

LAMPIRAN

<!DOCTYPE html>

```
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport"
content="width=device-width,
initial-scale=1.0" />
    <title>Document</title>
  </head>
  <body>
    <h1>Javascript & PHP push
notification</h1>
    <script>

navigator.serviceWorker.regist
er("serviceWorker.js");

        function enableNotif() {
Notification.requestPermission
().then((permission) => {
            if (permission ===
"granted") {

navigator.serviceWorker.ready.
then((sw) => {
                    sw.pushManager
                        .subscribe({

userVisibleOnly: true,

applicationServerKey:
                        "BKUf9wT-
uQHh-
QlJbtDIgSLlPwNNCaY98idD6RD_nob
INCzU-B9_3Zy-
ugpAoMeHbHV0GhZkcd3PA2SHx_Chb7
0",

                        })

                .then((subscription) => {
console.log(JSON.stringify(sub
cription));

                });

            }
        });
    }

console.log(location.pathname)
;
    </script>
```

```
<button
onclick="enableNotif()">Enable
</button>
</body>
</html>

@import
url('https://fonts.googleapis.
com/css2?family=Caveat:wght@40
0;500;600;700&family=Covered+B
y+Your+Grace&family=Oswald:wgh
t@200;300;400;500;600;700&fami
ly=Quicksand:wght@300;400;500;
600;700&Rajdhani:wght@300;400;
500;600;700&family=Roboto+Cond
ensed:ital,wght@0,300;0,400;0,
700;1,300;1,400;1,700&family=R
oboto+Slab:wght@100;200;300;40
0;500;600;700;800;900&family=Y
uji+Boku&family=Philosopher:it
al,wght@0,400;0,700;1,400;1,70
0&display=swap');

@tailwind base;
@tailwind components;
@tailwind utilities;

@layer components {
    .btn {
        @apply flex items-
center justify-center text-
slate-50 px-4 py-1.5 text-
center shadow-md font-semibold
rounded-md cursor-pointer;
    }
    .btn.btn-sm{
        @apply py-1 px-2 font-
medium text-sm
    }
    .btn.btn-xs{
        @apply py-0.5 px-1
font-medium text-xs
    }

    .input-group {
        @apply relative flex
flex-col justify-center my-2;
    }
    .input-group.input-sm {
        @apply !my-1;
    }
    .input-group label {
```

```

        @apply font-bold ml-
0.5 text-base text-primary
capitalize;
    }

    .input-group
:is(input:not([type=file],[typ
e=checkbox]), textarea){
        @apply w-full shadow
relative outline-none
focus:ring-1 ring-primary/30
px-2 py-0.5 border rounded-md
text-slate-500 bg-white font-
semibold border-slate-400;
    }

    .input-group.input-sm
input:not([type=file],[type=ch
eckbox]){
        @apply !px-2 !py-0
!text-sm;
    }

    .input-group > input:read-
only{
        @apply bg-slate-200
border-slate-400 text-slate-
700
    }

    .input-group >
input:is([type=file]) {
        @apply w-full relative
outline-none focus:ring-1
ring-primary/30 border border-
slate-400 rounded-md text-
slate-500 file:py-1
    }

    /* .input-group >
input[type="file"]{
        @apply w-full shadow
relative outline-none
focus:ring-1 ring-primary/30
px-2 py-0.5 border border-
slate-400 rounded-md text-
slate-500 bg-white font-
semibold;
    } */

    .input-group select{
        @apply w-full shadow
relative outline-none
focus:ring-1 ring-primary/30
px-2 py-0.5 border border-
slate-400 rounded-md text-
slate-500 bg-white font-
semibold;
    }

    .input-group.input-sm
select{

```

```

        @apply !px-1 text-sm
    }

    .input-group
input[currency]{
        @apply !font-robotoSlab
    }

    :is(input,select,
textarea).profil {
        @apply border-0 border-b
focus:!ring-0 focus:!border-
slate-400 focus:!text-slate-
500 p-0 w-full border-black;
    }

    /* navbar */
    .navbar {
        @apply flex flex-col
items-center justify-center w-
full px-5 py-2 bg-white
shadow-md lg:justify-center
lg:px-20 lg:items-center;
    }

    .navbar .navbar-nav{
        @apply items-center
gap-4 overflow-x-auto text-lg
font-medium tracking-wide
lg:flex first-line:text-slate-
700 whitespace-nowrap;
    }

    .navbar .navbar-nav .nav-
link{
        @apply flex items-
center gap-2 font-semibold
text-black transition-colors
cursor-pointer max-md:text-
black hover:text-indigo-400
hover:border-indigo-400;
    }

    .nav-link.aktiv {
        @apply !text-indigo-400
    }

    /* table */
    table {
        @apply w-full text-xs
!border-separate md:text-base
border-spacing-0;
    }

    /* table :is(thead, tbody)
tr{
        @apply text-black bg-
white border border-slate-300;
    }

```

```

    } */
    table tbody tr:nth-
child(odd) {
        @apply bg-white text-
slate-800
    }
    table tbody tr:nth-
child(odd) {
        @apply bg-slate-100
text-slate-800
    }
    td, th {
        @apply px-4 py-1
border-b border-l border-
slate-300;
    }
    th {
        @apply border-t;
    }

    :is(td,th):last-child {
        @apply border-r
    }

    table.transparent {
        @apply w-max
    }
    table.transparent thead
tr{
        @apply bg-transparent
text-slate-800;
    }
    table.transparent tbody tr
{
        @apply bg-transparent
text-slate-800
    }
    .transparent td,
.transparent th {
        @apply border-none
px-2 !w-max;
    }

    /* modal */
    .modal {
        @apply fixed top-0
left-0 z-[60] flex items-start
justify-center w-full max-h-
screen min-h-screen px-5
transition-all duration-500
opacity-0 pointer-events-none
bg-white/70 py-14 backdrop-
blur-sm;
    }

    .modal-dialog {

```

```

        @apply grid grid-rows-
[0fr] transition-all duration-
500;
    }
    .modal-content {
        @apply w-full flex
flex-col overflow-hidden bg-
white rounded-md shadow-lg
md:max-w-xl max-h-[35rem];
    }
    .modal-header {
        @apply py-4 px-5
border-b flex items-center
justify-between w-full min-w-
max
    }
    .modal-header .title {
        @apply text-xl font-
semibold tracking-wider
    }
    .modal-body{
        @apply overflow-x-
hidden overflow-y-auto
leading-relaxed;
    }

    .modal.active{
        @apply opacity-100
pointer-events-auto
    }

    .modal.active .modal-
dialog{
        @apply grid-rows-[1fr]
    }
    .modal.active .modal-
dialog .modal-content{
        @apply border
    }
    .modal .close{
        @apply relative h-
[25px] w-[25px] cursor-pointer
    }
    .modal .close::after {
        @apply content-[""] h-
[2px] w-full bg-black absolute
top-0 left-0 rotate-[45deg]
origin-top-left translate-x-1
translate-y-[3px]
    }
    .modal .close::before {
        @apply content-[""] h-
[2px] w-full bg-black absolute
bottom-0 left-0 -rotate-
[45deg] origin-bottom-left
    }

```

```

translate-x-1 -translate-y-
[3px]
}

/* card */
.card {
    @apply w-full rounded
    shadow-sm bg-gray-50;
}

.card .header {
    @apply flex items-
    center justify-between flex-
    wrap gap-y-1.5 gap-x-2 p-5
    border-b rounded-t
    dark:border-gray-600;
}

.card .header .title {
    @apply text-xl font-
    semibold text-gray-900
    capitalize dark:text-white;
}

.card .body {
    @apply p-5 ;
}

/* end card */

/* sidebar */
.sidebar-item {
    @apply flex items-center
    p-2 text-gray-900 rounded-lg
    dark:text-white hover:bg-gray-
    100 dark:hover:bg-gray-700;
}

.sidebar-icon {
    @apply w-5 h-5 text-
    gray-500 transition duration-
    75 dark:text-gray-400 group-
    hover:text-gray-900
    dark:group-hover:text-white;
}

.sidebar-item.active {
    @apply text-gray-200;
}

.image .action {
    @apply w-full absolute
    bottom-0 justify-center flex
    items-center bg-transparent
    hover:bg-black/80 transition-
    all duration-[0.6s] ease-
    linear py-2
}

#image-container {
    @apply [&_img]:h-[25rem]
    [&_img]:object-contain border
    bg-black shadow-lg relative;
}

.menu-card {
    @apply bg-white py-5
    shadow-md px-2 border-2
}

.menu-header {
    @apply font-bold text-lg
    lg:!text-xl uppercase mb-5
    text-center
}

.menu-body {
    @apply flex flex-wrap
    gap-2 w-full items-stretch
    !justify-center
}

.menu-items {
    @apply text-xs lg:!text-
    lg w-full p-4 gap-2 relative
}

.menu-title, .menu-items >
.title {
    @apply font-semibold
    whitespace-normal h-8 lg:!h-12
    line-clamp-2 transition-all
}

.menu-price, .menu-items >
.price {
    @apply mt-4 font-bold
    text-lime-600 transition-all
}

.menu-price > .sold,
.menu-items > .price > .sold {
    @apply text-xs font-
    normal text-black transition-
    all
}

.menu-umkm, .menu-items >
.umkm {
    @apply font-bold
    transition-all
}

.menu-items > .deskripsi {
    @apply text-xs lg:!text-
    sm bg-white translate-y-[100%]
    font-semibold duration-500
}

```

```

border-t shadow-md w-full
!absolute left-0 transition-
all bottom-0 overflow-hidden
flex justify-center items-
center h-[2.7rem]
}
.menu-items:hover >
.deskripsi {
  @apply -translate-y-[0%]
}
.card-slider-item {
  @apply transition-all
ease-linear
}

.card-slider-
item:has(>.menu-items:hover){
  @apply -translate-y-2
}

.options {
  @apply bg-white absolute
z-50 border rounded divide-y-2
overflow-y-auto max-h-[200px]
hidden !capitalize
}

.options > li {
  @apply p-2 cursor-
pointer select-none hover:bg-
slate-200;
}

.group-floating {
  @apply relative flex
gap-2 pt-4 pb-1 overflow-auto
bg-white border shadow-md ps-2
pe-4 ;
}

.group-floating > input {
  @apply border-0 px-0 py-
0 focus:border-0
placeholder:text-transparent
focus:ring-0 max-md:w-full
}

.group-floating >
input:placeholder-shown ~
label {
  @apply top-3 text-black
text-base
}

.group-floating > label,
.group-floating > input:focus
~ label {

```

```

@apply absolute text-xs
transition-all duration-500
left-7 text-slate-400 top-1;
}

.filter > .title {
  @apply font-bold text-lg
uppercase
}

.filter > .list {
  @apply text-sm font-
semibold
}

.filter .list-item {
  @apply flex items-center
gap-x-4
}

.filter
input[type="checkbox"] {
  @apply border-slate-300
}
main .content {
  @apply border w-full px-
4 py-4 grid grid-cols-3
transition-all duration-300
}

main .content >
:is(.title, .destinations) {
  @apply !flex flex-col
}
main .content > .title >
h4 {
  @apply font-bold text-lg
font-quickSand capitalize
}
main .content > .title >
span {
  @apply text-sm font-
quickSand capitalize
}

main .destinations >
:is(.time){
  @apply font-bold text-lg
}

/* #toggle-rotate.aktiv {
  @apply rotate-180
} */

```

```

        button#submit:disabled {
            @apply bg-red-600
            cursor-not-allowed
        }

        .message-error {
            @apply text-red-400
            text-sm ms-1 mt-0.5 font-
            medium
        }

        .container-alert{
            @apply fixed top-0
            right-0 flex flex-col items-
            end mt-3 mr-3 z-20
        }
        .alert .content {
            @apply px-3 py-1.5
            font-semibold min-h-0 flex
            items-center justify-between
            gap-x-4 rounded-md border
            shadow transition-all
            duration-500 overflow-hidden
        }

        .alert.danger .content{
            @apply bg-red-100 text-
            red-800 border-red-800
        }

        .alert .dialog {
            @apply min-h-0
        }

        .alert {
            @apply grid grid-rows-
            [0fr] overflow-hidden
            transition-all duration-500
            mb-0;
        }

        .alert:has(.show),
        .alert.show{
            @apply !grid-rows-[1fr]
            !mb-3
        }

        .alert.success .content{
            @apply bg-indigo-100
            border-indigo-800 text-indigo-
            800
        }

        .notif {

```

```

            @apply relative flex
            items-center w-12 overflow-
            hidden h-6 bg-red-700 rounded-
            full transition-all
        }

        .notif label {
            @apply w-4 h-4
            transition-all duration-300
            rounded-full bg-white absolute
            translate-x-1 cursor-pointer
        }

        input#notif:checked +
        .notif{
            @apply bg-indigo-600
        }
        input#notif:checked +
        .notif label{
            @apply translate-x-
            [calc(100%+12px)]
        }

        .progress-seat {
            @apply relative flex
            items-center justify-center h-
            2 overflow-hidden bg-indigo-
            600 rounded-full w-28 ;
        }

        .progress-seat::after{
            @apply bg-slate-200;
        }

        main .content:hover {
            box-shadow: 6px 5px 11px
            rgb(199, 195, 195);
            /* animation: hover_animate
            1s; */
        }

        @keyframes hover_animate {
            0% {
                background: linear-
                gradient(90deg, rgba(2,0,36,1)
                0%, rgba(9,9,121,1) 35%,
                rgba(0,212,255,1) 100%);
            }
            100% {
                background: linear-
                gradient(120deg,
                rgba(2,0,36,1) 0%,
                rgba(9,9,121,1) 35%,
                rgba(0,212,255,1) 100%);
            }
        }

```

```

}

#toggle-rotate.aktiv {
  animation: rotate .8s;
}
@keyframes rotate {
  100% {
    transform: rotate(180deg)
    translateX(0.25rem);
  }
}

#tooltip::after {
  content: "";
  position: absolute;
  top: 100%;
  left: 50%;
  margin-left: -5px;
  border-width: 5px;
  border-style: solid;
  border-color: black
transparent transparent
transparent;
}

i.fas {
  cursor: pointer;
}

/* Next & previous buttons */
#image-container :is(.prev,
.next) {
  cursor: pointer;
  position: absolute;
  top: 50%;
  width: auto;
  padding: 16px;
  margin-top: -22px;
  color: rgba(255, 255, 255,
.3);
  font-weight: bold;
  font-size: 18px;
  transition: 0.6s ease;
  border-radius: 0 3px 3px 0;
  user-select: none;
}

/* Position the "next button"
to the right */
#image-container .next {
  right: 0;
  border-radius: 3px 0 0 3px;
}

/* On hover, add a black
background color with a little
bit see-through */
/* .prev:hover, .next:hover {
  background-color:
rgba(0,0,0,0.8);
  color: white;
} */

.image {
  display: none;
  position: relative;
  justify-content: center;
  align-items: center;
  flex-direction: column;
}

/* .image .action {

  width: 100%;
  position: absolute;
  bottom: 0px;
  display: flex;
  justify-content: center;
  align-items: center;
  background-color:
transparent;
  transition: all ease 150ms;
} */

/* .numbertext:has(+
.action:hover),
.image:hover :is(.numbertext,
.action),
.image:hover ~ :is(.prev,
.next),
:is(.next, .prev):hover {
  background-color:
rgba(0,0,0,0.8);
  color: white;
} */

.image .delete {
  color: rgb(239 68 68);
  opacity: 0.2;
  transition: all ease 0.6s;
}

.image .update {
  color: rgb(59 130 246);
  opacity: 0.2;
  transition: all ease 0.6s;
}

div:has(> .image):hover
:is(.prev, .next, .action,
.numbertext) {
  background-color:
rgba(0,0,0,0.8);

```

```

    color: white;
}
div:has(> .image):hover
:is(.update, .delete){
    opacity: .6;
}

:is(.update, .delete):hover{
    opacity: 1 !important;
}

.numbertext {
    color: #f2f2f2;
    font-size: 12px;
    padding: 8px 12px;
    position: absolute;
    top: 0;
    border-radius: 3px 0 0 3px;
    transition: 0.6s ease;
}

.image:nth-child(1) {
    display: flex;
}

.fade {
    animation-name: fade;
    animation-duration: 1.5s;
}

@keyframes fade {
    from {
        opacity: .4;
    }
    to {
        opacity: 1;
    }
}

#copy {
    position: relative;
    color: rgba(0,0,0,.8)
!important;
}

#copy::after {
    content: "dicopy";
    position: absolute;
    font-size: .9rem;
    font-family: cursive;
    top: -16px;
    left: -12px;
    white-space: nowrap;
    color: rgba(0,0,0,.8);
    display: none;
}

```

```

#copy.active::after{
    display: inline;
}

.tooltip {
    position: relative;
    display: inline-block;
}

.tooltip .tooltiptext {
    visibility: hidden;
    background-color: black;
    color: #fff;
    text-align: center;
    border-radius: 6px;
    padding: 3px 4px 6px 4px;
    position: absolute;
    z-index: 1;
    top: -4px;
    left: 160%;
    font-size: .9rem;
    font-family: "'Quicksand',
sans-serif";
}

.tooltip .tooltiptext::after {
    content: "";
    position: absolute;
    top: 50%;
    right: 100%;
    margin-top: -5px;
    border-width: 5px;
    border-style: solid;
    border-color: transparent
black transparent transparent;
}

.tooltip.active .tooltiptext {
    visibility: visible;
}

@keyframes ldio-qexq6n0axc {
    0% {
        opacity: 1;
        backface-visibility:
hidden;
        transform: translateZ(0)
scale(1.5,1.5);
    } 100% {
        opacity: 0;
        backface-visibility:
hidden;
        transform: translateZ(0)
scale(1,1);
    }
}

```

```

}
.lldio-qexq6n0axc div > div {
  position: absolute;
  width: 12px;
  height: 12px;
  border-radius: 50%;
  background: #1d0e0b;
  animation: ldio-qexq6n0axc
1s linear infinite;
}.lldio-qexq6n0axc div:nth-
child(1) > div {
  left: 74px;
  top: 44px;
  animation-delay: -0.875s;
}
.lldio-qexq6n0axc > div:nth-
child(1) {
  transform: rotate(0deg);
  transform-origin: 80px 50px;
}.lldio-qexq6n0axc div:nth-
child(2) > div {
  left: 65px;
  top: 65px;
  animation-delay: -0.75s;
}
.lldio-qexq6n0axc > div:nth-
child(2) {
  transform: rotate(45deg);
  transform-origin: 71px 71px;
}.lldio-qexq6n0axc div:nth-
child(3) > div {
  left: 44px;
  top: 74px;
  animation-delay: -0.625s;
}
.lldio-qexq6n0axc > div:nth-
child(3) {
  transform: rotate(90deg);
  transform-origin: 50px 80px;
}.lldio-qexq6n0axc div:nth-
child(4) > div {
  left: 23px;
  top: 65px;
  animation-delay: -0.5s;
}
.lldio-qexq6n0axc > div:nth-
child(4) {
  transform: rotate(135deg);
  transform-origin: 29px 71px;
}.lldio-qexq6n0axc div:nth-
child(5) > div {
  left: 14px;
  top: 44px;
  animation-delay: -0.375s;
}

```

```

.lldio-qexq6n0axc > div:nth-
child(5) {
  transform: rotate(180deg);
  transform-origin: 20px 50px;
}.lldio-qexq6n0axc div:nth-
child(6) > div {
  left: 23px;
  top: 23px;
  animation-delay: -0.25s;
}
.lldio-qexq6n0axc > div:nth-
child(6) {
  transform: rotate(225deg);
  transform-origin: 29px 29px;
}.lldio-qexq6n0axc div:nth-
child(7) > div {
  left: 44px;
  top: 14px;
  animation-delay: -0.125s;
}
.lldio-qexq6n0axc > div:nth-
child(7) {
  transform: rotate(270deg);
  transform-origin: 50px 20px;
}.lldio-qexq6n0axc div:nth-
child(8) > div {
  left: 65px;
  top: 23px;
  animation-delay: 0s;
}
.lldio-qexq6n0axc > div:nth-
child(8) {
  transform: rotate(315deg);
  transform-origin: 71px 29px;
}
.loadingio-spinner-spin-
t6woujb5b7 {
  width: 30px;
  height: 30px;
  display: inline-block;
  overflow: hidden;
  background: none;
}
.lldio-qexq6n0axc {
  width: 100%;
  height: 100%;
  position: relative;
  transform: translateZ(0)
scale(0.3);
  backface-visibility: hidden;
  transform-origin: 0 0; /*
see note above */
}
.lldio-qexq6n0axc div { box-
sizing: content-box; }

```

```

/* generated by
https://loading.io/ */

.box-shadow {
    box-shadow: -5px -5px 9px
rgb(255 255 255 / 45%), 5px
5px 9px rgb(94 104 121 / 30%);
}

/* slider card */
div:has(> .card-slider){
    width: 100%;
    position: relative;
    display: flex;
    flex-direction: column;
    align-items: start;
    justify-content: center;
    -webkit-user-select: none;
/* Safari */
    -ms-user-select: none; /*
IE 10 and IE 11 */
    user-select: none; /*
Standard syntax */
}
/* div{
    width: 100%;
} */

.card-slider {
    position: relative;
    display: inline-block;
    width: 100%;
    overflow: hidden;
    white-space: nowrap;
    margin: 0px;
    padding: 0 5px 10px 5px;
    cursor: pointer;
    scroll-behavior: smooth;
    -webkit-user-select: none;
/* Safari */
    -ms-user-select: none; /*
IE 10 and IE 11 */
    user-select: none; /*
Standard syntax */
}

.card-slider-item {
    width: calc(100% / 5);
    display: inline-flex;
    position: relative;
    align-items: start;
    justify-content: start;
    flex-direction: column;
    /* border: 1px solid
rgb(101 163 13); */
    background-color: white;

    box-shadow: 1px 1px 5px
rgb(102, 100, 100);
    padding-bottom: 0px;
    cursor: pointer;
    border-radius: 5px;
    overflow: hidden;
    -webkit-user-select: none;
/* Safari */
    -ms-user-select: none; /*
IE 10 and IE 11 */
    user-select: none; /*
Standard syntax */
}

.card-slider-item:first-child
{
    margin-left: 0px;
}

.card-slider-item > img {
    height: 200px;
    width: 100%;
    object-fit: cover;
    aspect-ratio: 3/2;
}

.dragging {
    scroll-behavior: auto;
    cursor: grab;
    scroll-snap-type: none;
}

div:has(>.card-slider)
:is(.next, .prev){
    position: absolute;
    background-color: white;
    font-weight: 700;
    color: black;
    height: 2rem;
    width: 2rem;
    border-radius: 100%;
    display: flex;
    align-items: center;
    justify-content: center;
    border: 0;
    transition: .2s ease-in;
    cursor: pointer;
    font-size: 1.1rem;
}
div:has(>.card-slider) .prev{
    left: -15px;
    box-shadow: -2px 2px 5px
rgb(197, 188, 188);
    padding-right: 3px;
    display: none;
}

```

```

}
div:has(>.card-slider) .next{
    right: -15px;
    box-shadow: 2px 2px 5px
    rgb(197, 188, 188);
}

div:has(>.card-slider)
:is(.prev, .next):hover {
    transform: scale(1.5);
}

@media screen and (max-width:
780px) {
    .card-slider-item{
        width: calc(100% / 2);
    }
    div:has(>.card-slider)
:is(.next, .prev){
        display: none;
    }
    .card-slider-item > img {
        height:150px;
        width: 100%;
        object-fit: cover;
        aspect-ratio: 3/2;
    }
    div:has(>.card-slider)
:is(.prev, .next):hover {
        transform: scale(1.2);
    }
}

/* end slider card */

/* Chrome, Safari, Edge, Opera
*/
input::-webkit-outer-spin-
button,
input::-webkit-inner-spin-
button {
    -webkit-appearance: none;
    margin: 0;
}

/* Firefox */
input[type=number] {
    -moz-appearance: textfield;
}

/* .alert{
    animation: shake 600ms;
} */

@keyframes shake {
    /* 0% {

```

```

        transform: scale(0.9);
    }
    25% {
        transform: scale(1);
    }
    50% {
        transform: scale(0.9);
    }
    100%{
        transform: scale(1);
    } */
    from {
        grid-template-rows: 0fr;
    }
    to {
        grid-template-columns:
1fr;
    }
}

/* label[for="notif"]{
    background: white;
    position: absolute;
    height: 1rem;
    width: 1rem;
    left: 5px;
    border-radius:9999px ;
} */

/* #bar::after {
    @apply absolute content-
[""] w-[calc(38/60*100%)]
left-0 bg-indigo-500 top-0 h-2
} */

.progress-seat::after{
    /* --width :calc(60/60); */
    position: absolute;
    content: "";
    width: var(--progress);
    top: 0;
    left: 0;
    height:0.5rem;
    text-align: center;
}

<?php

namespace App\Controllers;

use App\Helpers\Auth;
use App\Helpers\Controllers;
use App\Models\Customer;
use
App\Models\CustomerAddress;

class CustomerController
extends Controllers

```

```

{
    public function
__construct()
    {
        if
(is_null(Auth::customer()))
return $this->redirect("/");
    }
    public function index()
    {
        $customer =
Customer::findWithAddress(Auth
::customer()->id);
        $this->view("customer.index",
compact("customer"));
    }

    // public function
update($request)
    // {
    //
Customer::update(Auth::custome
r()->id, (array)$request);
    //     $customer =
Customer::find(Auth::customer(
)->id);
    //
Auth::login($customer,
"customer");
    //     $this->redirect("/profil", "profil
berhasil diubah");
    // }

    public function address()
    {
        $address =
CustomerAddress::where("custom
er_id", Auth::customer()->id)-
>get();
        $this->view("customer.address",
compact("address"));
    }
}

<?php

namespace App\Controllers;

use App\Helpers\Auth;
use App\Helpers\Controllers;
use App\Helpers\Database;
use App\Helpers\Request;
use App\Models\Bus;

```

```

use App\Models\BusType;
use App\Models\Customer;
use App\Models\Jadwal;
use App\Models\Menu;
use App\Models\Orders;
use App\Models\Province;
use App\Models\Regency;
use App\Models\Travel;
use App\Models\Tujuan;
use App\Models\Ukm;
use App\Models\Users;

class HomeController extends
Controllers
{
    public function index()
    {
        $this->view("home.index");
    }

    public function location()
    {
        $q = Request::query()-
>q ?? "";
        if ($q == "") {
            http_response_code(400);
            return;
        }
        $q = "%$q%";
        $locations =
Regency::where("name", $q,
"like")->get();

        // $this->dd($locations);
        foreach ($locations as
$regency) {
            $province =
Province::find($regency->
province_id);
            echo "<li
province_id='$regency->
province_id'
regency_id='$regency->id'
class='option-item'>" .
strtolower($regency->name) .
", " . strtolower($province->
name) . " </li>";
        }

        $locations =
Province::where("name", $q,
"like")->get();
    }
}

```

```

        foreach ($locations as
$province) {
            $regencies =
Regency::where("province_id",
$province->id)->where("not
name", $q, "like")->get();
            foreach
($regencies as $regency) {
                echo "<li
province_id='$regency-
>province_id'
regency_id='$regency->id'
class='option-item'>" .
strtolower($regency->name) .
", " . strtolower($province-
>name) . " </li>";
            }
        }
        // $this-
>view("home.location");
    }

```

```

        public function search()
        {

extract(get_object_vars(Reques
t::query()));
            $day = $this-
>formatDayToInd(date("w",
strtotime($date)));
            $hometown = explode(",
", $hometown);
            $destination_city =
explode(", ",
$destination_city);

            $home_regency =
Regency::where("name",
strtoupper($hometown[0]))-
>first();
            $home_province =
Province::where("name",
strtoupper($hometown[1]))-
>first();
            $destination_regency =
Regency::where("name",
strtoupper($destination_city[0
]))->first();
            $destination_province
= Province::where("name",
strtoupper($destination_city[1
]))->first();

            $destinations =
Tujuan::query("SELECT

```

```

tujuan.*, jadwal.hari,
jadwal.id as jadwal_id,
group_concat(jadwal.jam) as
jam, group_concat(jadwal.trip)
as trip FROM tujuan INNER JOIN
jadwal ON
tujuan.id=jadwal.tujuan_id
WHERE (

province_from_id=:home_provinc
e

        AND

tujuan.regency_from_id=:home_r
egency

        AND

province_to_id=:destination_pr
ovince

        AND

tujuan.regency_to_id=:destinat
ion_regency AND
jadwal.hari=:day AND
jadwal.trip='berangkat' AND
jadwal.archived=:archived)

        OR

(province_from_id=:destination
_province

        AND

tujuan.regency_from_id=:destin
ation_regency

        AND

province_to_id=:home_province

        AND

tujuan.regency_to_id=:home_reg
ency AND
pulang_pergi=:pulang_pergi AND
jadwal.hari=:day AND
jadwal.trip='pulang' AND
jadwal.archived=:archived)

        group
by(tujuan.id

        )->bind(
        [

"home_province" =>
$home_province->id,
                "home_regency"
=> $home_regency->id,

"destination_regency" =>
$destination_regency->id,

"destination_province" =>
$destination_province->id,
                "pulang_pergi"
=> true,

```

```

        "day" => $day,
"archived" => false
    ]
    )->get();

    $destinations =
array_map(
    function
($destination) use
(&$hometown,
&$destination_city) {
        $destination->
>hometown = implode(" ",
$hometown);
        $destination->
>destination_city = implode(" ",
$destination_city);
        $destination->
>bus = Bus::where("tujuan_id",
$destination->id)->first();

        $type =
BusType::find($destination->
>bus->type_id);
        $destination->
>bus->type = $type;

        $destination->
>travel =
Travel::find($destination->
>bus->travel_id);
        $fasilitas_id
= array_filter(explode(" ",
$destination->bus->
>fasilitas_id), "intval");
        $destination->
>test =
            date("Y-m-
d H:i:s",
strtotime(Request::query()-
>date . $destination->jam));
        foreach
($fasilitas_id as $id) {

            $destination->fasilitas[] =
Database::find($id,
"fasilitas_bus");
        }

        $destination->
>seats = Bus::seats(bus:
$destination->bus,
schedule_id: $destination->
>jadwal_id, date:
Request::query("date"));

```

```

        return
$destination;
    },
    array_filter(
        $destinations,
        fn
($destination) => date("Y-m-d
H:i:s", (strtotime($date .
$destination->jam))) >=
date("Y-m-d H:i:s",
(strtotime("now"))))
    );
    // filter tipe bus
    if (isset($tipe_bus)
&& $tipe_bus != "") {
        $tipe_bus =
array_values(array_filter(expl
ode(" ", $tipe_bus), fn ($arr)
=> $arr != ""));
        $destinations =
array_filter($destinations, fn
($destination) =>
in_array($destination->bus->
>type->tipe, $tipe_bus));
    }

    if (isset($fasilitas)
&& $fasilitas != "") {
        $fasilitas_name =
array_values(array_filter(expl
ode(" ", $fasilitas), fn
($arr) => $arr != ""));
        $destinations =
array_filter($destinations,
function ($destination) use
(&$fasilitas_name) {
            if
(isset($destination->
>fasilitas))
                foreach
($destination->fasilitas as
$fasilitas) {
                    if
(in_array($fasilitas->name,
$fasilitas_name)) return
$destination;
                }
            });
    }

    // filter jam
    berangkat
    if
(isset($jam_berangkat) &&
$jam_berangkat != "") {

```

```

        $jam_berangkat =
array_values(array_filter(explode(" ", $jam_berangkat), fn
($arr) => $arr != ""));

```

```

        $destinations =
array_filter($destinations,
function ($destination) use
(&$jam_berangkat, &$date) {
    foreach
($jam_berangkat as $jam) {
        [$start,
$end] = explode(" - ", $jam);
        $start =
date("Y-m-d H:i:s",
strtotime("$date.$start"));
        $end =
date("Y-m-d H:i:s",
strtotime("$date.$end"));

```

```

$destination_jam = date("Y-m-d
H:i:s", (strtotime($date .
$destination->jam)));
        if ($start
<= $destination_jam && $end >=
$destination_jam) return
$destination;
    }
});
}

```

```

        $hometown =
$hometown[0];
        $destination_city =
$destination_city[0];
        $filters = [
            "jam_berangkat" =>
[
                "06:00 -
11:59", "12:00 - 17:59",
"18:00 - 23:59", "00:00 -
05:59"
            ],
            "tipe_bus" =>
BusType::all(),
            "fasilitas" =>
Database::all(table:
"fasilitas_bus"),
        ];

```

```

        $date = $this-
>formatDate((strtotime($date))
);
        if (
isset($_SERVER["CONTENT_TYPE"]

```

```

) && $_SERVER["CONTENT_TYPE"]
=== "application/json"
) {
    $this-
>single("home.content",
compact("destinations",
"hometown",
"destination_city", "filters",
"date"));
    return;
}

```

```

        $this-
>view("home.search",
compact("destinations",
"hometown",
"destination_city", "filters",
"date"));
    }

```

```

        public function profil()
        {
            $user =
Customer::find(Auth::customer(
)->id);
            // $this->dd($user);
            $this-
>view("home.profil",
compact("user"));
        }

```

```

        public function
profil_update($request, $id)
        {
            try {
                Customer::update($id,
(array)$request);
                $this-
>redirect("/profil", [
                    "message" =>
"Porilf berubah"
                ]);
            } catch (\Throwable
$th) {
                //throw $th;
            }
        }
    }
}

```

```

<?php
namespace App\Controllers;

use App\Helpers\Auth;
use App\Helpers\Controllers;

```

```

use App\Helpers\Database;
use App\Helpers\Request;
use App\Models\Notification;
use App\Models\Orders;
use App\Models\Subscription;

class NotificationController
extends Controllers
{
    // public function
    __construct()
    // {
    //     if
    (is_null(Auth::customer()))
    return $this->redirect("/");
    public function
    add_time($notif)
    {
        $strtotime =
        strtotime($notif->updated_at);
        $notif->time =
        $this->formatDate($strtotime)
        . date("H:i:s", $strtotime);
        return $notif;
    }
    // }
    public function index()
    {
        $this->
        >view("notification.index");
    }

    public function content()
    {
        $notifications =
        array_map(array:
        Notification::where("customer_
        id", Auth::customer()->id)-
        >orderBy("be_read, updated_at
        desc")->get(), callback:
        array($this, "add_time"));
        $this->
        >single("notification.content"
        , compact("notifications"));
    }

    public function
    subscription($request)
    {
        $content = $request->
        >content;
        $is_customer =
        boolval($request->
        >is_customer);

```

```

        $user_id = ($request->
        >is_customer ?
        Auth::customer()->id :
        Auth::user()->id);
        if
        (!Subscription::where("content
        ", $content)->first())

        Database::insert(table:
        "subscription", columns: [
        "user_id" =>
        $user_id,
        "content" =>
        $content,
        "is_customer"
        => $is_customer
        ]);
        return;
    }

    public function show()
    {
        extract((array)Request::query(
        ));
        $link =
        "/notifikasi/detail?order_id="
        . $order_id;
        $notif =
        Notification::where("link",
        $link)->first();
        if (!$notif->be_read)
        {
            Notification::update(compact("
            link"), ["be_read" => true]);
        }

        if (isset($read) &&
        $read) return $this->
        >redirect("/notifikasi");

        $order = $this->
        >order_detail(Orders::find($or
        der_id));
        $status = $this->
        >message($order->status);

        $strtotime =
        strtotime($notif->updated_at);
        $notif->time = $this->
        >formatDate($strtotime) .
        date("H:i:s", $strtotime);

        $this->
        >view("notification.show",

```

```

compact("order", "notif",
"status"));
    }
}

<?php

namespace App\Controllers;

use App\Helpers\Auth;
use App\Helpers\Controllers;
use App\Helpers\Database;
use App\Helpers\Request;
use App\Models\Bus;
use App\Models\BusType;
use App\Models\Cart;
use App\Models\Jadwal;
use App\Models\Menu;
use App\Models\Notification;
use App\Models\OrderItems;
use App\Models\Orders;
use App\Models\Travel;
use App\Models\Tujuan;
use App\Models\Ukm;

class OrderController extends
Controllers
{
    public function
__construct()
    {
        if
(is_null(Auth::customer()))
return $this->redirect("/");
    }
    public function index()
    {
        // $this->dd($orders);
        $orders = [];
        $this->
view("order.index",
compact("orders"));
    }

    public function content()
    {
        $status =
Request::query()->q ??
"waiting";
        $date =
Request::query()->date ?? "";
        // $this->dd("okle");
        if ($status !=
"waiting") {

```

```

        $orders =
Orders::where("customer_id",
Auth::customer()->id)-
>where("status", "$status")-
>orderBy("updated_at desc")-
>get();
        } else {
            $orders =
Orders::where("customer_id",
Auth::customer()->id)-
>whereGroup("status",
"$status")-
>endWhereGroup("status",
"otw")->orderBy("status,
updated_at desc")->get();
        }
        // $this->dd($orders);
        $orders =
array_filter(array: $orders,
callback: function ($order)
use (&$date) {
            if
(!str_contains($order-
>tanggal, $date))
                return false;

            $order->bus =
Bus::_first($order->bus_id);
            $order->bus->type
= BusType::find($order->bus-
>type_id);
            $order->bus-
>fasilitas =
array_map(function ($id) {
                return
Database::find($id,
"fasilitas_bus");
            },
array_filter(explode(",",
$order->bus->fasilitas_id),
"intval"));
            $order->schedule =
Jadwal::find($order-
>jadwal_id);
            $order->bus-
>tujuan =
Tujuan::_first($order->bus-
>tujuan_id, $order->schedule-
>trip);
            $order->travel
= Travel::find($order->bus-
>travel_id);
            $order->date
= $this->

```

```

>formatDate(strtotime:
strtotime($order->tanggal));
    $order->time
    = $this-
>formatDate(strtotime:
strtotime($order->updated_at))
. date("H:i:s",
strtotime($order-
>updated_at));
    $order->message
    = $this->message($order-
>status);
    $order->color
    = $this->color($order-
>status);
    return $order;
    });
    $this-
>single("order.content",
compact("orders"));
    }

```

```

    public function create()
    {
        extract($_GET);
        $schedule =
Jadwal::find($schedule_id);
        $destination_id =
$schedule->tujuan_id;
        $destination =
Tujuan::_first($destination_id
);
        $bus =
Bus::where("tujuan_id",
$destination_id)->first();
        $bus->type =
BusType::find($bus->type_id);
        $bus->travel =
Travel::find($bus->travel_id);
        $strtotime =
strtotime($date . $schedule-
>jam);

        $bus->seats =
Bus::seats(bus: $bus,
schedule_id: $schedule_id,
date: $date);

        $departure = $this-
>formatDate($strtotime);
        // $this-
>dd($departure);
        // $this->dd([
        //     $schedule,
        $destination, $bus
        // ]);
    }

```

```

    $this-
>view("order.create",
compact("schedule",
"destination", "bus",
"hometown",
"destination_city",
"departure"));
    }

    public function
store($request)
    {
        $request->customer_id
= Auth::customer()->id;
        $request->total
= Bus::find(request-
>query("bus_id"))->harga *
$request->kursi;
        $request->jam
= Jadwal::find(request-
>query("jadwal_id"))->jam;

        $request =
(object)[...(array)$request,
...(array)request->query()];
        Orders::insert(
Orders::data($request)
);
        $this->redirect("/", [
            "message" =>
"Order berhasil, mohon
menunggu admin untuk
konfirmasi order"
        ]);
    }

    public function
update_address($request)
    {
        try {
            $id = request-
>query("id");

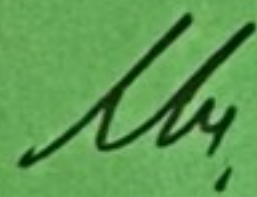
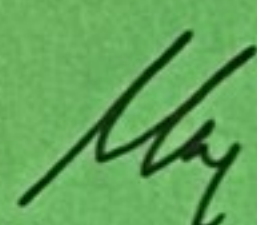
            Orders::update($id, [
                "alamat" =>
$request->alamat
            ]);
            echo json_encode([
                "message" =>
"berhasil mengubah alamat",
            ]);
        } catch (\Throwable
$th) {
            echo json_encode([

```


KARTU MONITORING BIMBINGAN
 MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
 FAKULTAS TEKNIK
 UNIVERSITAS MUHAMMADIYAH PAREPARE

SKRIPSI

Siswa : HENRIANI	Pembimbing I : Ade Hastuty, S.T., S.Kom., M.T
: 219280081	Pembimbing II : Marlina, S.Kom., M.Kom.
Skripsi : E - BOOKING KENDARAAN BUS PARIWISATA BERBASIS WEB	

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
si 1 Jurnal	24/7/2023 	Konsultasi 1 Tambahkan Desain Sistem sesuai dg	20/7/2023 
si 2 jika berpikirnya	27/7/2023 	Konsultasi 2 Cari penelitian Terkait yg ada di Umpare.	21/7/2023 
si 3 in Artemnya knpa	28/7/2023 	Konsultasi 3 Sesuaikan dg Isi peb Penelitian Terkait.	21/7/2023 
si 4		Konsultasi 4 ACC Ujian proposal	29/7/2023 
si 5		Konsultasi 5	

Lanjut ke halaman sebelah...

1:
 siswa wajib konsultasi minimal 5 kali
 ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
 ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
 ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik

KARTU MONITORING BIMBINGAN

MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

HASIL

: HENRIANI	Pembimbing I : Ade Hastuty, ST., S.Kom., MT.
: 219280081	Pembimbing II : Marlina, S.Kom., M.Kom.
KENDARAAN BUS PARIWISATA BERBASIS WEB	

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
bakan Uitur tarif gambar ab 1 sampai 5.		Konsultasi 1 - Semua Bahasa Asing dicetak Miring - Sematkan Daftar pustaka di Isi Skripsinya	
track dan flowchart		Konsultasi 2	
car Seminar hasil 29/07/2024		Konsultasi 3	
4		Konsultasi 4 Acc Ujian Hasil	
5		Konsultasi 5	

Lanjut ke halaman sebelah...

an :

siswa wajib konsultasi minimal 5 kali
itu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
itu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
itu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik

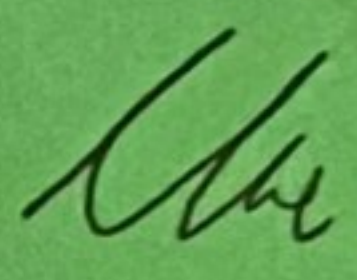
SKRIPSI

Nama : HENRIANI
NPM : 219280081

Pembimbing I : Ade Hastuty, ST., S.Kom., MT.

Pembimbing II : Marlina, S.Kom., M.Kom.

JUDUL : RENCANA KENDARAAN BUS PARIWISATA BERBASIS WEB

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
1 dan coding di lampiran.		Konsultasi 1 Tambahkan di kesimpulan Hasil dari Pengujian Aplikasi	
2 hard disetorkan sebelum diak Program.		Konsultasi 2 	
3 pulan menjawab rumusan an.		Konsultasi 3 Acc Ujian Tutup	19/8 2024. 
4 benarkan sesuai aplikasi rangian aplikasi		Konsultasi 4	
5 for ujian skripsi		Konsultasi 5	

Lanjut ke halaman sebelah...

ian :
Mahasiswa wajib konsultasi minimal 5 kali
itu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
itu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
itu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik



Nomor : 443/PRK/LB-TI/II.3.AU/D/2023

diberikan kepada :

Henriani

NIM. 219280081

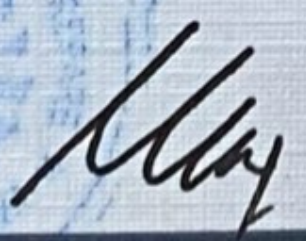
telah menyelesaikan

PRAKTIKUM V

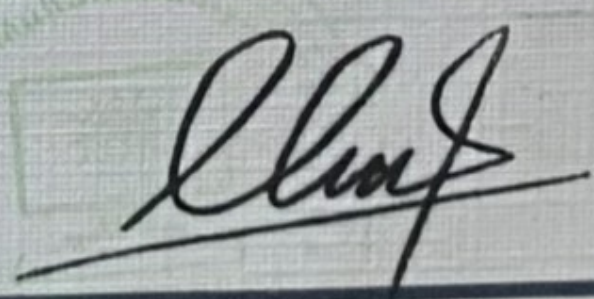
Laboratorium Teknik Informatika
Program Studi Teknik Informatika Fakultas Teknik
Universitas Muhammadiyah Parepare

Pada tanggal 16 Januari 2023 - 29 Maret 2023
Parepare, 4/12/2023

**KETUA PRODI
TEKNIK INFORMATIKA**


MARLINA. S.Kom., M.Kom

**KEPALA
LABORATORIUM**


Ir. UNTUNG SUWARDYO, S.Kom., MT., IPP.