PROJECT 1

SISTEM INFORMASI LOUNDRY

Dibuat untuk memenuhi tugas Pemrograman Web Dasar

**Dosen : Leila Fajrie Auddiena N. F. A., M.Pd.**



Disusun oleh :

Miftakhuddin Falaki

(201843501551)

Informatika

Fakultas Teknik dan Ilmu Komputer

Universitas Indraprasta PGRI

2020

* Struktur

laundry

├─ assets

│  ├─ css

│  │  ├─ laundry.css

│  │  └─ style.css

│  ├─ img

│  │  ├─ banner2.jpg

│  │  └─ img\_avatar2.png

│  └─ js

│     └─ js.js

├─ menu

│  ├─ daftar.html

│  ├─ home.html

│  └─ laundry.html

├─ proses

│  ├─ proses.html

│  └─ trimakasih.html

├─ index.html

└─ readme.md

1. index.html

<html>

<head>

    <title>SISTEM INFORMASI LOUNDRY</title>

</head>

<frameset>

    <frame src="menu/home.html"></frame>

</frameset>

</html>

Disini saya menggunakan frameset untuk menampilkan menu/home.html

1. home.html
2. Sytax home.html
   * head

<head>

    <title>SISTEM INFORMASI LOUNDRY</title>

    <link rel="stylesheet" type="text/css" href="../assets/css/style.css">

</head>

Link ke main css.

* + Header

<header>

        <nav class="menu">

            <ul>

                <li>

                    <h1 style="padding-left: 25px;">SISTEM INFORMASI LOUNDRY</h1>

                </li>

                <li>

                    <a href="./home.html" style="margin-left: 480px;">Home</a>

                </li>

                <li>

                    <a href="../menu/laundry.html">Jasa Laundry</a>

                </li>

                <li>

                    <a href="../menu/daftar.html">Daftar/Masuk</a>

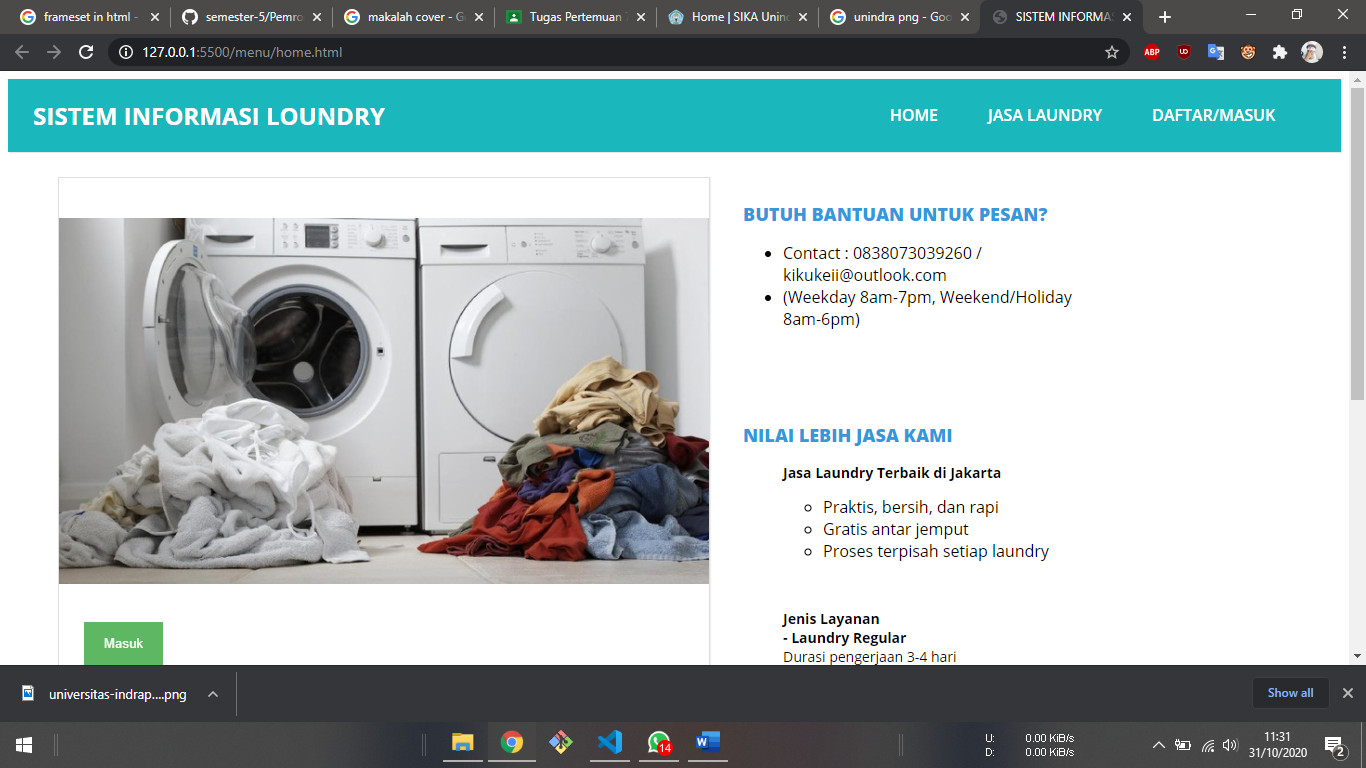
                </li>

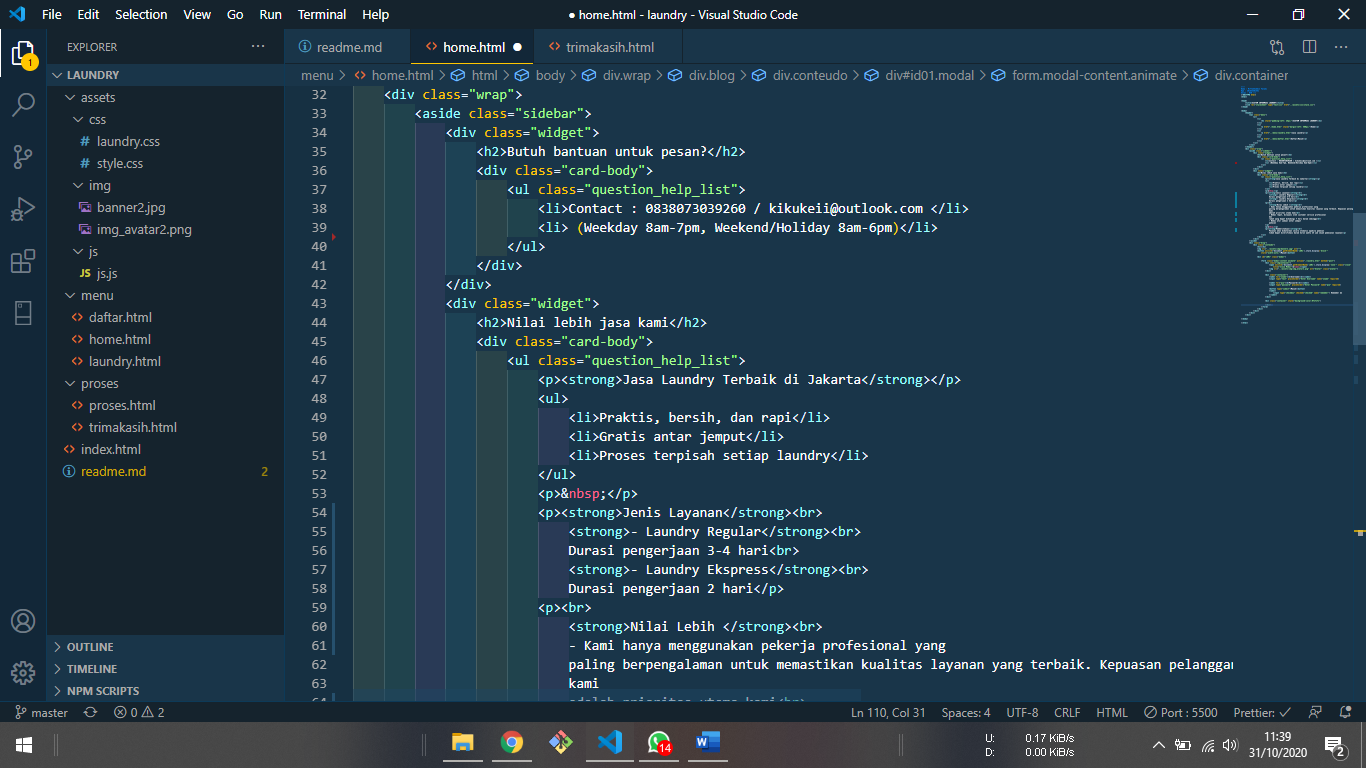
            </ul>

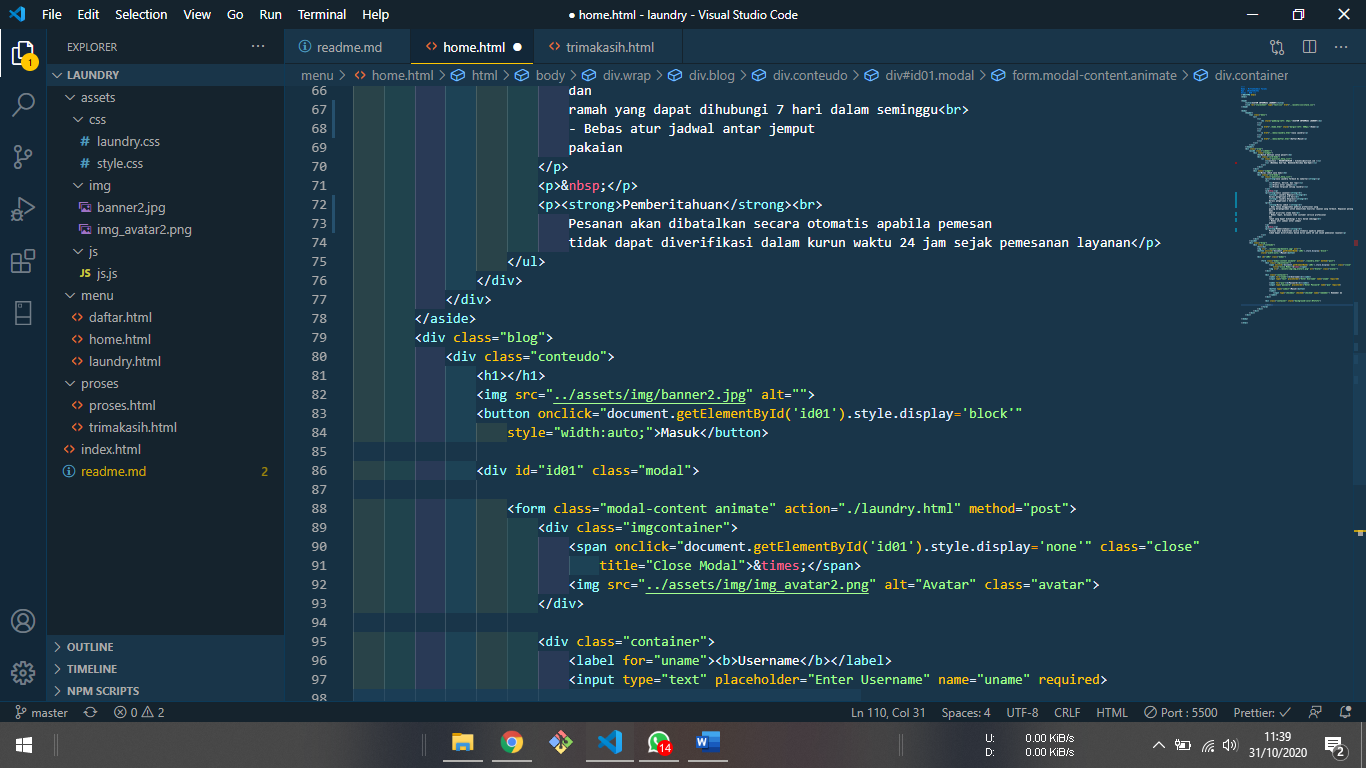
        </nav>

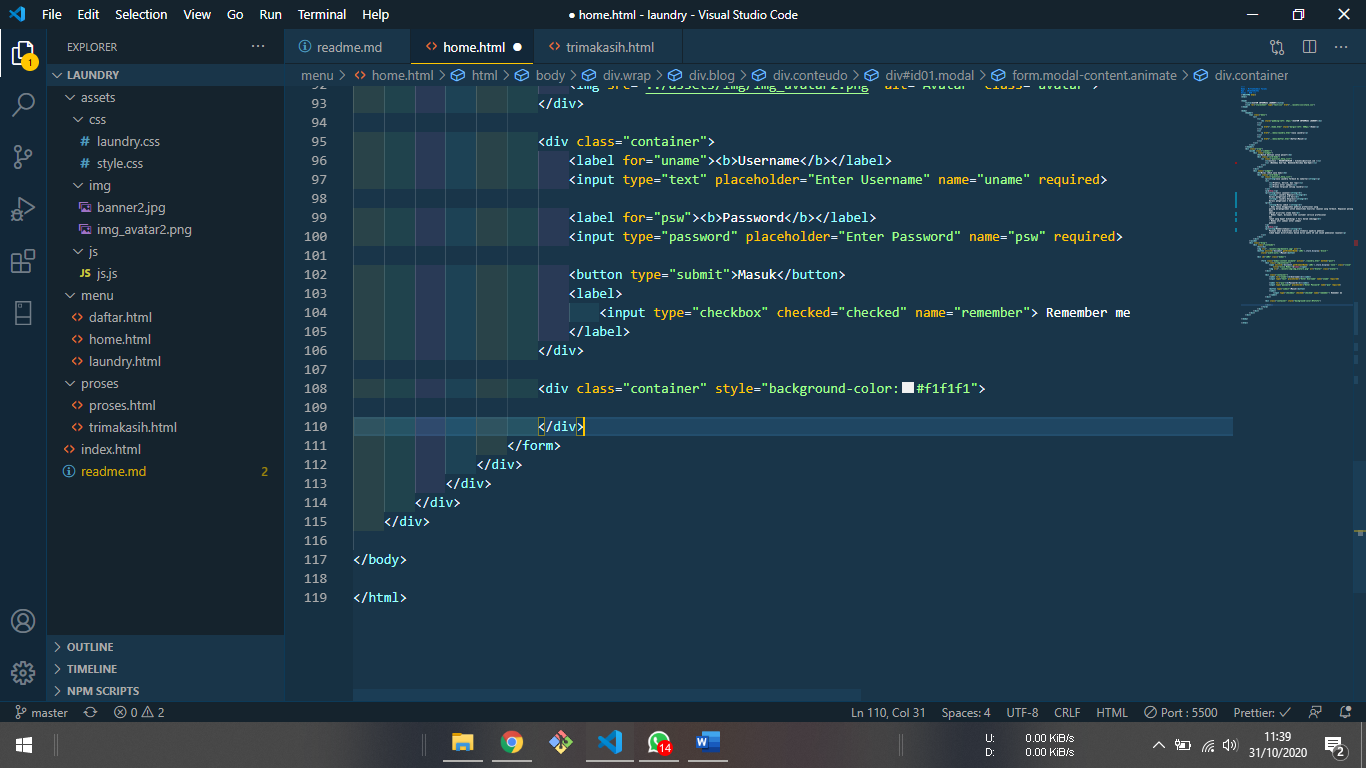
    </header>

Di Tag header saya buat navbar dengan menampilkan Judul, Link(Heme, Jasa Laundry, Daftar/masuk)

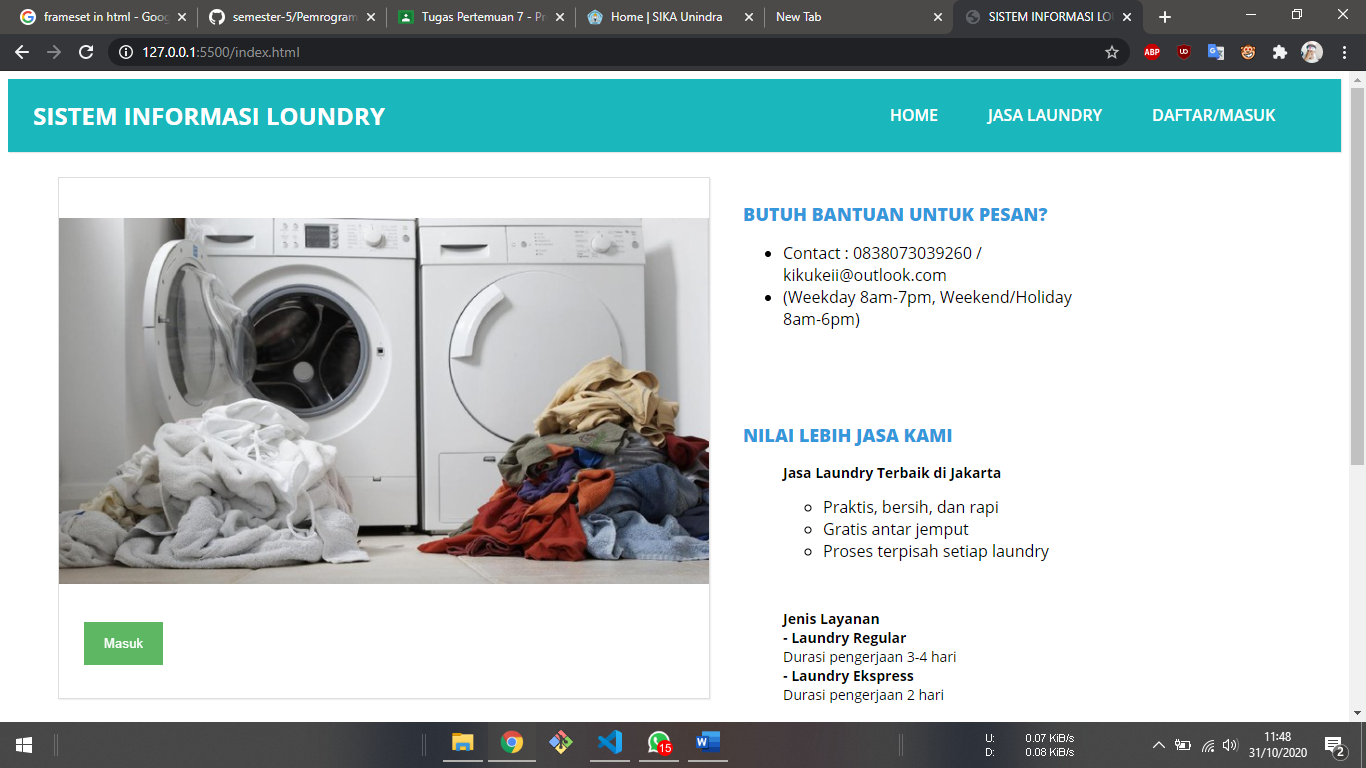


* + Warper

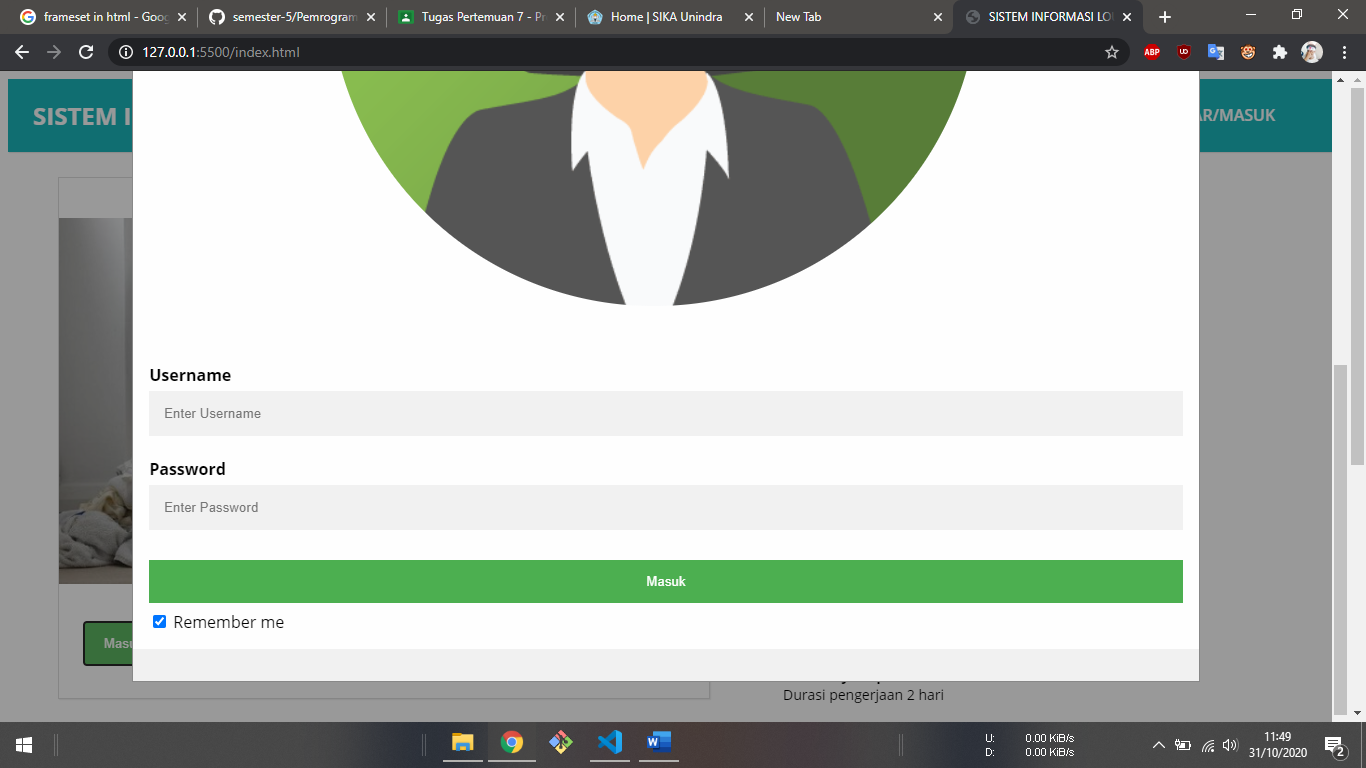




1. Output home.html



Saat tekan button masuk maka akan tampil jendela pop-up login.



Lalu saat masuk akan diarahkan ke laundry.html

1. Laundry.html
2. Sytax laundry.html
   * Head

<head>

    <title>Laundry</title>

    <link rel="stylesheet" type="text/css" href="../assets/css/style.css">

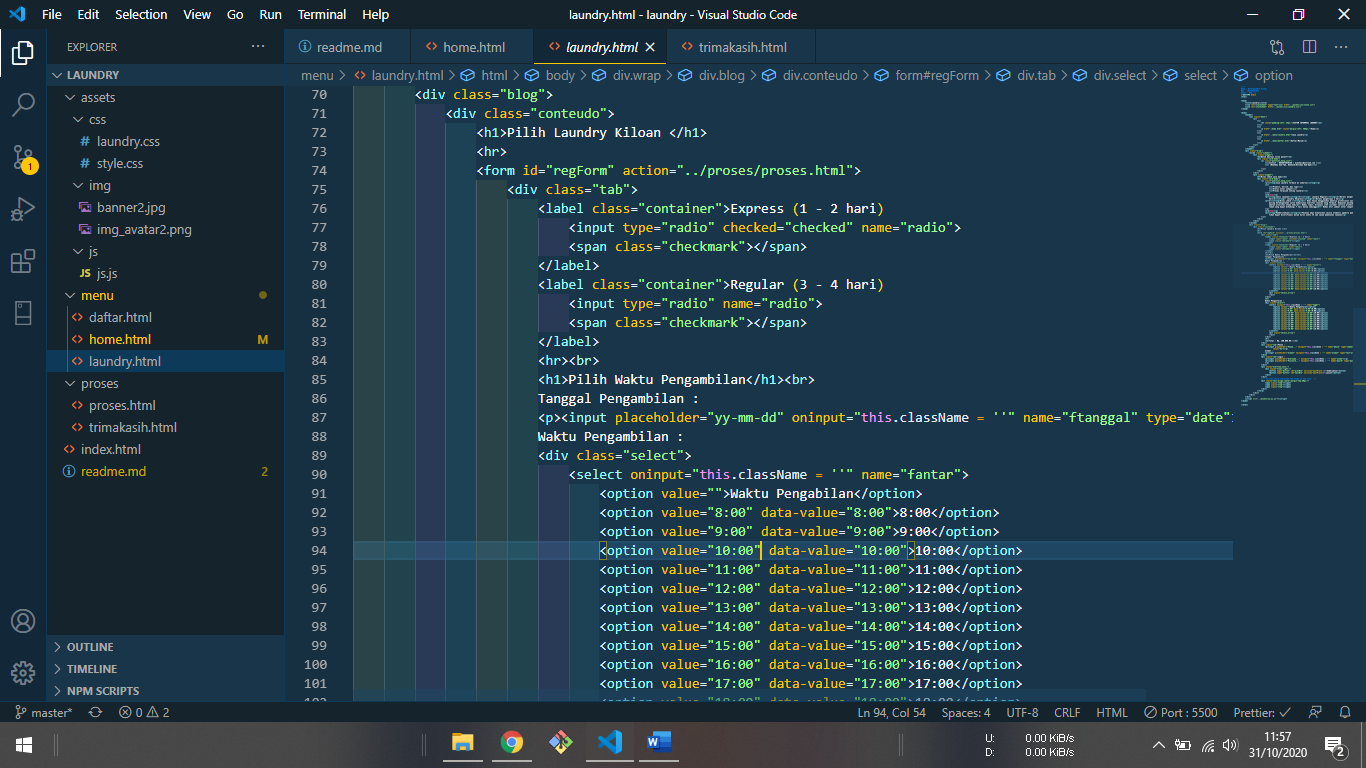
    <link rel="stylesheet" href="../assets/css/laundry.css">

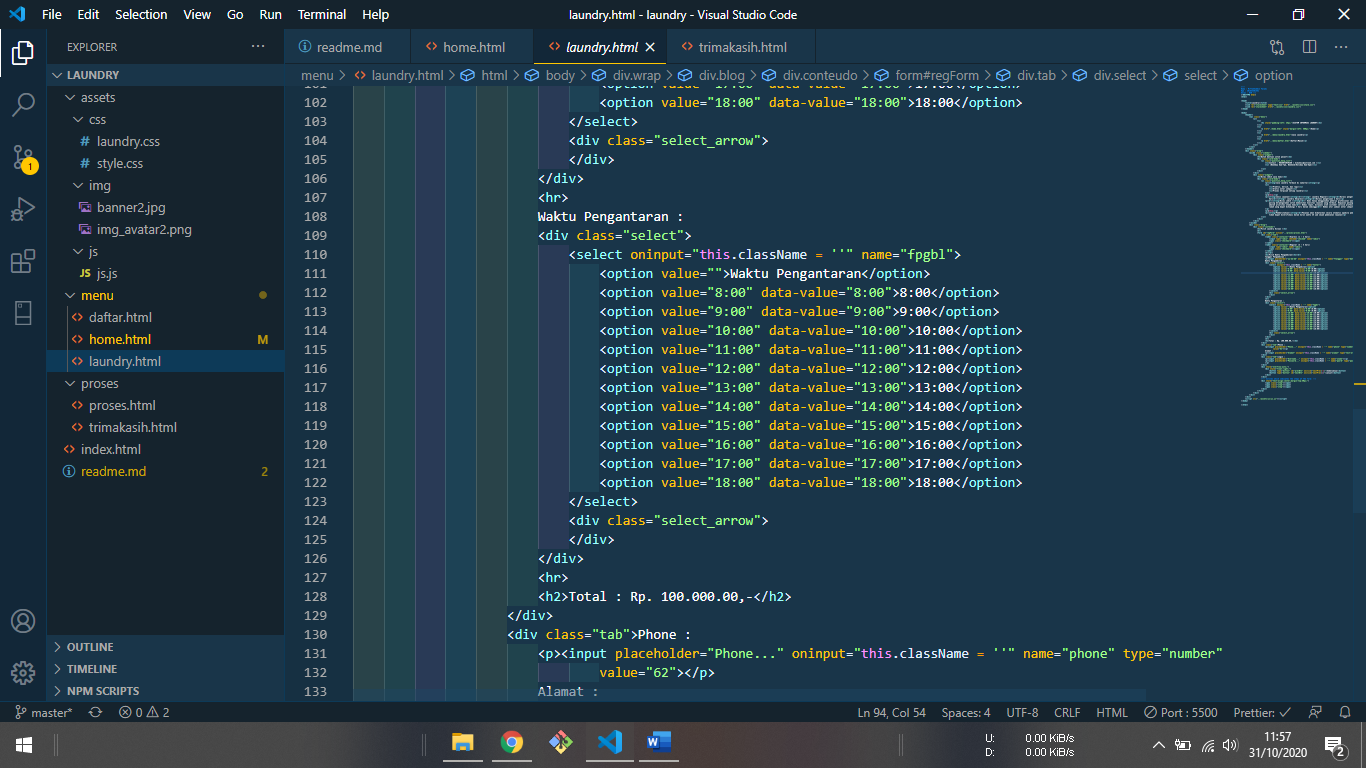
</head>

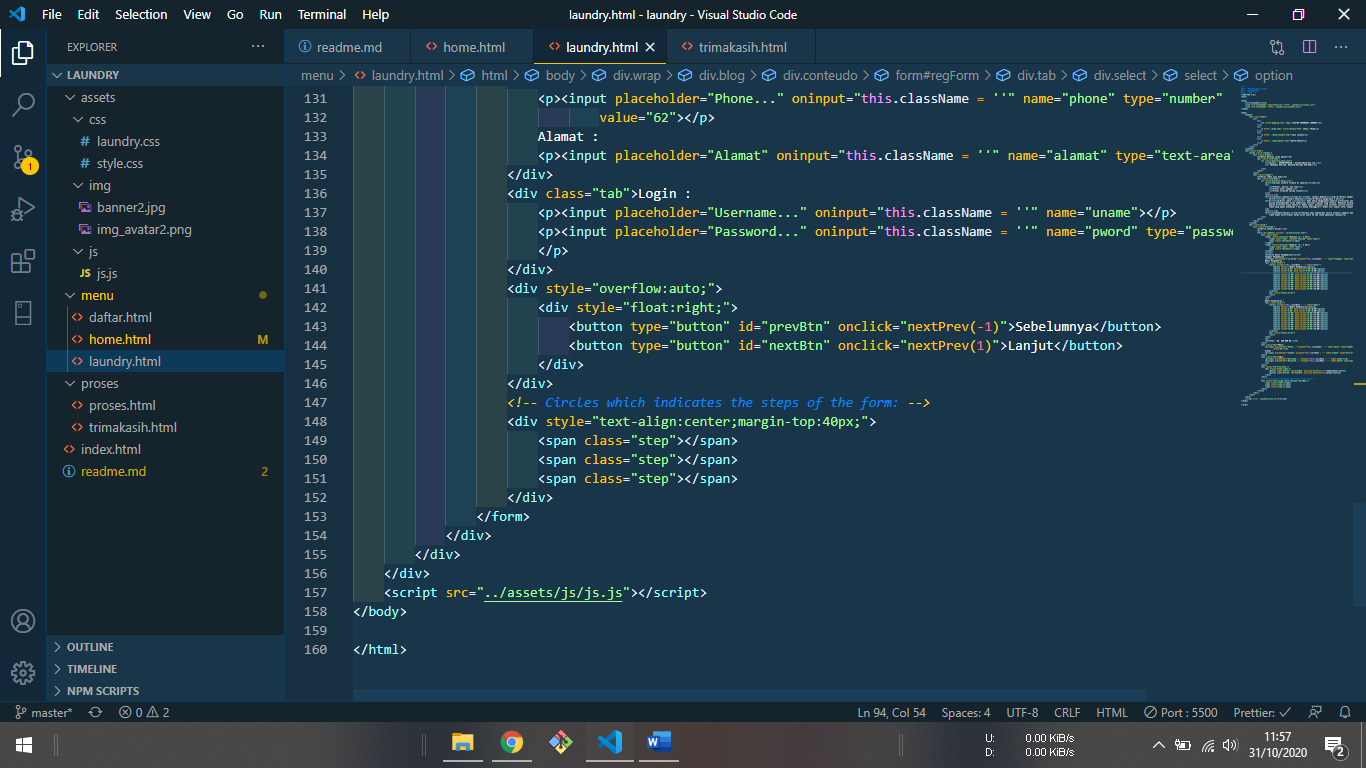
* + Header dan aside

Sama dengan home.html

Warper > blog



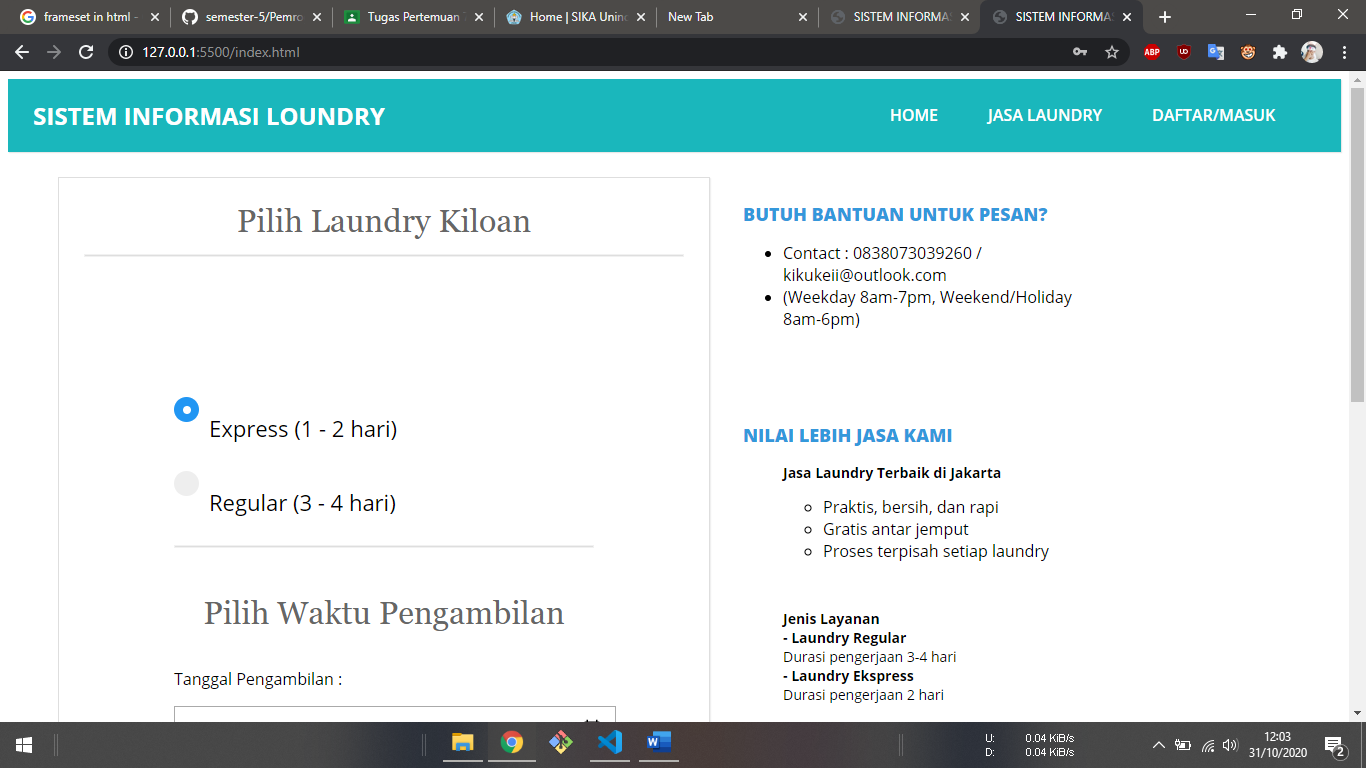


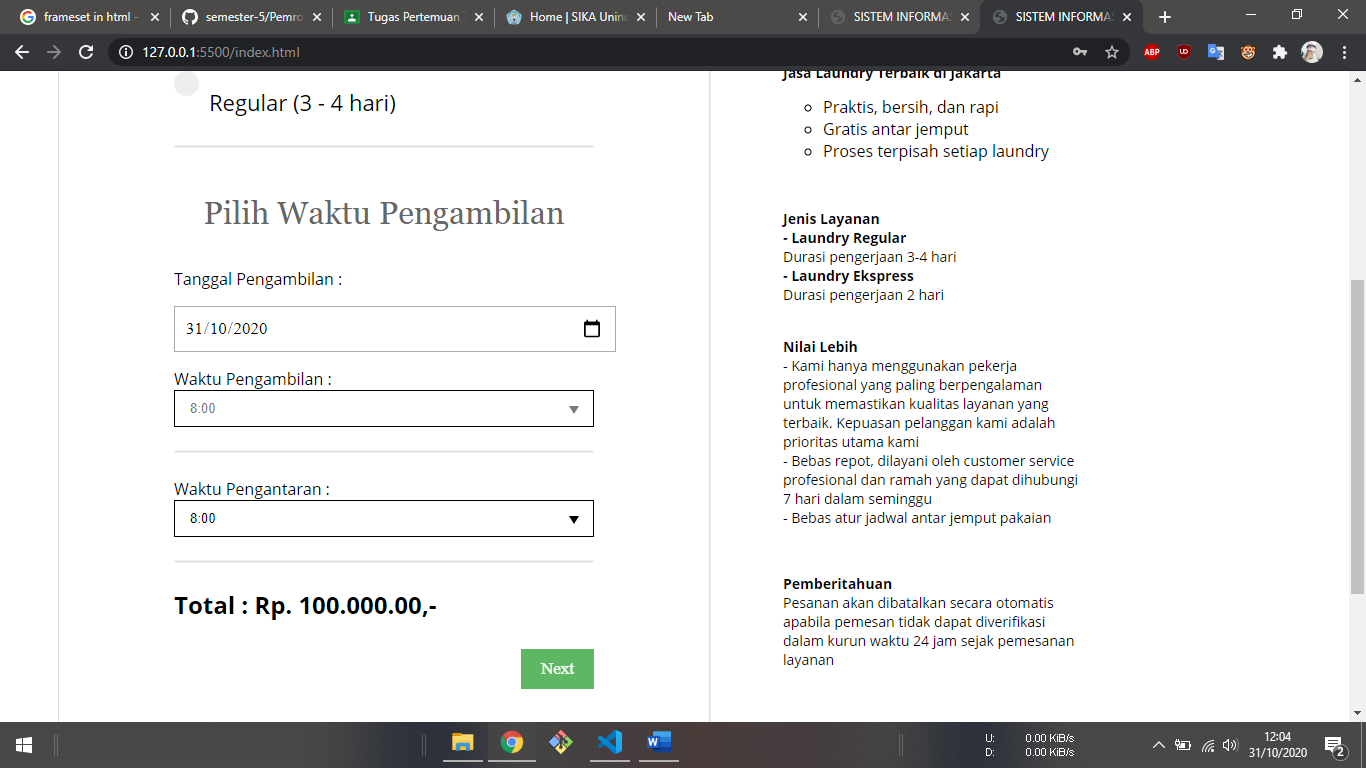


Form laundy.html dibuat dengan tab step lalu diarahkan ke proses.html(action)

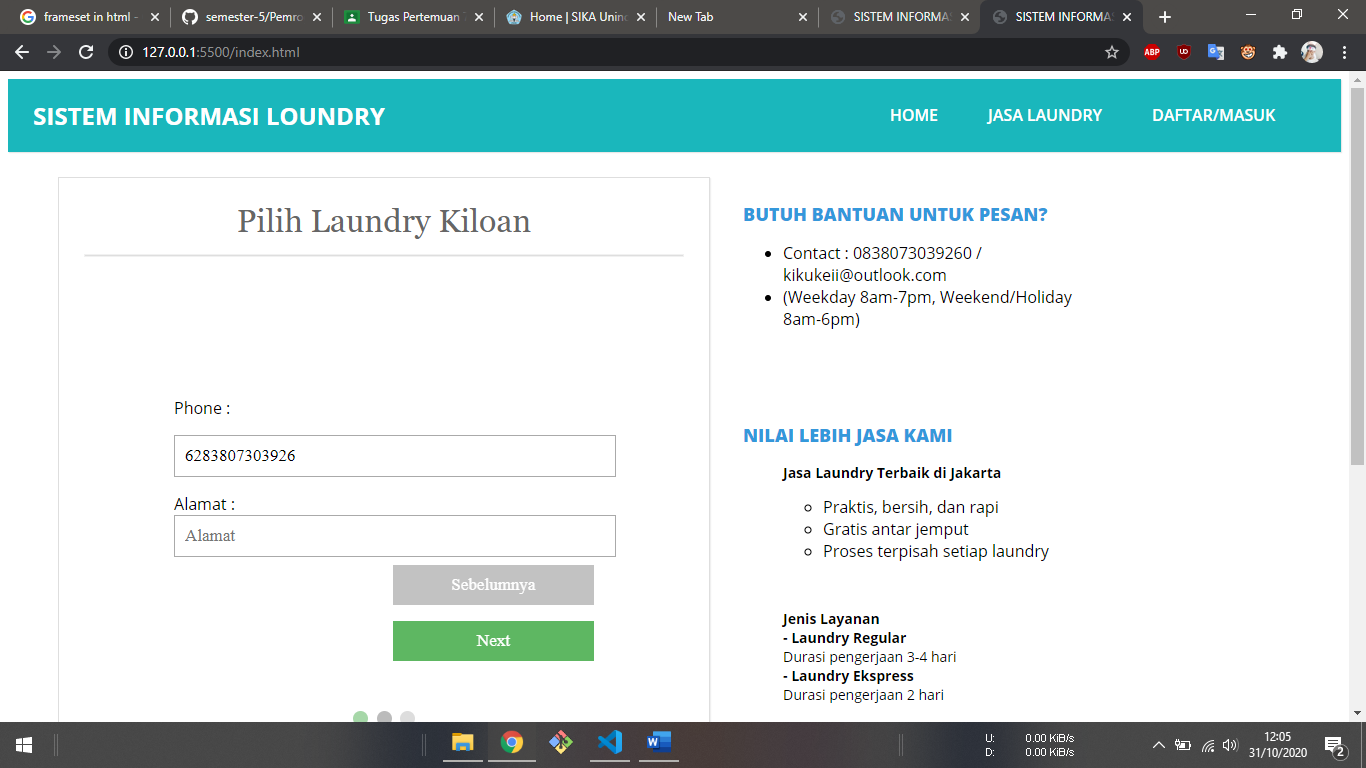
1. Output laundry html

Tpe dan waktu penambilan

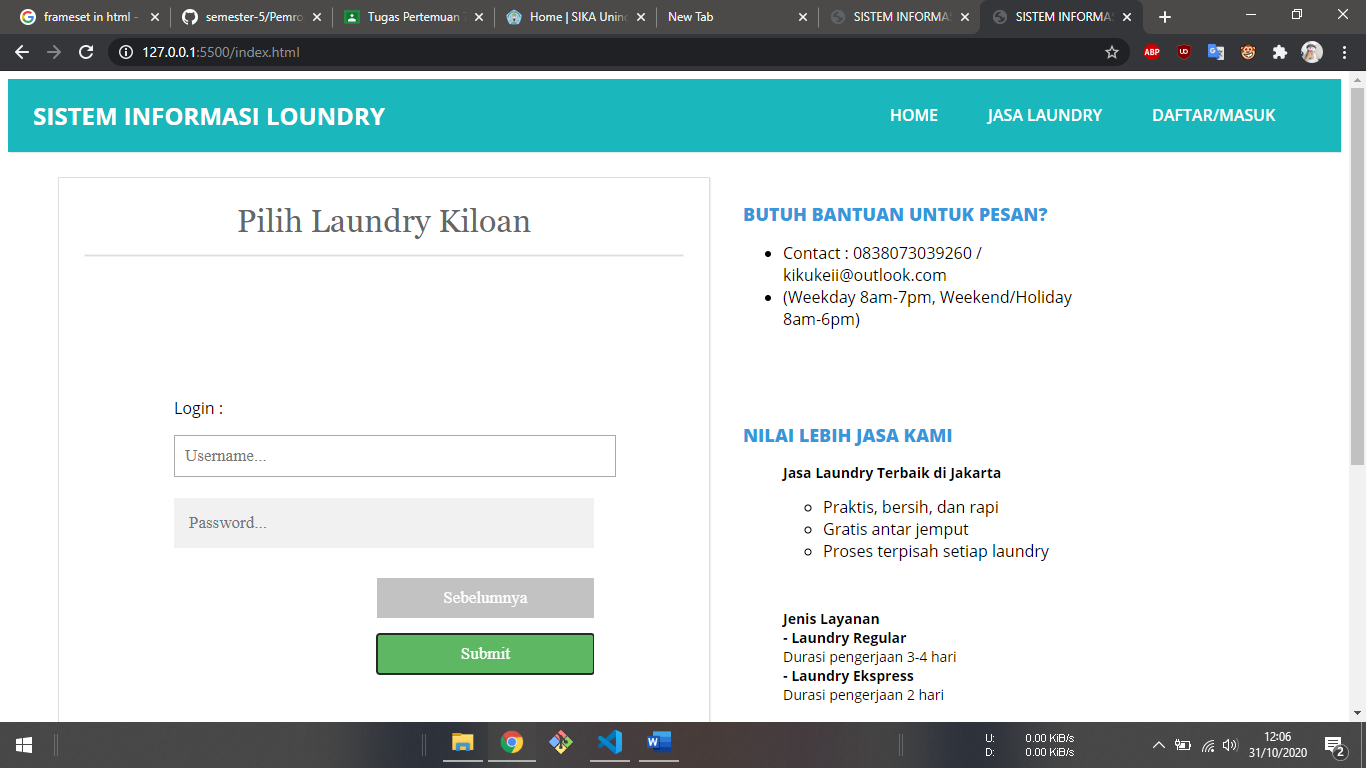




Next kontak dan alamat



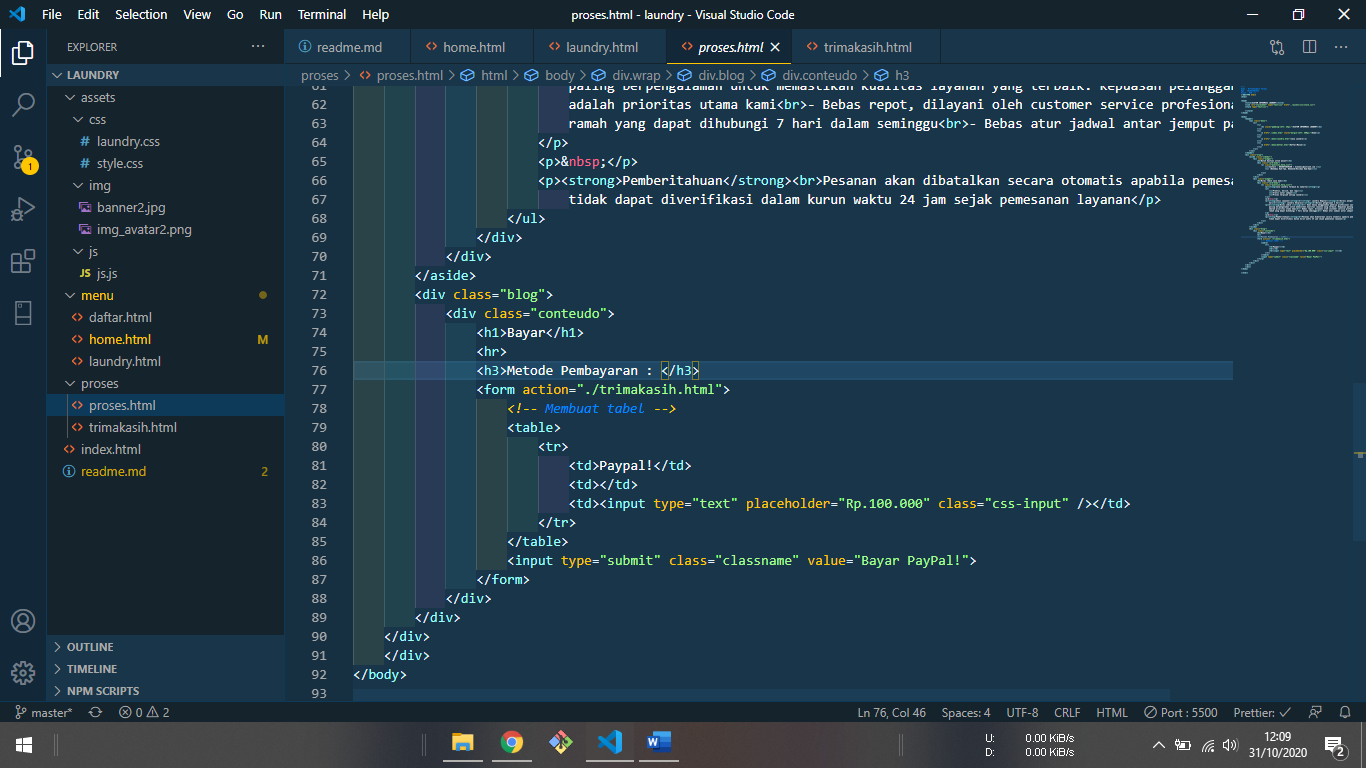
Next diminta masuakan username dan password



Lalu submit akan diarahkan ke proses.html

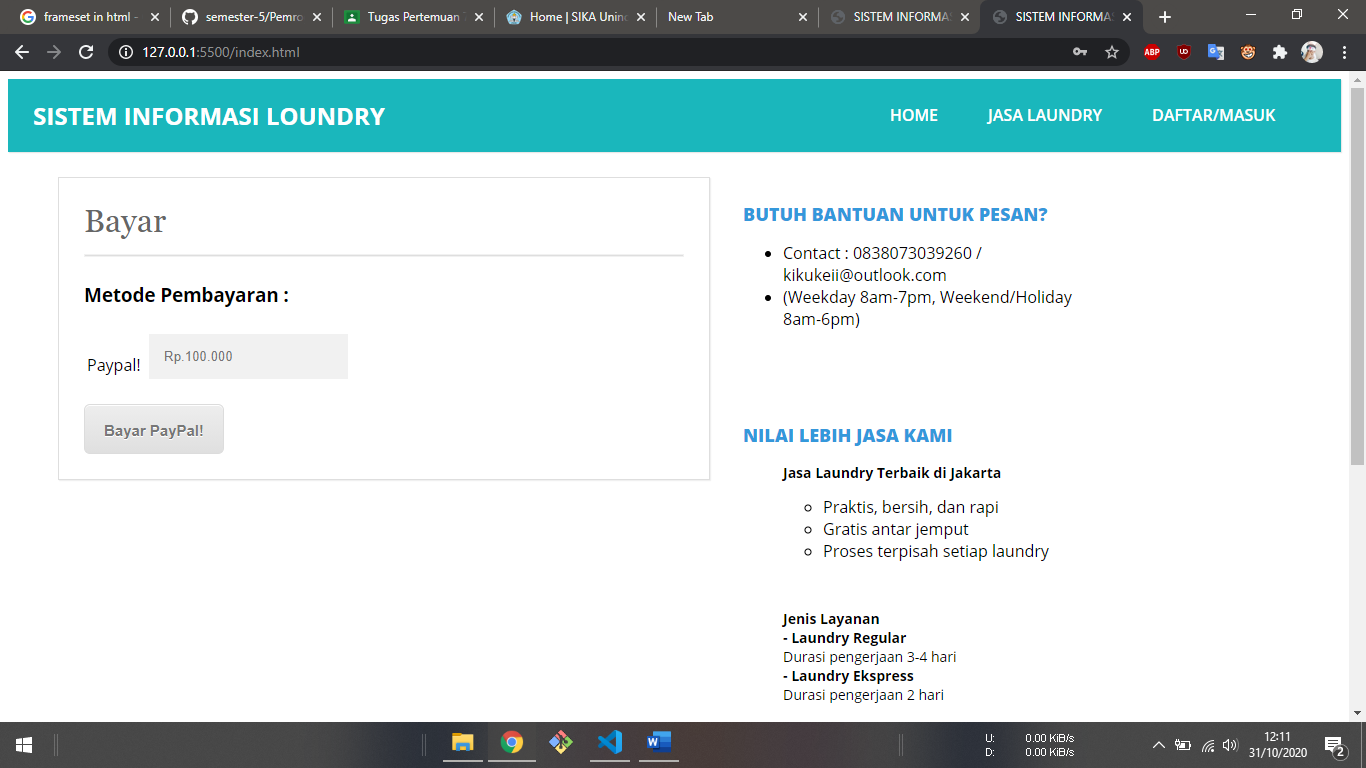
1. proses.html
2. Syntax proses.html

Tag lainya masih sama hanya di warper > blog saja saya tambahkan Form pembayaran



Form akan diarahkan ke terimaksih.html

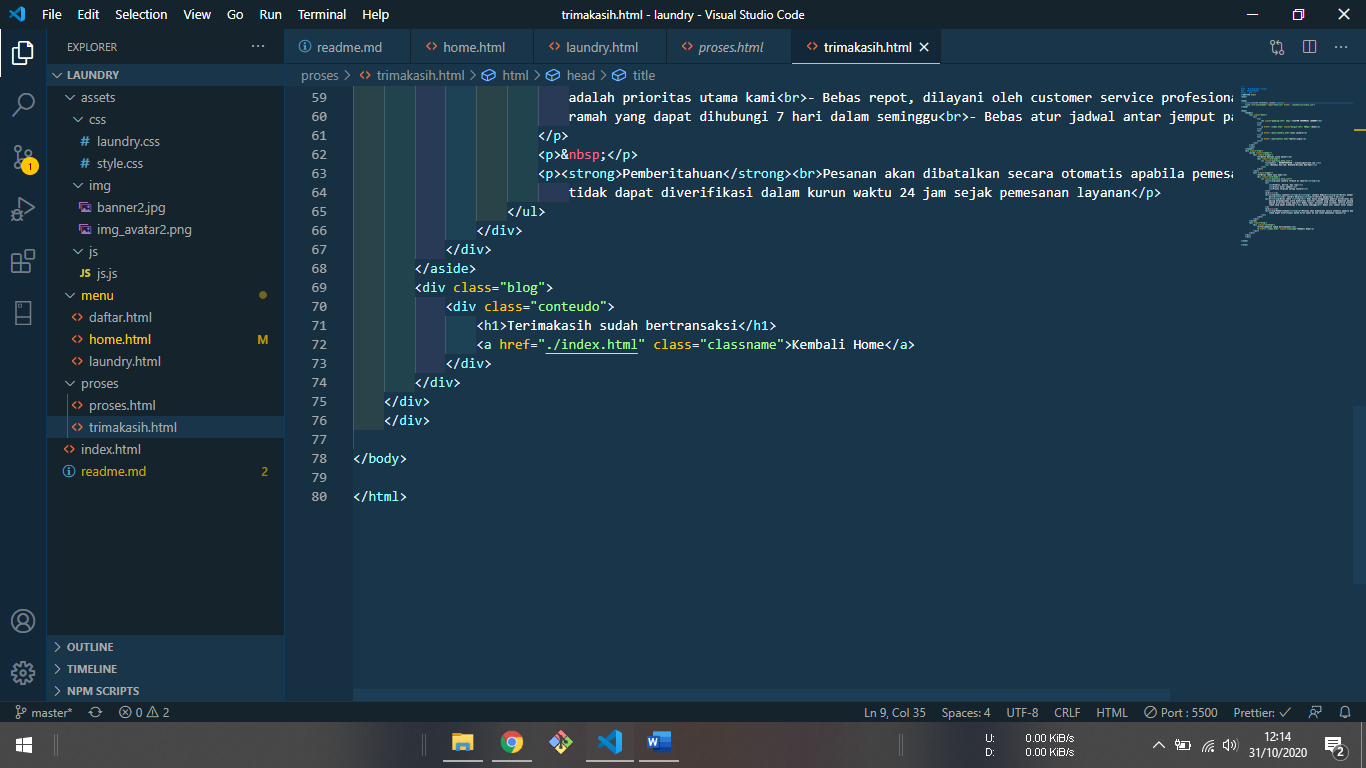
1. Output proses.html



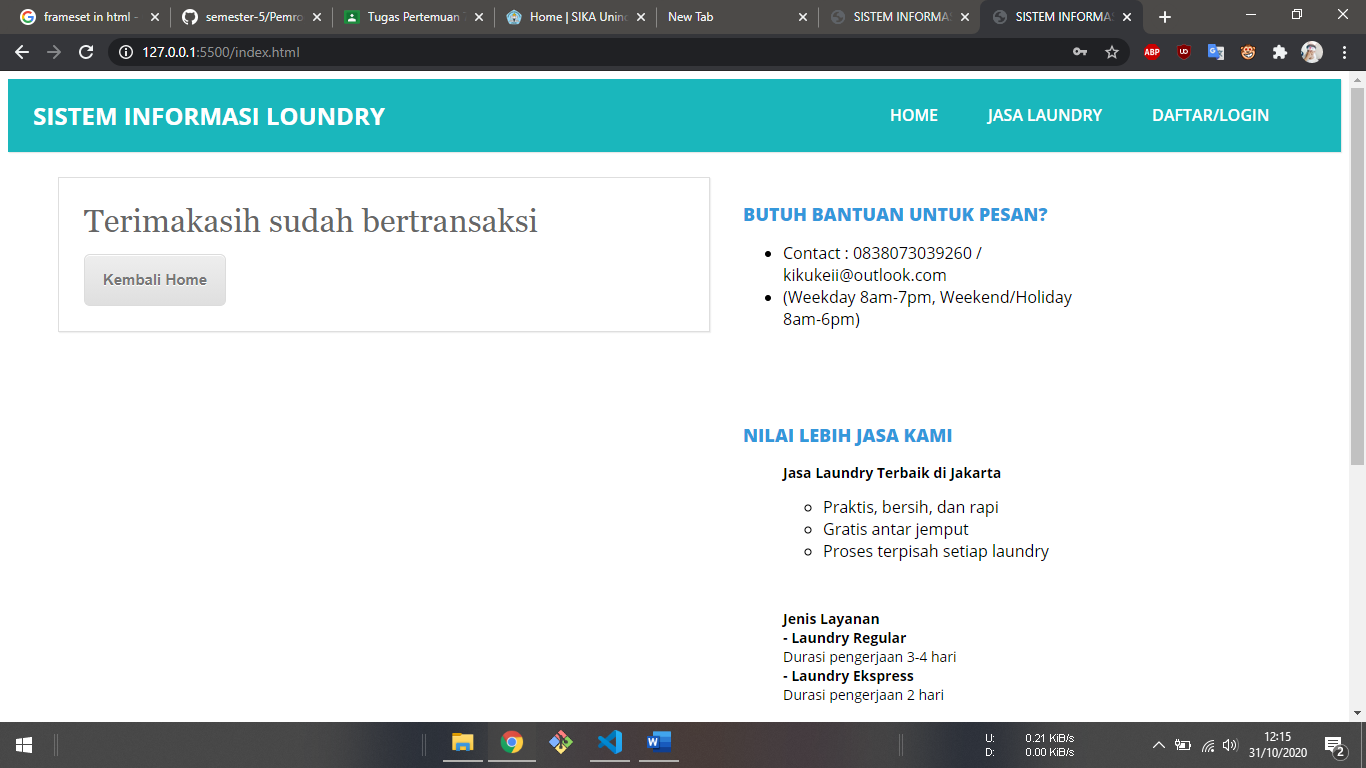
Saat tekan button Bayar Paypal! Akan di arakan ke terimakasih.html

1. terimaksi.html
2. Syntax proses.html

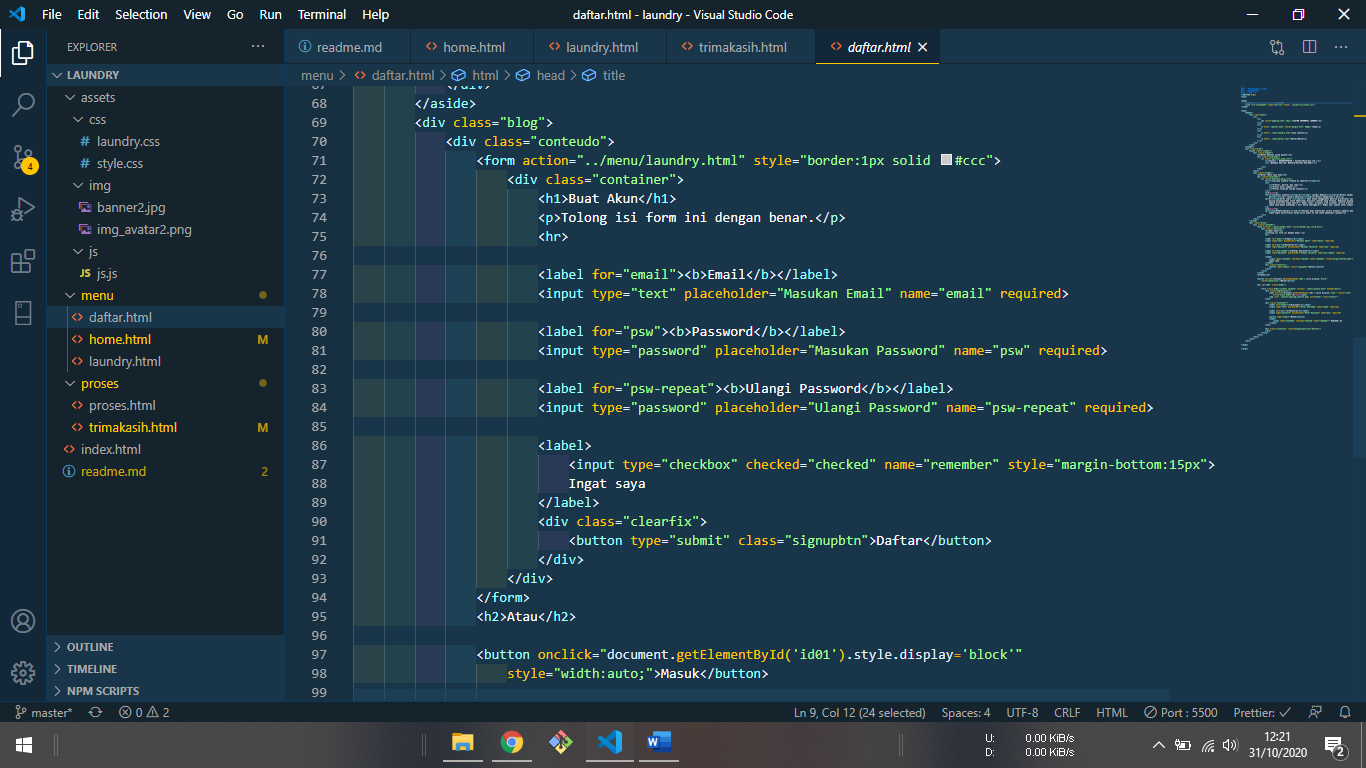
Tag lainya masih sama hanya di warper > blog saja saya tambahkan button Kembali.

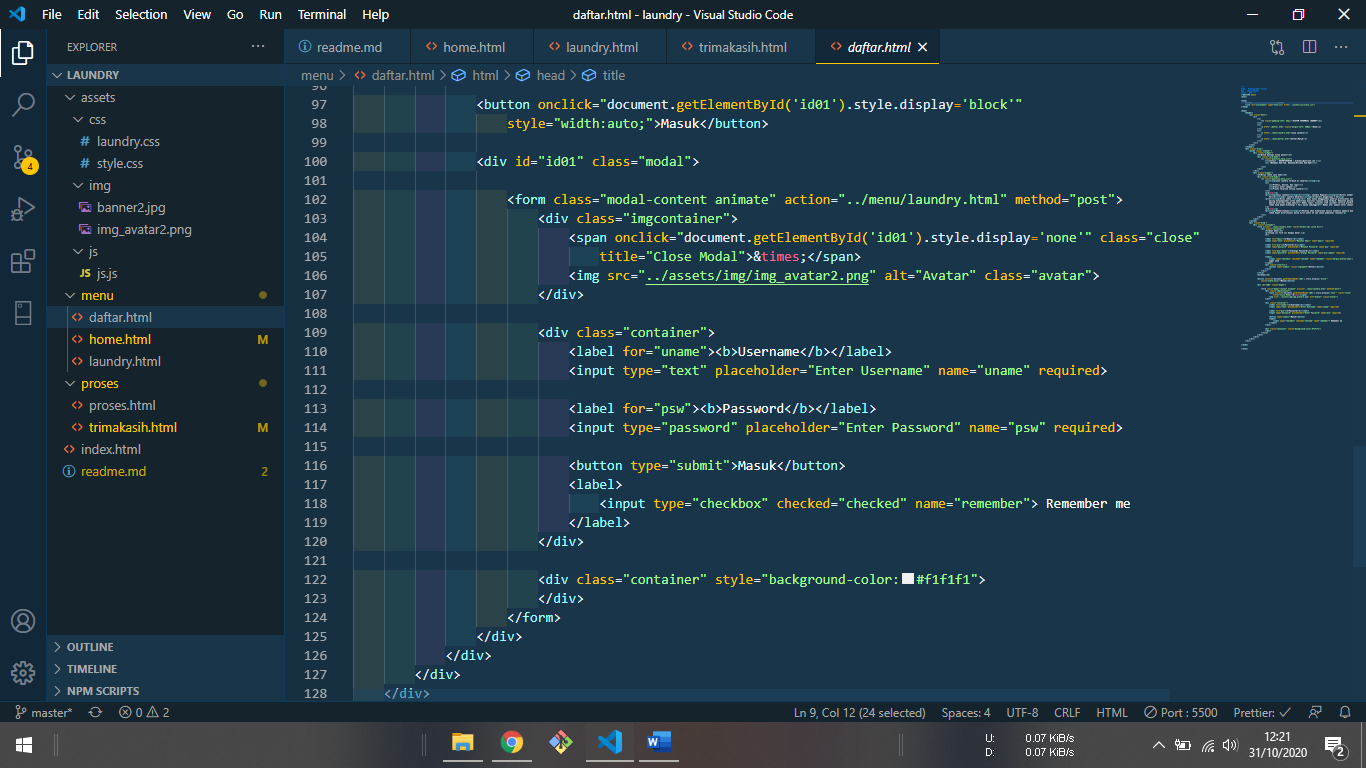


1. Output proses.html



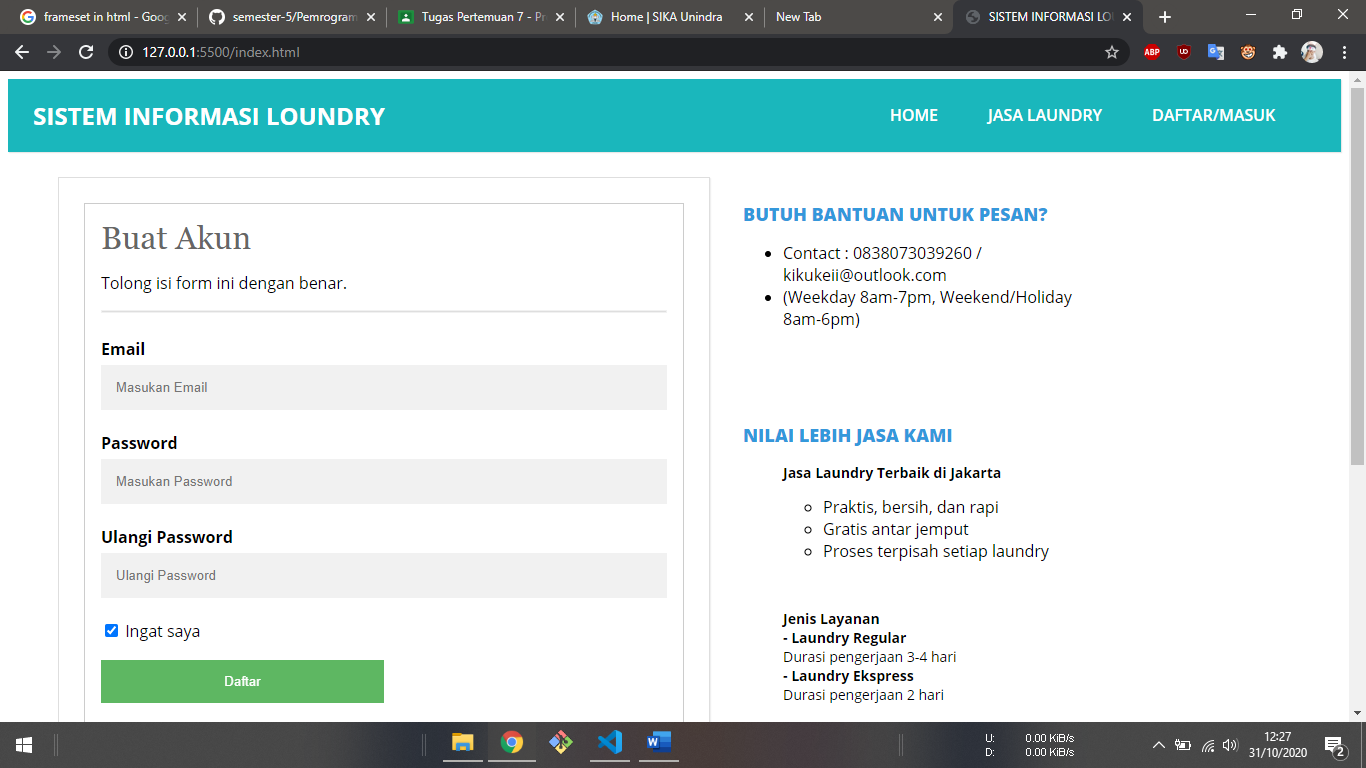
1. Daftar/masuk (daftar.html)
2. Syntax daftar.html





Form daftar dan login action ke laundry.html

1. Output daftar.html



1. Source code

Semua tag dan element peretemuan 1 – 7 sudah ada. Source code bisa diambil/lihat di github saya. Untuk tugas ini : [Klik disini](https://github.com/kikuKeii/semester-5/tree/master/Pemrograman-Web-Dasar/source-code/laundry)

Github-Page : [Klik Disini](https://kikukeii.github.io/semester-5/Pemrograman-Web-Dasar/source-code/laundry/)

1. CSS & JS

# [Syntax style.css (main css)](https://raw.githubusercontent.com/kikuKeii/semester-5/master/Pemrograman-Web-Dasar/source-code/laundry/assets/css/style.css)

/\* <!--

Nama : Miftakhuddin Falaki

Npm : 201843501551

Kelas : R5W --> \*/

@import url(https://fonts.googleapis.com/css?family=Open+Sans:400,800,700,600,300);

body {

font-family: "Open Sans", sans-serif;

}

hr {

border: 0;

background: #dedede;

height: 1px;

}

header {

/\* text-align: center; \*/

color: #232323;

}

header .judul {

font-size: 17pt;

padding-left: 50px;

}

header .deskripsi {

font-size: 11pt;

padding-left: 50px;

}

.wrap {

width: 1050px;

/\* margin: 25px auto; \*/

padding-left: 50px;

}

nav.menu ul {

overflow: hidden;

color: #fff;

list-style: none;

float: left;

padding: 0;

width: 100%;

margin: 0 0 0;

background: #1ab7bc;

-webkit-box-shadow: 1px 1px 1px 0px rgba(204, 204, 204, 0.55);

-moz-box-shadow: 1px 1px 1px 0px rgba(204, 204, 204, 0.55);

box-shadow: 1px 1px 1px 0px rgba(204, 204, 204, 0.55);

}

nav.menu ul li {

margin: 0;

float: left;

}

nav.menu ul a {

padding: 25px;

display: block;

font-weight: 600;

font-size: 16px;

color: #fff;

text-transform: uppercase;

transition: all 0.5s ease;

text-decoration: none;

}

nav.menu ul a:hover {

text-decoration: underline;

background: #1674a0;

}

.sidebar {

padding-top: 25px;

float: right;

width: 390px;

}

.sidebar .widget {

padding: 25px;

margin: 0 0 25px;

background: #fff;

border-bottom: 2px solid #fff;

transition: all 0.5s ease;

}

.sidebar .widget:hover {

border-bottom: 2px solid #3498db;

}

.sidebar .widget h2 {

padding: 0;

margin: 0 0 15px;

color: #3498db;

font-size: 18px;

font-weight: 800;

text-transform: uppercase;

}

.sidebar .widget p {

font-size: 14px;

}

.sidebar .widget p:last-child {

margin: 0;

}

.blog {

float: left;

}

.conteudo {

width: 600px;

padding: 25px;

margin: 25px auto;

background: #fff;

border: 1px solid #dedede;

-webkit-box-shadow: 1px 1px 1px 0px rgba(204, 204, 204, 0.35);

-moz-box-shadow: 1px 1px 1px 0px rgba(204, 204, 204, 0.35);

box-shadow: 1px 1px 1px 0px rgba(204, 204, 204, 0.35);

}

.conteudo img {

min-width: 650px;

margin: 0 0 25px -25px;

max-width: 650px;

}

.conteudo h1 {

padding: 0;

margin: 0 0 15px;

font-weight: normal;

color: #666;

font-family: Georgia;

}

.conteudo p:last-child {

margin: 0;

}

.conteudo .continue-lendo {

color: #000;

transition: all 0.5s ease;

text-decoration: none;

font-weight: 700;

}

.conteudo .continue-lendo:hover {

margin-left: 10px;

}

.post-info {

float: right;

font-size: 12px;

margin: -10px 0 15px;

text-transform: uppercase;

}

@media screen and (max-width: 960px) {

.header {

position: inherit;

}

.wrap {

width: 90%;

margin: 25px auto;

}

.sidebar {

width: 100%;

margin: 25px 0 0;

float: right;

}

.sidebar .widget {

padding: 5%;

}

nav.menu ul {

width: 100%;

}

nav.menu ul {

float: inherit;

}

nav.menu ul li {

float: inherit;

margin: 0;

}

nav.menu ul a {

padding: 15px;

font-size: 16px;

border-top: 1px solid #1abf9f;

border-bottom: 1px solid #16a085;

}

.blog {

width: 90%;

}

.conteudo {

float: inherit;

margin: 0 auto 25px;

width: 101%;

border: 1px solid #dedede;

padding: 5%;

background: #fff;

}

.conteudo img {

max-width: 110%;

margin: 0 0 25px -5%;

min-width: 110%;

}

.conteudo .continue-lendo:hover {

margin-left: 0;

}

}

@media screen and (max-width: 460px) {

nav.menu ul a {

padding: 15px;

font-size: 14px;

}

.sidebar {

display: none;

}

.post-info {

display: none;

}

.conteudo {

margin: 25px auto;

}

.conteudo img {

margin: -5% 0 25px -5%;

}

}

/\* login index \*/

/\* Full-width input fields \*/

input[type="text"],

input[type="password"] {

width: 100%;

padding: 12px 20px;

margin: 8px 0;

display: inline-block;

border: 1px solid #ccc;

box-sizing: border-box;

}

/\* Set a style for all buttons \*/

button {

background-color: #4caf50;

color: white;

padding: 14px 20px;

margin: 8px 0;

border: none;

cursor: pointer;

width: 100%;

}

button:hover {

opacity: 0.8;

}

/\* Extra styles for the cancel button \*/

.cancelbtn {

width: auto;

padding: 10px 18px;

background-color: #f44336;

}

/\* Center the image and position the close button \*/

.imgcontainer {

text-align: center;

margin: 24px 0 12px 0;

position: relative;

}

img.avatar {

width: 40%;

border-radius: 50%;

}

.container {

padding: 16px;

}

span.psw {

float: right;

padding-top: 16px;

}

/\* The Modal (background) \*/

.modal {

display: none; /\* Hidden by default \*/

position: fixed; /\* Stay in place \*/

z-index: 1; /\* Sit on top \*/

left: 0;

top: 0;

width: 100%; /\* Full width \*/

height: 100%; /\* Full height \*/

overflow: auto; /\* Enable scroll if needed \*/

background-color: rgb(0, 0, 0); /\* Fallback color \*/

background-color: rgba(0, 0, 0, 0.4); /\* Black w/ opacity \*/

padding-top: 60px;

}

/\* Modal Content/Box \*/

.modal-content {

background-color: #fefefe;

margin: 5% auto 15% auto; /\* 5% from the top, 15% from the bottom and centered \*/

border: 1px solid #888;

width: 80%; /\* Could be more or less, depending on screen size \*/

}

/\* The Close Button (x) \*/

.close {

position: absolute;

right: 25px;

top: 0;

color: #000;

font-size: 35px;

font-weight: bold;

}

.close:hover,

.close:focus {

color: red;

cursor: pointer;

}

/\* Add Zoom Animation \*/

.animate {

-webkit-animation: animatezoom 0.6s;

animation: animatezoom 0.6s;

}

@-webkit-keyframes animatezoom {

from {

-webkit-transform: scale(0);

}

to {

-webkit-transform: scale(1);

}

}

@keyframes animatezoom {

from {

transform: scale(0);

}

to {

transform: scale(1);

}

}

/\* Change styles for span and cancel button on extra small screens \*/

@media screen and (max-width: 300px) {

span.psw {

display: block;

float: none;

}

.cancelbtn {

width: 100%;

}

}

/\* button \*/

.classname {

-moz-box-shadow: inset 0px 1px 0px 0px #ffffff;

-webkit-box-shadow: inset 0px 1px 0px 0px #ffffff;

box-shadow: inset 0px 1px 0px 0px #ffffff;

background: -webkit-gradient(

linear,

left top,

left bottom,

color-stop(0.05, #ededed),

color-stop(1, #dfdfdf)

);

background: -moz-linear-gradient(center top, #ededed 5%, #dfdfdf 100%);

filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#ededed', endColorstr='#dfdfdf');

background-color: #ededed;

-webkit-border-top-left-radius: 6px;

-moz-border-radius-topleft: 6px;

border-top-left-radius: 6px;

-webkit-border-top-right-radius: 6px;

-moz-border-radius-topright: 6px;

border-top-right-radius: 6px;

-webkit-border-bottom-right-radius: 6px;

-moz-border-radius-bottomright: 6px;

border-bottom-right-radius: 6px;

-webkit-border-bottom-left-radius: 6px;

-moz-border-radius-bottomleft: 6px;

border-bottom-left-radius: 6px;

text-indent: 0;

border: 1px solid #dcdcdc;

display: inline-block;

color: #777777;

font-family: arial;

font-size: 15px;

font-weight: bold;

font-style: normal;

height: 50px;

line-height: 50px;

width: 140px;

text-decoration: none;

text-align: center;

text-shadow: 0px 1px 0px #ffffff;

}

.classname:hover {

background: -webkit-gradient(

linear,

left top,

left bottom,

color-stop(0.05, #dfdfdf),

color-stop(1, #ededed)

);

background: -moz-linear-gradient(center top, #dfdfdf 5%, #ededed 100%);

filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#dfdfdf', endColorstr='#ededed');

background-color: #dfdfdf;

}

.classname:active {

position: relative;

top: 1px;

}

/\* sign up \*/

input[type="text"],

input[type="password"] {

width: 100%;

padding: 15px;

margin: 5px 0 22px 0;

display: inline-block;

border: none;

background: #f1f1f1;

}

input[type="text"]:focus,

input[type="password"]:focus {

background-color: #ddd;

outline: none;

}

hr {

border: 1px solid #f1f1f1;

margin-bottom: 25px;

}

/\* Set a style for all buttons \*/

button {

background-color: #4caf50;

color: white;

padding: 14px 20px;

margin: 8px 0;

border: none;

cursor: pointer;

width: 100%;

opacity: 0.9;

}

button:hover {

opacity: 1;

}

/\* Extra styles for the cancel button \*/

.cancelbtn {

padding: 14px 20px;

background-color: #f44336;

}

/\* Float cancel and signup buttons and add an equal width \*/

.cancelbtn,

.signupbtn {

float: left;

width: 50%;

}

/\* Add padding to container elements \*/

.container {

padding: 16px;

}

/\* Clear floats \*/

.clearfix::after {

content: "";

clear: both;

display: table;

}

/\* Change styles for cancel button and signup button on extra small screens \*/

@media screen and (max-width: 300px) {

.cancelbtn,

.signupbtn {

width: 100%;

}

}

# [Syntax laundy.css](https://raw.githubusercontent.com/kikuKeii/semester-5/master/Pemrograman-Web-Dasar/source-code/laundry/assets/css/laundry.css)

/\* <!--

Nama : Miftakhuddin Falaki

Npm : 201843501551

Kelas : R5W --> \*/

/\* form \*/

#regForm {

background-color: #ffffff;

margin: 100px auto;

padding: 40px;

width: 70%;

min-width: 300px;

}

h1 {

text-align: center;

}

input {

padding: 10px;

width: 100%;

font-size: 17px;

font-family: Raleway;

border: 1px solid #aaaaaa;

}

/\* Mark input boxes that gets an error on validation: \*/

input.invalid {

background-color: #ffdddd;

}

/\* Hide all steps by default: \*/

.tab {

display: none;

}

button {

background-color: #4caf50;

color: #ffffff;

border: none;

padding: 10px 20px;

font-size: 17px;

font-family: Raleway;

cursor: pointer;

}

button:hover {

opacity: 0.8;

}

#prevBtn {

background-color: #bbbbbb;

}

/\* Make circles that indicate the steps of the form: \*/

.step {

height: 15px;

width: 15px;

margin: 0 2px;

background-color: #bbbbbb;

border: none;

border-radius: 50%;

display: inline-block;

opacity: 0.5;

}

.step.active {

opacity: 1;

}

/\* Mark the steps that are finished and valid: \*/

.step.finish {

background-color: #4caf50;

}

.css-input {

padding: 5px;

font-size: 16px;

border-width: 1px;

border-color: #cccccc;

background-color: #ffffff;

color: #000000;

border-style: solid;

border-radius: 18px;

box-shadow: 0px 0px 5px rgba(66, 66, 66, 0.75);

text-shadow: 0px 0px 5px rgba(66, 66, 66, 0.75);

}

.css-input:focus {

outline: none;

}

.container {

display: block;

position: relative;

padding-left: 35px;

margin-bottom: 12px;

cursor: pointer;

font-size: 22px;

-webkit-user-select: none;

-moz-user-select: none;

-ms-user-select: none;

user-select: none;

}

/\* Hide the browser's default radio button \*/

.container input {

position: absolute;

opacity: 0;

cursor: pointer;

}

/\* Create a custom radio button \*/

.checkmark {

position: absolute;

top: 0;

left: 0;

height: 25px;

width: 25px;

background-color: #eee;

border-radius: 50%;

}

/\* On mouse-over, add a grey background color \*/

.container:hover input ~ .checkmark {

background-color: #ccc;

}

/\* When the radio button is checked, add a blue background \*/

.container input:checked ~ .checkmark {

background-color: #2196f3;

}

/\* Create the indicator (the dot/circle - hidden when not checked) \*/

.checkmark:after {

content: "";

position: absolute;

display: none;

}

/\* Show the indicator (dot/circle) when checked \*/

.container input:checked ~ .checkmark:after {

display: block;

}

/\* Style the indicator (dot/circle) \*/

.container .checkmark:after {

top: 9px;

left: 9px;

width: 8px;

height: 8px;

border-radius: 50%;

background: white;

}

/\* select \*/

.select {

position: relative;

display: inline-block;

margin-bottom: 15px;

width: 100%;

}

.select select {

font-family: "Arial";

display: inline-block;

width: 100%;

cursor: pointer;

padding: 10px 15px;

outline: 0;

border: 1px solid #000000;

border-radius: 0px;

background: #ffffff;

color: #7b7b7b;

appearance: none;

-webkit-appearance: none;

-moz-appearance: none;

}

.select select::-ms-expand {

display: none;

}

.select select:hover,

.select select:focus {

color: #000000;

background: #ffffff;

}

.select select:disabled {

opacity: 0;

pointer-events: none;

}

.select\_arrow {

position: absolute;

top: 16px;

right: 15px;

pointer-events: none;

border-style: solid;

border-width: 8px 5px 0px 5px;

border-color: #7b7b7b transparent transparent transparent;

}

.select select:hover ~ .select\_arrow,

.select select:focus ~ .select\_arrow {

border-top-color: #000000;

}

.select select:disabled ~ .select\_arrow {

border-top-color: #cccccc;

}

# [Stnyax js.js](https://raw.githubusercontent.com/kikuKeii/semester-5/master/Pemrograman-Web-Dasar/source-code/laundry/assets/js/js.js)

// <!--

// Nama : Miftakhuddin Falaki

// Npm : 201843501551

// Kelas : R5W -->

var currentTab = 0; // Current tab is set to be the first tab (0)

showTab(currentTab); // Display the current tab

function showTab(n) {

// This function will display the specified tab of the form...

var x = document.getElementsByClassName("tab");

x[n].style.display = "block";

//... and fix the Previous/Next buttons:

if (n == 0) {

document.getElementById("prevBtn").style.display = "none";

} else {

document.getElementById("prevBtn").style.display = "inline";

}

if (n == (x.length - 1)) {

document.getElementById("nextBtn").innerHTML = "Submit";

} else {

document.getElementById("nextBtn").innerHTML = "Next";

}

//... and run a function that will display the correct step indicator:

fixStepIndicator(n)

}

function nextPrev(n) {

// This function will figure out which tab to display

var x = document.getElementsByClassName("tab");

// Exit the function if any field in the current tab is invalid:

if (n == 1 && !validateForm()) return false;

// Hide the current tab:

x[currentTab].style.display = "none";

// Increase or decrease the current tab by 1:

currentTab = currentTab + n;

// if you have reached the end of the form...

if (currentTab >= x.length) {

// ... the form gets submitted:

document.getElementById("regForm").submit();

return false;

}

// Otherwise, display the correct tab:

showTab(currentTab);

}

function validateForm() {

// This function deals with validation of the form fields

var x, y, i, valid = true;

x = document.getElementsByClassName("tab");

y = x[currentTab].getElementsByTagName("input");

// A loop that checks every input field in the current tab:

for (i = 0; i < y.length; i++) {

// If a field is empty...

if (y[i].value == "") {

// add an "invalid" class to the field:

y[i].className += " invalid";

// and set the current valid status to false

valid = false;

}

}

// If the valid status is true, mark the step as finished and valid:

if (valid) {

document.getElementsByClassName("step")[currentTab].className += " finish";

}

return valid; // return the valid status

}

function fixStepIndicator(n) {

// This function removes the "active" class of all steps...

var i, x = document.getElementsByClassName("step");

for (i = 0; i < x.length; i++) {

x[i].className = x[i].className.replace(" active", "");

}

//... and adds the "active" class on the current step:

x[n].className += " active";

}

// Get the modal

var modal = document.getElementById('id01');

// When the user clicks anywhere outside of the modal, close it

window.onclick = function (event) {

if (event.target == modal) {

modal.style.display = "none";

}

}