditambah dengan middleware autentikasi dipermasalahan pertama, tidak perlu cek di controller. Namun saya baru kepikiran setelah project ini telah selesai saya kerjakan. Mungkin untuk selanjutnya akan saya implementasikan cara tersebut.

3. Postman (dokumentasi di <https://documenter.getpostman.com/view/40551639/2sB2qcE1tp>)



```
Route::post('/login', [LoginController::class, 'login'])->withoutMiddleware(AuthToken::class);
Route::post('/signup', [RegisterController::class, 'register'])->withoutMiddleware(AuthToken::class);

Route::middleware(AuthToken::class)->group(function () {
    Route::post('/logout', [LogoutController::class, 'logout']);

    Route::get('/products/{userId}', [ProductController::class, 'index']);
    Route::post('/products', [ProductController::class, 'store']);
    Route::get('/product/{productId}', [ProductController::class, 'show']);
    Route::put('/products/{productId}', [ProductController::class, 'update']);
    Route::delete('/products/{productId}', [ProductController::class, 'destroy']);

    Route::get('/transactions/{userId}', [TransactionController::class, 'index']);
    Route::post('/transactions', [TransactionController::class, 'store']);
    Route::get('/transaction/{transactionId}', [TransactionController::class, 'show']);
    Route::put('/transactions/{transactionId}', [TransactionController::class, 'update']);
    Route::delete('/transactions/{transactionId}', [TransactionController::class, 'destroy']);

    Route::get('/products-list', [TransactionController::class, 'getProductList']);
});
```