VISOKA ŠKOLA STRUKOVNIH STUDIJA ZA INFORMACIONE I KOMUNIKACIONE TEHNOLOGIJE

DOKUMENTACIJA

<https://gringus0.github.io/Kriptografija/>

Web Programiranje 1

Aleksandar Jovanović 104/21

Smer: Internet Tehnologije

Modul: Web Programiranje

Predmet: Web Programiranje 1

Contents

[Uvod 3](#_Toc121994233)

[1. Programski jezici: 3](#_Toc121994234)

[2. Funkcionalnosti: 3](#_Toc121994235)

[3. Template: 3](#_Toc121994236)

[Organizacija i šema sajta: 5](#_Toc121994237)

[1. Sajt se sastoji samo od stranice index.html 5](#_Toc121994238)

[2. Sitemap 5](#_Toc121994239)

[3. Slike stranice i opisi funkcionalnosti: 6](#_Toc121994240)

[HTML: 8](#_Toc121994241)

[CSS: 18](#_Toc121994242)

[JavaScript: 34](#_Toc121994243)

# Uvod

## 1. Programski jezici:

Programski jezici korišćeni u izradi sajta: HTML5, CSS3, JavaScript, JQuery, XML

Radno okruženje: Visual Studio Code

Radi lakšeg dizajniranja sajta korišćen je: Bootstrap

## 2. Funkcionalnosti:

1. Dinamički ispisan navigacioni meni

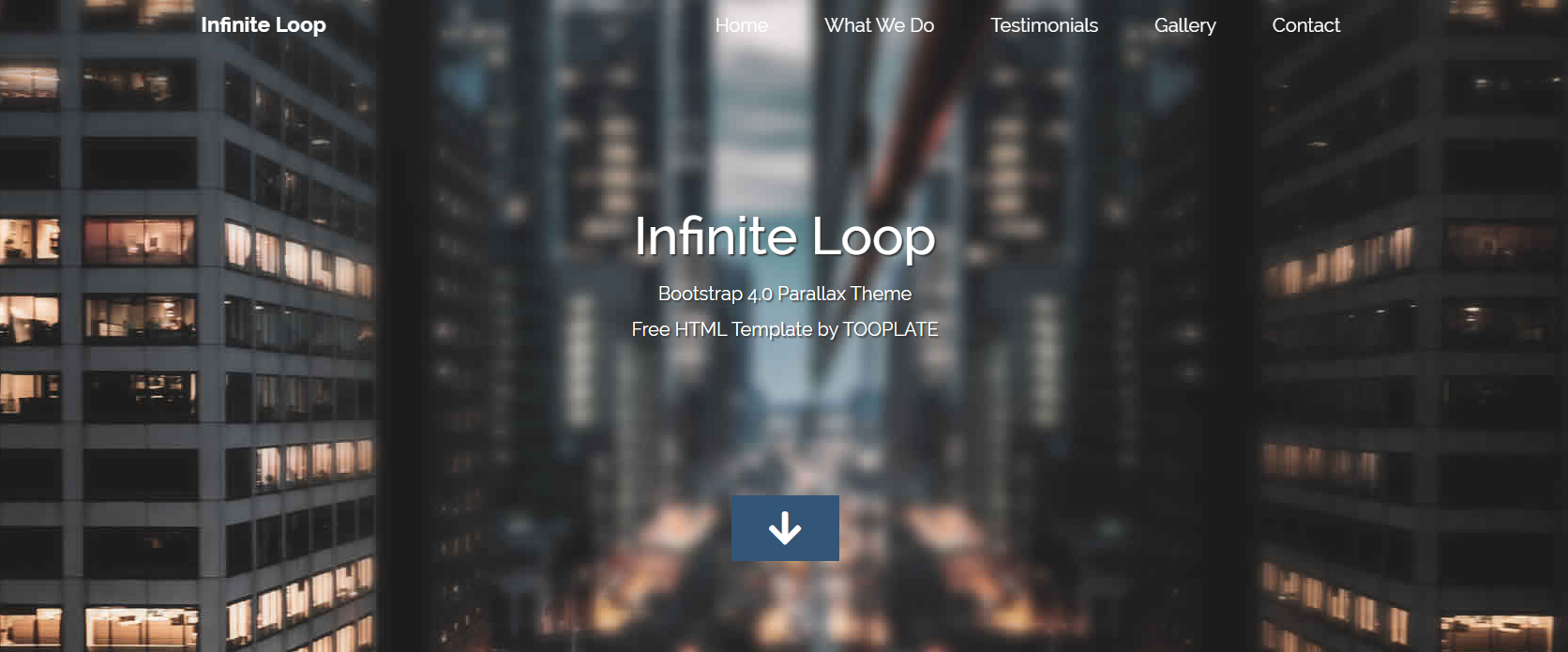
2. Mogućnost korišćenja sajta za šifrovanje/dešifrovanje teksta po kvadratnoj šifri

3. Tajmer sa odbrojavanjem do određenog datuma

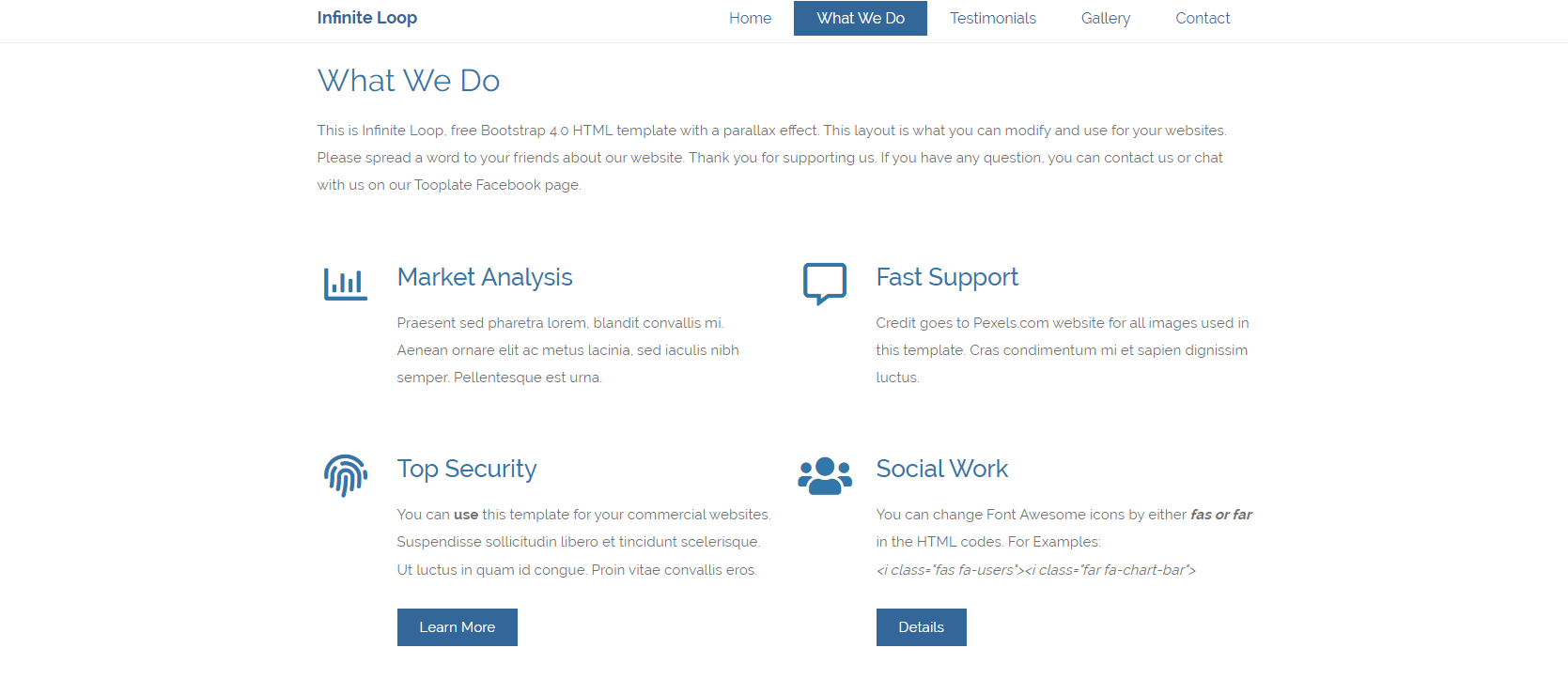
4. Dinamički ispisana forma uz pomoć Create Element i regEx proverama

## 3. Template:

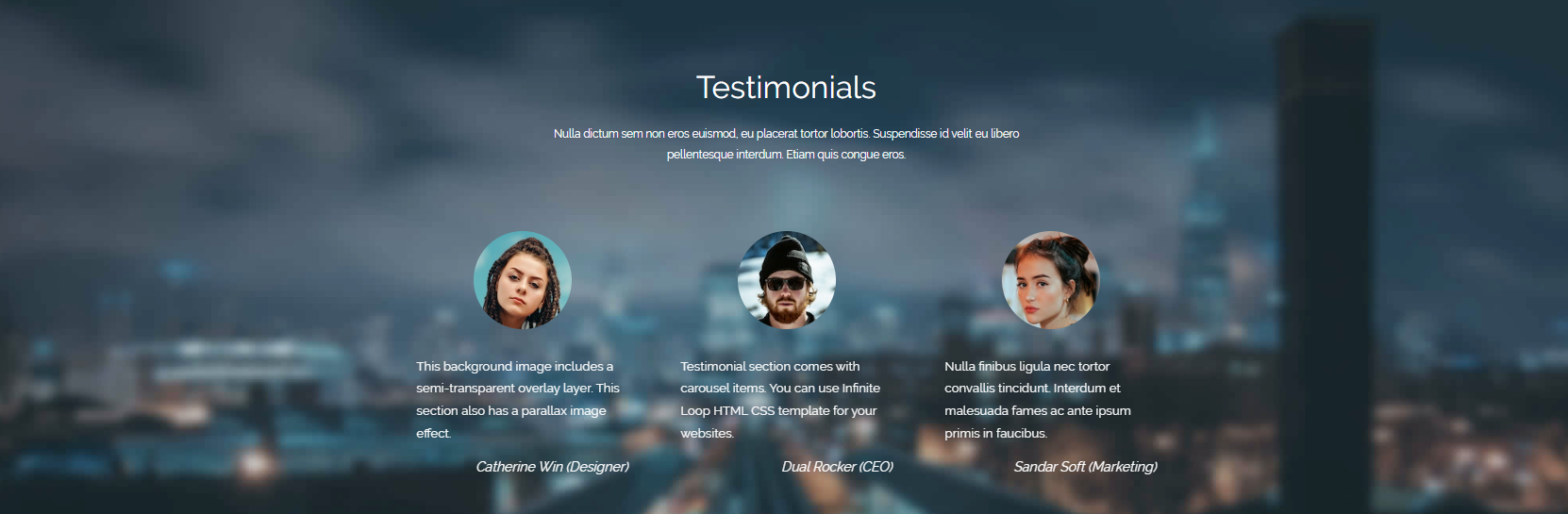
https://www.free-css.com/free-css-templates/page272/infinite-loop



Korišćena je početna stranica sa promenom pozadine i navigacije uz dinamičko ispisivanje.



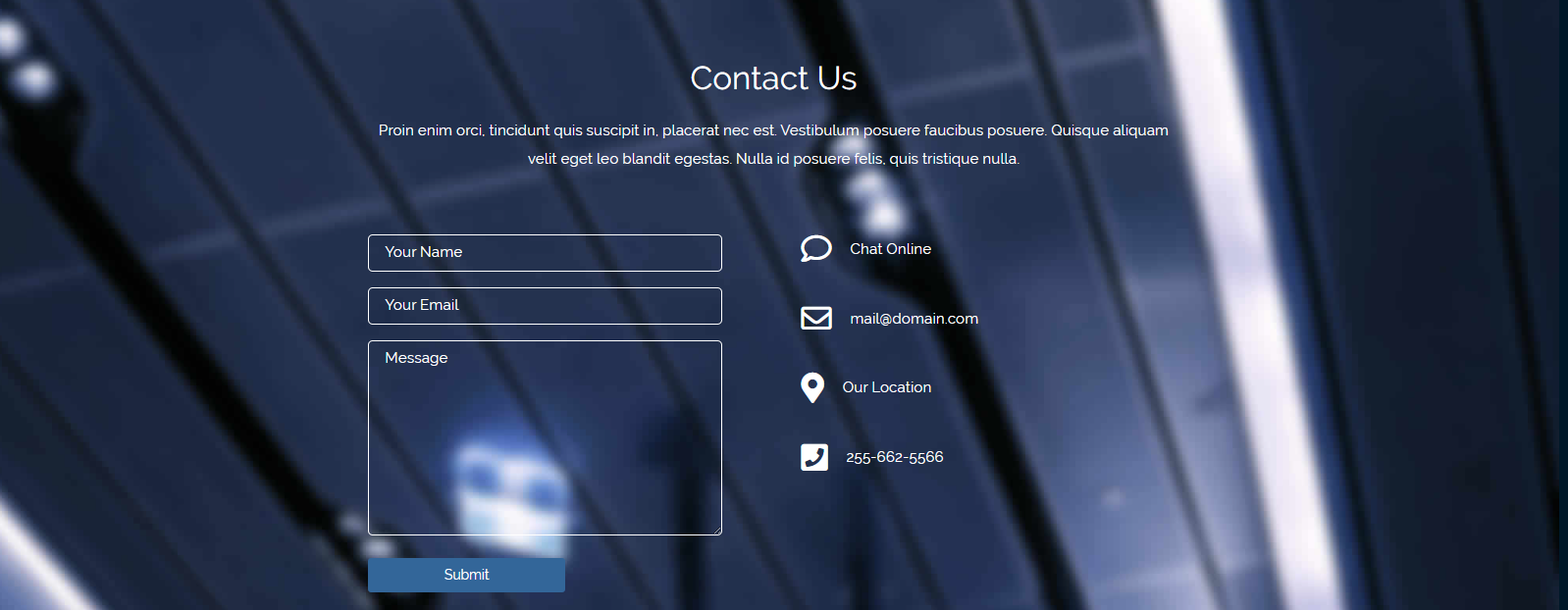
Korišćeno za ispisivanje teksta o temi sajta.



Promenjen deo koji je bio iskorišćen za ispisivanje primera jedne od šifara.



Kompletno promenjen deo u textbox za šifrovanje i dešifrovanje.



Forma je delimično iskorišćena uz prepravke zajedno sa dinamičkim ispisivanjem.

# Organizacija i šema sajta:

## Sajt se sastoji samo od stranice index.html

## Sitemap

1. <?xml version="1.0" encoding="UTF-8"?>
2. <urlset
3. xmlns="http://www.sitemaps.org/schemas/sitemap/0.9"
4. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5. xsi:schemaLocation="http://www.sitemaps.org/schemas/sitemap/0.9
6. http://www.sitemaps.org/schemas/sitemap/0.9/sitemap.xsd">
7. <url>
8. <loc>https://gringus0.github.io/Kriptografija/</loc>
9. <lastmod>2022-12-15T00:37:29+00:00</lastmod>
10. <changefreq>Monthly</changefreq>
11. <priority>1.00</priority>
12. </url>
13. </urlset>

## 3. Slike stranice i opisi funkcionalnosti:



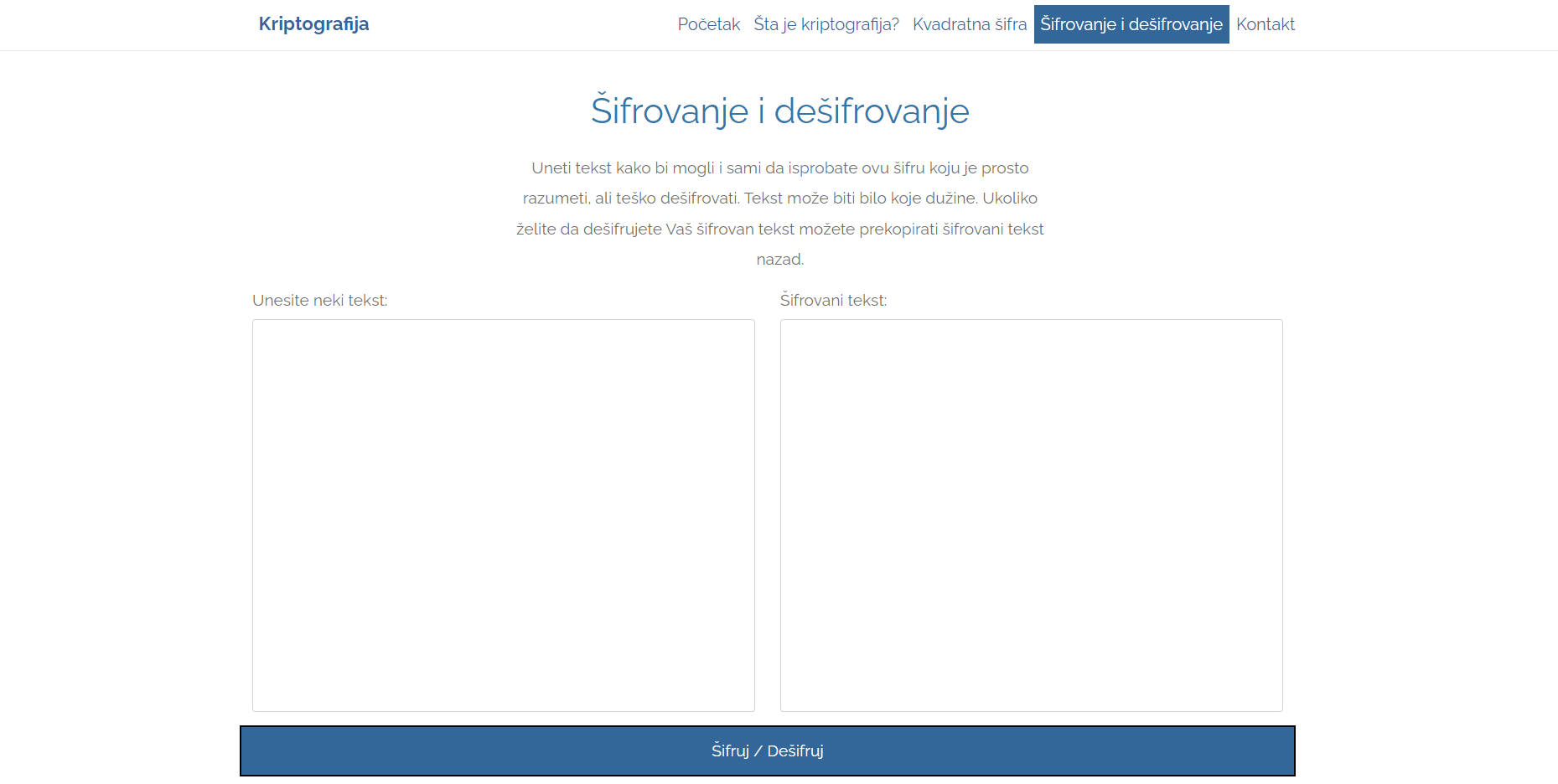
Početna stranica se sastoji iz naslova i navigacije. Klikom na linkove navigacionog menija se može otići na druge delove stranice.



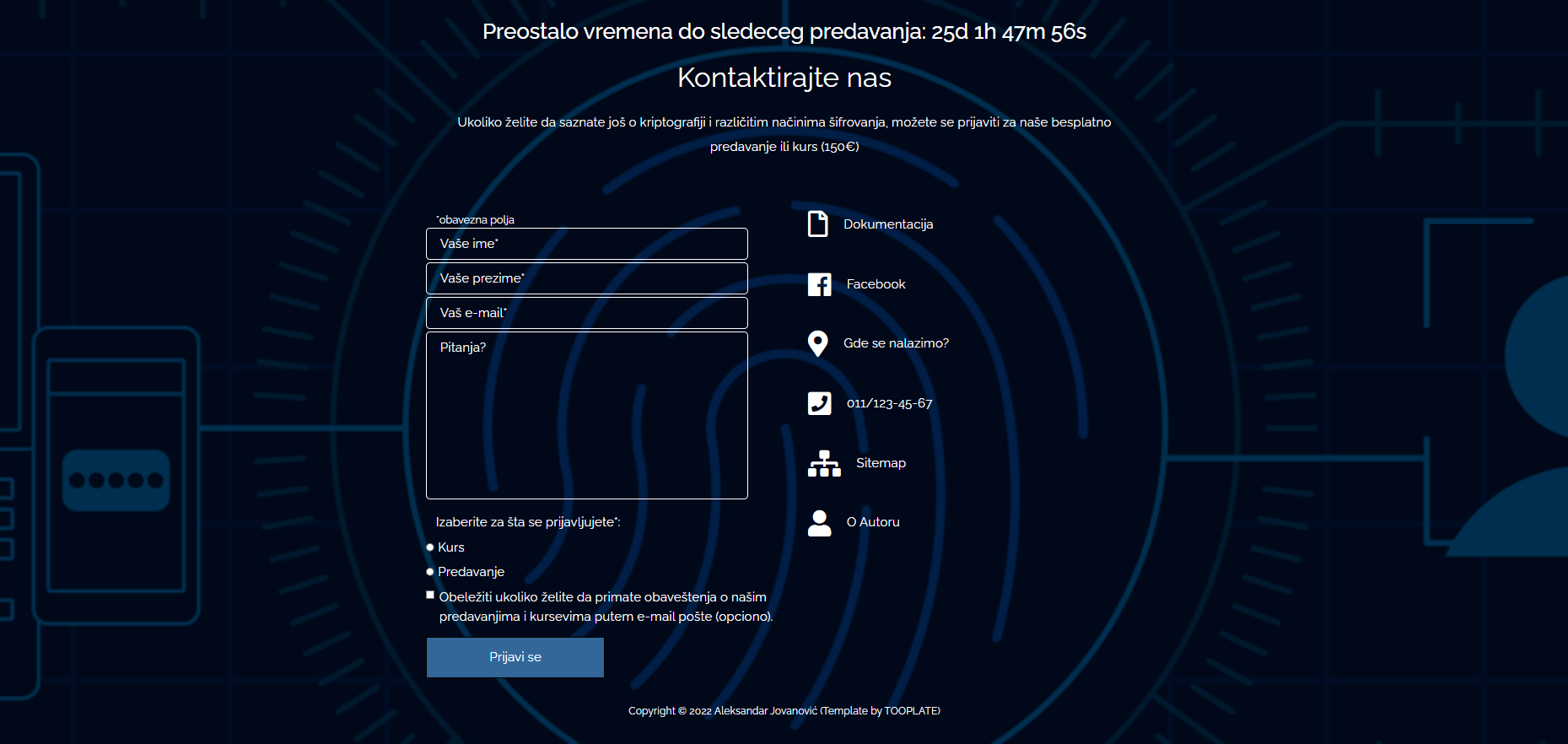
Dodatne informacije o šifrovanju. Klikom na „saznaj više” se otkriva i sakriva dodatni tekst. Klikom na „o šifrovanju“ se prosleđuje dalje na stranicu.



Detaljan opis primera šifrovanja i dešifrovanja kvadratnom šifrom.



Mogućnost isprobavanja kvadratne šifre za šifrovanje i dešifrovanje. Nakon unosa teksta u tekstualno polje, klikom na taster “šifruj/dešifruj” moguće je šifrovati ili dešifrovati ključem kvadratne šifre.



Dinamički ispisana kontakt forma sa Create Element sa proverom grešaka prilikom klika na taster “Prijavi se” I prilikom izlaska sa bilo kog od obaveznih polja. Tajmer koji odbrojava do fiksnog datum i vremena.

# HTML:

Pored dodavanja klasa I id-jeva kod koji sam iskucao u HTML-u je sve sem obojeno žutom bojom.

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <meta http-equiv="X-UA-Compatible" content="ie=edge" />

    <meta name="author" content="Aleksandar Jovanović 104/21"/>

    <meta name="description" content="Kriptografija sa primerom kvadratne šifre."/>

    <meta name="keywords" content="kriptografija, šifrovanje, kvadratna šifra"/>

    <link rel="icon" href="img/key\_icon.png"/>

    <title>Kriptografija</title>

    <link rel="stylesheet" href="fontawesome-5.5/css/all.min.css" />

    <link rel="stylesheet" href="slick/slick.css">

    <link rel="stylesheet" href="slick/slick-theme.css">

    <link rel="stylesheet" href="magnific-popup/magnific-popup.css">

    <link rel="stylesheet" href="css/bootstrap.min.css" />

    <link rel="stylesheet" href="css/tooplate-infinite-loop.css" />

<!--

Tooplate 2117 Infinite Loop

https://www.tooplate.com/view/2117-infinite-loop

-->

  </head>

  <body>

    <!-- Navigacija -->

    <section id="pocetak" class="text-white tm-font-big tm-parallax">

      <!-- Navigation -->

      <nav class="navbar navbar-expand-lg tm-navbar" id="tmNav">

        <div class="container">

          <div class="tm-next">

              <a href="#pocetak" class="navbar-brand">Kriptografija</a>

          </div>

          <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">

            <i class="fas fa-bars navbar-toggler-icon"></i>

          </button>

          <div class="collapse navbar-collapse" id="navbarSupportedContent">

            <ul class="navbar-nav ml-auto" id="navitems">

            </ul>

          </div>

        </div>

      </nav>

      <div class="text-center tm-hero-text-container">

        <div class="tm-hero-text-container-inner">

            <h2 class="tm-hero-title">Kriptografija</h2>

            <p class="tm-hero-subtitle">

              Čime se bavi?

              <br/>Jedan mali primer

            </p>

        </div>

      </div>

      <div class="tm-next tm-intro-next">

        <a href="#staJeKriptografija" class="text-center tm-down-arrow-link">

          <i class="fas fa-2x fa-arrow-down tm-down-arrow"></i>

        </a>

      </div>

    </section>

    <!-- Šta je kriptografija? -->

    <section id="staJeKriptografija" class="tm-section-pad-top">

      <div class="container">

            <div class="row tm-content-box"><!-- first row -->

                <div class="col-lg-12 col-xl-12">

                    <div class="tm-intro-text-container">

                        <h2 class="tm-text-primary mb-4 tm-section-title">Šta je kriptografija?</h2>

                        <p class="mb-4 tm-intro-text" id="oKriptografiji">

                          Kriptografija je nauka koja se bavi metodima očuvanja tajnosti informacija. Kada se lične, finansijske, vojne ili informacije državne bezbednosti prenose sa mesta na mesto, one postaju ranjive na prisluškivačke taktike.</p><p id="nastavak"> Ovakvi problemi se mogu izbeći kriptovanjem (šifrovanjem) informacija koje ih čini nedostupnim neželjenoj strani. Šifra i digitalni potpis su kriptografske tehnike koje se koriste da bi se implementirali bezbednosni servisi. Osnovni element koji se koristi naziva se šifarski sistem ili algoritam šifrovanja. Svaki šifarski sistem obuhvata par transformacija podataka, koje se nazivaju šifrovanje i dešifrovanje.</p>

                          <a href="#staJeKriptografija" id="taster" class="tm-intro-text tm-btn-primary btn-primary">Saznaj više</a>

                    </div>

                </div>

            </div><!-- first row -->

            <div class="row tm-content-box"><!-- second row -->

            <div class="col-lg-1">

                    <i class="far fa-3x fa-eye-slash text-center tm-icon"></i>

                </div>

                <div class="col-lg-5">

                    <div class="tm-intro-text-container">

                        <h2 class="tm-text-primary mb-4">Šifrovanje</h2>

                        <p class="mb-4 tm-intro-text">

                          Procedura koja transformiše originalnu informaciju (otvoreni tekst) u šifrovane podatke (šifrat)</p>

                    </div>

                </div>

                <div class="col-lg-1">

                    <i class="far fa-3x fa-eye-slash text-center tm-icon"></i>

                </div>

                <div class="col-lg-5">

                    <div class="tm-intro-text-container">

                        <h2 class="tm-text-primary mb-4">Dešifrovanje</h2>

                        <p class="mb-4 tm-intro-text">

                          Obrnut proces, dešifrovanje, rekonstruiše otvoreni tekst na osnovu šifrata.</p>

                    </div>

                </div>

            </div><!-- second row -->

            <div class="row tm-content-box"><!-- third row -->

            <div class="col-lg-1">

                    <i class="fas fa-3x fa-lock text-center tm-icon"></i>

                </div>

                <div class="col-lg-5">

                    <div class="tm-intro-text-container">

                        <h2 class="tm-text-primary mb-4">Ključ Šifrovanja</h2>

                        <p class="mb-4 tm-intro-text">

                          Prilikom šifrovanja, pored otvorenog teksta, koristi se jedna nezavisna vrednost koja se naziva ključ šifrovanja.</p>

                    </div>

                </div>

                <div class="col-lg-1">

                    <i class="fas fa-3x fa-key text-center tm-icon"></i>

                </div>

                <div class="col-lg-5">

                    <div class="tm-intro-text-container">

                      <h2 class="tm-text-primary mb-4">Ključ dešifrovanja</h2>

                        <p class="mb-4 tm-intro-text">

                          Slično kao i kod šifrovanja, transformacija za dešifrovanje koristi ključ dešifrovanja.</p>

                    </div>

                </div>

            </div><!-- third row -->

            <div class="tm-continue row text-center mx-5">

              <a href="#kvadratnaSifra" class="tm-intro-text tm-btn-primary col-12">O šifrovanju</a>

          </div>

        </div>

    </section>

    <!-- Kvadratna šifra -->

    <section id="kvadratnaSifra" class="tm-section-pad-top tm-parallax-2">

      <div class="container tm-testimonials-content">

        <div class="row">

          <div class="col-lg-12 tm-content-box">

            <h2 class="text-white text-center mb-4 tm-section-title">Kvadratna šifra</h2>

            <p class="mx-auto tm-section-desc text-center aj-kvadratna-sifra-tekst">

                Kvadratna šifra je jedan od najzanimljivijih načina šifrovanja. Prvo se prebroji broj karaktera (slova, brojeva, razmaka, uzvičnika, taraba, itd.), a nakon toga se računa koji je najbliži(manji) kvadrat broja koga smo dobili. Npr. ukoliko smo uneli 70 karaktera, najbliži manji kvadrat od 70 je 64 (8\*8). Prvih 64 karaktera rasporedimo u kvadrat oblika 8\*8. Krećemo da ih pišemo u kvadrat od gornjeg levog ugla na desno dok nam ne ponestane mesta u tom redu. Nakon toga silazimo u novi red i nastavljamo dalje da pišemo. Kada ispunimo kvadrat, te karaktere koje smo upisali, prepisujemo počevši od gornjeg levog ugla, na dole i tako redom. U našem slučaju imamo 6 karaktera "viška" (70-64=6). Sa preostalim karakterima samo ponovimo isto što smo i do sada radili sve dok nam ne ostane 3 ili manje karaktera. Taj, najmanji ostatak samo prepišemo na kraj. Dešifrovanje se radi identično kao i šifrovanje.

              </p>

          </div>

        </div>

      </div>

      <div class="tm-bg-overlay"></div>

    </section>

    <!-- Šifrovanje i dešifrovanje -->

    <section id="sifrovanje" class="tm-section-pad-top">

      <div class="container tm-container-gallery">

        <div class="row">

          <div class="text-center col-12">

              <h2 class="tm-text-primary tm-section-title mb-4">Šifrovanje i dešifrovanje</h2>

              <p class="mx-auto tm-section-desc">

                Uneti tekst kako bi mogli i sami da isprobate ovu šifru koju je prosto razumeti, ali teško dešifrovati. Tekst može biti bilo koje dužine. Ukoliko želite da dešifrujete Vaš šifrovan tekst možete prekopirati šifrovani tekst nazad.

              </p>

          </div>

          <form class="col-12 row">

            <div class="form-group col-6">

              <label for="input">Unesite neki tekst:</label>

              <textarea class="form-control" id="input" rows="3"></textarea>

            </div>

            <div class="form-group col-6">

              <label>Šifrovani tekst:</label>

              <textarea class="form-control" id="output" rows="3"></textarea>

            </div>

            <button type="button" id="sifruj" class="tm-intro-text tm-btn-primary col-12">Šifruj / Dešifruj</button>

          </form>

        </div>

        <div class="row">

          </div>

      </div>

    </section>

    <!-- Kontakt -->

    <section id="kontakt" class="tm-section-pad-top tm-parallax-2">

      <div class="container tm-container-contact">

        <div class="row">

            <div class="text-center col-12">

                <h2 class="tm-section-title mb-4">Kontaktirajte nas</h2>

                <p class="mb-5">

                  Ukoliko želite da saznate još o kriptografiji i različitim načinima šifrovanja, možete se prijaviti za naše besplatno predavanje ili kurs (150&euro;)

                </p><br/>

            </div>

            <!-- Forma -->

            <div class="col-sm-12 col-md-6">

              <form action="" method="get" id="formular" class="row">

              </form>

            </div>

            <!-- Forma kraj -->

            <div class="col-sm-12 col-md-6">

                <div class="contact-item">

                  <a rel="nofollow" href="Dokumentacija.pdf" target="\_blank" class="item-link">

                      <i class="far fa-2x fa-file mr-4"></i>

                      <span class="mb-0">Dokumentacija</span>

                  </a>

                </div>

                <div class="contact-item">

                  <a rel="nofollow" href="https://www.facebook.com" target="\_blank" class="item-link">

                      <i class="fab fa-2x fa-facebook mr-4"></i>

                      <span class="mb-0">Facebook</span>

                  </a>

                </div>

                <div class="contact-item">

                  <a rel="nofollow" href="https://www.google.com/maps" target="\_blank" class="item-link">

                      <i class="fas fa-2x fa-map-marker-alt mr-4"></i>

                      <span class="mb-0">Gde se nalazimo?</span>

                  </a>

                </div>

                <div class="contact-item">

                  <a rel="nofollow" href="tel:0111234567" class="item-link">

                      <i class="fas fa-2x fa-phone-square mr-4"></i>

                      <span class="mb-0">011/123-45-67</span>

                  </a>

                </div>

                <div class="contact-item">

                  <a rel="nofollow" href="sitemap.xml" target="\_blank" class="item-link">

                      <i class="fa fa-2x fa-sitemap mr-4"></i>

                      <span class="mb-0">Sitemap</span>

                  </a>

                </div>

                <div class="contact-item">

                  <a rel="nofollow" href="https://gringus0.github.io/002-portfolio-website/" target="\_blank" class="item-link">

                    <i class="fa fa-2x fa-user mr-4"></i>

                    <span class="mb-0">O Autoru</span>

                </a>

                </div>

            </div>

        </div><!-- row ending -->

      </div>

        <footer class="text-center small tm-footer">

          <p class="mb-0">

          Copyright &copy; 2022 Aleksandar Jovanović (Template by TOOPLATE) </p>

        </footer>

    </section>

    <script src="js/jquery-1.9.1.min.js"></script>

    <script src="slick/slick.min.js"></script>

    <script src="magnific-popup/jquery.magnific-popup.min.js"></script>

    <script src="js/easing.min.js"></script>

    <script src="js/jquery.singlePageNav.min.js"></script>

    <script src="js/bootstrap.min.js"></script>

    <script>

      function getOffSet(){

        var \_offset = 450;

        var windowHeight = window.innerHeight;

        if(windowHeight > 500) {

          \_offset = 400;

        }

        if(windowHeight > 680) {

          \_offset = 300

        }

        if(windowHeight > 830) {

          \_offset = 210;

        }

        return \_offset;

      }

      function setParallaxPosition($doc, multiplier, $object){

        var offset = getOffSet();

        var from\_top = $doc.scrollTop(),

          bg\_css = 'center ' +(multiplier \* from\_top - offset) + 'px';

        $object.css({"background-position" : bg\_css });

      }

      // Parallax function

      // Adapted based on https://codepen.io/roborich/pen/wpAsm

      var background\_image\_parallax = function($object, multiplier, forceSet){

        multiplier = typeof multiplier !== 'undefined' ? multiplier : 0.5;

        multiplier = 1 - multiplier;

        var $doc = $(document);

        // $object.css({"background-attatchment" : "fixed"});

        if(forceSet) {

          setParallaxPosition($doc, multiplier, $object);

        } else {

          $(window).scroll(function(){

            setParallaxPosition($doc, multiplier, $object);

          });

        }

      };

      var background\_image\_parallax\_2 = function($object, multiplier){

        multiplier = typeof multiplier !== 'undefined' ? multiplier : 0.5;

        multiplier = 1 - multiplier;

        var $doc = $(document);

        $object.css({"background-attachment" : "fixed"});

        $(window).scroll(function(){

          if($(window).width() > 768) {

            var firstTop = $object.offset().top,

                pos = $(window).scrollTop(),

                yPos = Math.round((multiplier \* (firstTop - pos)) - 186);

            var bg\_css = 'center ' + yPos + 'px';

            $object.css({"background-position" : bg\_css });

          } else {

            $object.css({"background-position" : "center" });

          }

        });

      };

      $(function(){

        // Hero Section - Background Parallax

        background\_image\_parallax($(".tm-parallax"), 0.30, false);

        background\_image\_parallax\_2($("#kontakt"), 0.80);

        background\_image\_parallax\_2($("#kvadratnaSifra"), 0.80);

        // Handle window resize

        window.addEventListener('resize', function(){

          background\_image\_parallax($(".tm-parallax"), 0.30, true);

        }, true);

        // Detect window scroll and update navbar

        $(window).scroll(function(e){

          if($(document).scrollTop() > 120) {

            $('.tm-navbar').addClass("scroll");

          } else {

            $('.tm-navbar').removeClass("scroll");

          }

        });

        // Close mobile menu after click

        $('#tmNav a').on('click', function(){

          $('.navbar-collapse').removeClass('show');

        })

        // Scroll to corresponding section with animation

        $('#tmNav').singlePageNav({

          'easing': 'easeInOutExpo',

          'speed': 600

        });

        // Add smooth scrolling to all links

        // https://www.w3schools.com/howto/howto\_css\_smooth\_scroll.asp

        $("a").on('click', function(event) {

          if (this.hash !== "") {

            event.preventDefault();

            var hash = this.hash;

            $('html, body').animate({

              scrollTop: $(hash).offset().top

            }, 600, 'easeInOutExpo', function(){

              window.location.hash = hash;

            });

          } // End if

        });

        // Pop up

        $('.tm-gallery').magnificPopup({

          delegate: 'a',

          type: 'image',

          gallery: { enabled: true }

        });

        $('.tm-testimonials-carousel').slick({

          dots: true,

          prevArrow: false,

          nextArrow: false,

          infinite: false,

          slidesToShow: 3,

          slidesToScroll: 1,

          responsive: [

            {

              breakpoint: 992,

              settings: {

                slidesToShow: 2

              }

            },

            {

              breakpoint: 768,

              settings: {

                slidesToShow: 2

              }

            },

            {

              breakpoint: 480,

              settings: {

                  slidesToShow: 1

              }

            }

          ]

        });

        // Gallery

        $('.tm-gallery').slick({

          dots: true,

          infinite: false,

          slidesToShow: 5,

          slidesToScroll: 2,

          responsive: [

          {

            breakpoint: 1199,

            settings: {

              slidesToShow: 4,

              slidesToScroll: 2

            }

          },

          {

            breakpoint: 991,

            settings: {

              slidesToShow: 3,

              slidesToScroll: 2

            }

          },

          {

            breakpoint: 767,

            settings: {

              slidesToShow: 2,

              slidesToScroll: 1

            }

          },

          {

            breakpoint: 480,

            settings: {

              slidesToShow: 1,

              slidesToScroll: 1

            }

          }

        ]

        });

      });

    </script>

    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q" crossorigin="anonymous"></script>

    <script src="js/main.js"></script>

  </body>

</html>

# CSS:

Originalni CSS ima preko 600 linija koda. Dodati CSS se nalazi na samom kraju.

/\*

Tooplate 2117 Infinite Loop

https://www.tooplate.com/view/2117-infinite-loop

\*/

@import url("https://fonts.googleapis.com/css?family=Raleway:100,300,400,500,700,900");

body {

  font-family: "Raleway", sans-serif;

  font-size: 1.2em;

  color: #707070;

  margin: 0;

  padding: 0;

  overflow-x: hidden;

}

a {

  transition: all 0.3s ease;

  color: #38B;

}

a:hover,

a:focus {

  text-decoration: none;

}

a:focus {

  outline: none;

}

.btn {

  padding: 8px 32px;

}

.btn:hover {

  background-color: #ced4da;

}

blockquote {

  font-size: 0.86em;

  line-height: 1.8em;

}

.tm-section-pad-top {

  padding-top: 80px;

  padding-bottom: 40px;

}

.tm-content-box {

  padding-top: 20px;

  padding-bottom: 40px;

}

.tm-text-primary {

  color: #37A;

}

.tm-font-big {

  font-size: 1.25rem;

}

.tm-btn-primary {

  color: white;

  background-color: #369;

  padding: 14px 30px;

}

.tm-btn-primary:hover,

.tm-btn-primary:focus {

  color: white;

  background-color: #38B;

}

/\* Navbar \*/

.tm-navbar {

  position: fixed;

  width: 100%;

  z-index: 1000;

  background-color: transparent;

  transition: all 0.3s ease;

}

.tm-navbar.scroll {

  background-color: white;

  border-bottom: 1px solid #e9ecef;

}

.navbar-brand {

  color: white;

  font-size: 1.4rem;

  font-weight: bold;

}

.navbar-brand:hover,

.tm-navbar.scroll .navbar-brand:hover {

  color: #38B;

}

.tm-navbar.scroll .navbar-brand {

  color: #369;

}

.nav-item {

  list-style: none;

}

.tm-nav-link {

  color: white;

}

.tm-navbar.scroll .tm-nav-link {

  color: #369;

}

.tm-navbar.scroll .tm-nav-link:hover,

.tm-navbar.scroll .tm-nav-link.current,

.tm-nav-link:hover {

  color: #FFF;

  background-color: #369;

}

.navbar-toggler {

  border: 1px solid white;

  padding-left: 10px;

  padding-right: 10px;

}

.navbar-toggler-icon {

  color: white;

  padding-top: 6px;

}

.tm-navbar.scroll .navbar-toggler {

  border: 1px solid #707070;

}

.tm-navbar.scroll .navbar-toggler-icon {

  color: #707070;

}

/\* Hero \*/

@media (min-height: 600px) and (min-width: 1920px) {

  #pocetak {

    background-size: cover;

  }

}

@media (min-height: 830px) {

  #pocetak {

    background-position: center -210px;

  }

}

@media (min-height: 680px) and (max-height: 829px) {

  #pocetak {

    background-position: center -300px;

  }

}

@media (min-height: 500px) and (max-height: 679px) {

  #pocetak {

    background-position: center -400px;

  }

}

@media (max-height: 499px) {

  #pocetak {

    background-position: center -450px;

  }

}

.tm-hero-text-container {

  width: 100%;

  height: 100%;

  display: flex;

  flex-flow: column;

  justify-content: center;

}

.tm-hero-text-container-inner {

  margin-top: -90px;

}

.tm-hero-title {

  font-size: 3.5rem;

  text-shadow: 2px 2px 2px #333;

}

.tm-hero-subtitle {

  text-shadow: 2px 2px 2px #333;

}

.tm-intro-next {

  position: absolute;

  bottom: 100px;

  left: 0;

  right: 0;

}

@media (max-height: 480px) {

  .tm-hero-text-container-inner {

    margin-top: -40px;

  }

  .tm-intro-next {

    bottom: 20px;

  }

}

.tm-down-arrow-link {

  display: block;

  margin-top: 18%;

}

.tm-down-arrow {

    color: #FFF;

    cursor: pointer;

    background: #357;

    padding: 15px 40px;

    transition: all 0.3s ease;

}

.tm-down-arrow:hover,

.tm-down-arrow:focus {

  color: #FFF;

  background: #37A;

  padding: 20px 50px;

}

/\* Introduction \*/

#introduction {

  padding-bottom: 100px;

}

.tm-section-title {

  font-size: 2.6rem;

  font-weight: normal;

}

.tm-intro-text {

  font-size: 1.2rem;

}

.tm-icon {

  display: block;

  color: #37A;

}

.tm-continue {

  padding: 20px 0 30px 0;

}

/\* Testimonials \*/

.tm-kvadratnaSifra-content {

  position: relative;

  z-index: 100;

}

.tm-bg-overlay {

  width: 100%;

  height: 100%;

  background: rgba(20, 70, 80, 0.2);

  position: absolute;

  top: 0;

  left: 0;

  right: 0;

  bottom: 0;

  z-index: 0;

}

.tm-testimonials-carousel {

  max-width: 1050px;

  margin: 0 auto;

}

.tm-testimonial-item {

  max-width: 290px;

  margin-left: 35px;

  margin-right: 35px;

}

.tm-testimonial-item img {

  border-radius: 50%;

  margin-bottom: 35px;

}

.tm-testimonial-item figcaption {

  text-align: right;

  font-style: italic;

  font-size: 1.1rem;

}

/\* Work \*/

.tm-section-desc {

  max-width: 650px;

  width: 100%;

  font-size: 1.2rem;

}

.tm-gallery-container {

  padding-top: 70px;

  padding-bottom: 120px;

}

.tm-gallery-item {

  margin: 0 15px;

}

.slick-dots {

  bottom: -65px;

}

.slick-dots li {

  margin: 0 13px;

}

.slick-dots li button:hover:before,

.slick-dots li button:focus:before,

.slick-dots li.slick-active button:before {

  opacity: 1;

  color: #3ba0dd;

}

.tm-testimonials-carousel .slick-dots li button:before {

  color: white;

  opacity: 0.5;

}

.tm-testimonials-carousel .slick-dots li button:hover:before,

.tm-testimonials-carousel .slick-dots li button:focus:before,

.tm-testimonials-carousel .slick-dots li.slick-active button:before {

  color: white;

  opacity: 1;

}

.slick-dots li button:before {

  font-size: 18px;

}

/\* Hover Effect \*/

/\* Common style \*/

.grid figure {

  position: relative;

  float: left;

  overflow: hidden;

  background: #3085a3;

  text-align: center;

  cursor: pointer;

}

.grid figure img {

  position: relative;

  display: block;

  min-height: 100%;

  max-width: 100%;

  opacity: 0.8;

}

.grid figure figcaption {

  padding: 2em;

  color: #fff;

  text-transform: uppercase;

  font-size: 1.25em;

  -webkit-backface-visibility: hidden;

  backface-visibility: hidden;

}

.grid figure figcaption::before,

.grid figure figcaption::after {

  pointer-events: none;

}

.grid figure figcaption,

.grid figure figcaption > a {

  position: absolute;

  top: 0;

  left: 0;

  width: 100%;

  height: 100%;

}

/\* Anchor will cover the whole item by default \*/

/\* For some effects it will show as a button \*/

.grid figure figcaption > a {

  z-index: 1000;

  text-indent: 200%;

  white-space: nowrap;

  font-size: 0;

  opacity: 0;

}

.grid figure h2 {

  word-spacing: -0.15em;

  font-size: 0.9em;

  font-weight: 300;

}

.grid figure h2 span {

  font-weight: 600;

}

.grid figure h2,

.grid figure p {

  margin: 0;

}

.grid figure p {

  letter-spacing: 1px;

  font-size: 68.5%;

}

/\*---------------\*/

/\*\*\*\*\* Honey \*\*\*\*\*/

/\*---------------\*/

figure.effect-honey {

  background: #4a3753;

  max-width: 220px;

}

figure.effect-honey img {

  opacity: 1;

  -webkit-transition: opacity 0.35s;

  transition: opacity 0.35s;

}

figure.effect-honey:hover img {

  opacity: 0.4;

}

figure.effect-honey figcaption::before {

  position: absolute;

  bottom: 0;

  left: 0;

  width: 100%;

  height: 10px;

  background: #38C;

  content: "";

  -webkit-transform: translate3d(0, 10px, 0);

  transform: translate3d(0, 10px, 0);

}

figure.effect-honey h2 {

  position: absolute;

  bottom: 0;

  left: 0;

  padding: 1em 1.5em;

  width: 100%;

  text-align: left;

  -webkit-transform: translate3d(0, -30px, 0);

  transform: translate3d(0, -30px, 0);

}

figure.effect-honey h2 i {

  font-style: normal;

  opacity: 0;

  -webkit-transition: opacity 0.35s, -webkit-transform 0.35s;

  transition: opacity 0.35s, transform 0.35s;

  -webkit-transform: translate3d(0, -30px, 0);

  transform: translate3d(0, -30px, 0);

}

figure.effect-honey figcaption::before,

figure.effect-honey h2 {

  -webkit-transition: -webkit-transform 0.35s;

  transition: transform 0.35s;

}

figure.effect-honey:hover figcaption::before,

figure.effect-honey:hover h2,

figure.effect-honey:hover h2 i {

  opacity: 1;

  -webkit-transform: translate3d(0, 0, 0);

  transform: translate3d(0, 0, 0);

}

.tm-container-gallery {

  padding-top: 30px;

  }

/\* Contact \*/

.contact-item {

  margin-left: 20px;

  margin-bottom: 50px;

}

.item-link {

  display: flex;

  align-items: center;

}

.item-link i,

.item-link span {

  color: white;

  transition: all 0.3s ease;

}

.item-link:hover i,

.item-link:hover span {

  color: #3496d8;

}

.tm-input {

  margin: 0 0 20px 0;

  width: 90%;

  padding: 8px 20px;

  border-radius: 6px;

  border: 1px solid white;

  background: transparent;

  color: white;

}

.tm-btn-submit {

  font-size: 0.9em;

  color: #fff;

  background-color: #369;

  width: 50%;

  margin-bottom: 60px;

}

.tm-btn-submit:hover {

  color: #fff;

  background-color: #38B;

}

::placeholder {

  /\* Chrome, Firefox, Opera, Safari 10.1+ \*/

  color: white;

  opacity: 1; /\* Firefox \*/

}

:-ms-input-placeholder {

  /\* Internet Explorer 10-11 \*/

  color: white;

}

::-ms-input-placeholder {

  /\* Microsoft Edge \*/

  color: white;

}

.tm-footer {

  position: absolute;

  bottom: 35px;

  left: 0;

  right: 0;

  padding: 0 15px;

}

.tm-footer a {

  color: #fff;

}

.tm-footer a:hover {

  color: #9CF;

}

.tm-footer-link {

  color: white;

}

.tm-footer-link:hover,

.tm-footer-link:focus {

  color: #38B;

  text-decoration: none;

}

p {

  line-height: 1.9;

}

@media (min-width: 768px) {

  .tm-intro-text-container {

    padding-left: 0px;

  }

  .navbar-expand-md .navbar-nav .nav-link {

    padding-right: 30px;

    padding-left: 30px;

  }

}

@media (min-width: 1200px) {

  .container {

    max-width: 1275px;

  }

  .tm-container-gallery {

    max-width: 1290px;

  }

  .tm-container-contact {

    max-width: 1063px;

  }

}

@media (max-width: 991px) {

  .tm-intro-text-container {

    padding-left: 15px;

    padding-top: 30px;

    max-width: 600px;

    margin-left: auto;

    margin-right: auto;

  }

  .tm-intro-img {

    display: block;

    margin-left: auto;

    margin-right: auto;

  }

  .tm-btn-submit {

    margin-left: 0;

    margin-top: 20px;

  }

}

@media (max-width: 767px) {

  .navbar-nav {

    max-width: 200px;

    text-align: right;

  }

  .navbar-collapse {

    background-color: rgb(255, 255, 255);

    padding: 10px;

    border-radius: 3px;

  }

  .navbar-collapse .nav-link {

    color: #707070;

  padding-right: 20px;

  }

}

@media (max-width: 480px) {

  .tm-gallery-container {

    max-width: 220px;

    margin-left: auto;

    margin-right: auto;

  }

  .tm-gallery-container-2 {

    max-width: 350px;

  }

  .slick-dots li {

    margin: 0 8px;

  }

  .tm-gallery-item {

    margin: 0;

  }

}

/\* izmenjeni css \*/

.aj-sakrij {

  display: none;

  margin: 20px;

}

.aj-prikazi {

  display: block;

}

.aj-crveni-okvir{

  border: 2px solid #ff0000;

  border-radius: 10px;

}

.upozorenje {

  font-size: 10pt;

}

.aj-kvadratna-sifra-tekst {

  font-size: 15pt;

  font-weight: 500;

}

#pocetak {

  background-color: #222;

  background-image: url(../img/main\_key.webp);

  background-repeat: no-repeat;

  background-size: 300vh;

  height: 100vh;

  min-height: 375px;

  position: relative;

  background-blend-mode: overlay;

}

#kvadratnaSifra {

  color: white;

  background-image: url(../img/sifra.png);

  background-size: cover;

  background-repeat: no-repeat;

  background-size: 1000vh;

  background-size: 100%;

  background-attachment: scroll;

  position: relative;

  background-blend-mode: overlay;

  background-color: #222;

}

@media (max-width: 991px) {

  #kvadratnaSifra {

    background-image: url(../img/sifra\_mobile.png);

  }

}

#kontakt {

  color: white;

  background-color: #001828;

  background-image: url(../img/kontakt.png);

  background-position: center;

  background-repeat: no-repeat;

  background-size: 400vh;

  min-height: 980px;

  position: relative;

  padding-bottom: 50px;

  padding-top: 100px;

  background-blend-mode: overlay;

}

.obavezna {

  font-size: 12pt;

}

#output {

  font-size: medium;

  text-align: justify;

  height: 50vh;

}

#input {

  height: 50vh;

  font-size: medium;

  text-align: justify;

}

textarea {

  resize: none;

}

# JavaScript:

// JQuery

$(document).ready(function(){

    $("#nastavak").hide();

    let parNepar = 0;

    $("#taster").click(function(e){

    e.preventDefault();

    if(parNepar % 2 == 0){

        $("#nastavak").fadeIn();

        $("#taster").html("Sakrij");

    }else{

        $("#nastavak").fadeOut();

        $("#taster").html("Saznaj više");

    }

    parNepar++;

})});

// JQuery kraj

// Navigacija

let objNavMeni = document.querySelector("#navitems");

objNavMeni.innerHTML = ``

let nizNavHrefAtt = ["#pocetak", "#staJeKriptografija", "#kvadratnaSifra", "#sifrovanje", "#kontakt"];

let nizNavImena = ["Početak", "Šta je kriptografija?", "Kvadratna šifra", "Šifrovanje i dešifrovanje", "Kontakt"]

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

    objNavMeni.innerHTML += `<li class="nav-item">

                                <a class="nav-link tm-nav-link" href="${nizNavHrefAtt[i]}">${nizNavImena[i]}</a>

                            </li>`

}

objNavMeni.innerHTML += `</ul>`

// Navigacija kraj

// Kvadratna šifra šifrovanje i dešifrovanje

document.querySelector("#sifruj").addEventListener("click", sifrujDesifruj);

function sifrujDesifruj() {

    let tbInput = ``;

    tbInput += document.querySelector("#input").value;

    let tbOutput = document.querySelector("#output");

    //prvi niz - pravljenje niza od stringa

    let nizKaraktera = tbInput.split("");

    console.log(`Niz karaktera: `);

    console.log(nizKaraktera);

    let dvodimenzioniNizIndeks = Math.floor(Math.sqrt(nizKaraktera.length));

    //drugi niz - pravljenje dvodimenzionalnog niza

    let dvodimenzionalniNiz = []

    while(nizKaraktera.length) dvodimenzionalniNiz.push(nizKaraktera.splice(0, dvodimenzioniNizIndeks));

    console.log(`Dvodimenzionalni niz karaktera`);

    console.log(dvodimenzionalniNiz);

    //treci niz - pravljenje kvadratnog dvodimenzionalnog niza

        // funkcija za pravljenje kvadratnog dvodimenzionalnog niza

    function napraviNiz(length) {

        let arr = new Array(length || 0),

            i = length;

        if (arguments.length > 1) {

            let args = Array.prototype.slice.call(arguments, 1);

            while(i--) arr[length-1 - i] = napraviNiz.apply(this, args);

        }

        return arr;

    }

    let kvadratniDvodimenzionalniNiz = napraviNiz(dvodimenzioniNizIndeks, dvodimenzioniNizIndeks);

    for (let i = 0; i < dvodimenzioniNizIndeks; i++){

        for (let j = 0; j < dvodimenzioniNizIndeks; j++){

            kvadratniDvodimenzionalniNiz[i][j] = dvodimenzionalniNiz[j][i];

        }

    }

    console.log(`Kvadratni dvodimenzionalni niz karaktera: `);

    console.log(kvadratniDvodimenzionalniNiz);

    //cetvrti niz - vracanje u jednodimenzionalni niz

    let nizSifrovanihKaraktera = kvadratniDvodimenzionalniNiz.flat(1);

    console.log(`Niz sifrovanih karaktera: `);

    console.log(nizSifrovanihKaraktera);

    //peti niz - spajanje niza u string

    let stringOutput = nizSifrovanihKaraktera.join("");

    console.log(stringOutput);

    console.log("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

    dvodimenzionalniNiz = dvodimenzionalniNiz.flat(1);

    console.log(dvodimenzionalniNiz);

    kvadratniDvodimenzionalniNiz = kvadratniDvodimenzionalniNiz.flat(1);

    console.log(kvadratniDvodimenzionalniNiz);

    let razlika = dvodimenzionalniNiz.length - kvadratniDvodimenzionalniNiz.length;

    console.log(razlika);

    // ponavljanje ukoliko je potrebno

    while (razlika > 3){

        dvodimenzionalniNiz.splice(0, Math.pow(dvodimenzioniNizIndeks, 2));

        nizKaraktera = dvodimenzionalniNiz;

        console.log(nizKaraktera);

        dvodimenzioniNizIndeks = Math.floor(Math.sqrt(nizKaraktera.length));

        console.log(dvodimenzioniNizIndeks);

        dvodimenzionalniNiz = [];

        while(nizKaraktera.length) dvodimenzionalniNiz.push(nizKaraktera.splice(0, dvodimenzioniNizIndeks));

        console.log(dvodimenzionalniNiz);

        kvadratniDvodimenzionalniNiz = napraviNiz(dvodimenzioniNizIndeks, dvodimenzioniNizIndeks);

        for (let i = 0; i < dvodimenzioniNizIndeks; i++){

            for (let j = 0; j < dvodimenzioniNizIndeks; j++){

                kvadratniDvodimenzionalniNiz[i][j] = dvodimenzionalniNiz[j][i];

            }

        }

        console.log(kvadratniDvodimenzionalniNiz);

        nizSifrovanihKaraktera = kvadratniDvodimenzionalniNiz.flat(1);

        console.log(nizSifrovanihKaraktera);

        stringOutput += nizSifrovanihKaraktera.join("");

        console.log(stringOutput);

        dvodimenzionalniNiz = dvodimenzionalniNiz.flat(1);

        console.log(dvodimenzionalniNiz);

        kvadratniDvodimenzionalniNiz = kvadratniDvodimenzionalniNiz.flat(1);

        console.log(kvadratniDvodimenzionalniNiz);

        let razlika = dvodimenzionalniNiz.length - kvadratniDvodimenzionalniNiz.length;

        console.log(razlika);

        if (razlika == 0){

            break;

        }

    }

    dvodimenzionalniNiz.splice(0, Math.pow(dvodimenzioniNizIndeks, 2));

    dvodimenzionalniNiz = dvodimenzionalniNiz.join("");

    stringOutput += dvodimenzionalniNiz;

    console.log(`String sifrovanih karaktera`);

    console.log(stringOutput);

    console.log(stringOutput.length);

    tbOutput.value = stringOutput;

}

// Kvadratna šifra šifrovanje i dešifrovanje kraj

// Tajmer

let kontaktSekcija = document.querySelector(`#kontakt`);

let tajmer = document.createElement(`h2`);

tajmer.setAttribute(`id`, `tajmer`);

tajmer.classList.add(`text-center`);

tajmer.classList.add(`mb-4`);

let vremeSledecegPredavanja = new Date(`Jan 15, 2023 08:00:00`).getTime();

let preostaloVreme = setInterval(function(){

    let danasnjeVreme = new Date().getTime();

    let razlika = vremeSledecegPredavanja - danasnjeVreme;

    let preostaloDana = Math.floor(razlika / 1000 / 60 / 60 / 24);

    let preostaloSati = Math.floor((razlika % (1000 \* 60 \* 60 \* 24)) / (1000 \* 60 \* 60));

    let preostaloMinuta = Math.floor((razlika % (1000 \* 60 \* 60)) / (1000 \* 60));

    let preostaloSekundi = Math.floor((razlika % (1000 \* 60)) / 1000);

    tajmer.innerHTML = `Preostalo vremena do sledeceg predavanja: ${preostaloDana}d ${preostaloSati}h ${preostaloMinuta}m ${preostaloSekundi}s`;

})

kontaktSekcija.insertBefore(tajmer, kontaktSekcija.firstChild);

// Tajmer kraj

// Forma

    // Dinamicko pravljenje i ispisivanje forme

let objForma = document.querySelector("#formular");

let forma = document.querySelector("#formular");

let formaInputId = ["ime", "prezime", "email"];

let formaInputImena = ["name", "name", "email"];

let formaInputType = ["text", "text", "email"];

let formaInputPlaceholder = ["Vaše ime\*", "Vaše prezime\*", "Vaš e-mail\*"];

let inputKlase = ["tm-input", "mb-1"];

let spanKlase = ["col-11", "upozorenje", "mb-3", "aj-sakrij"];

let spanTekst = ["Ime nije u dobrom formatu. Dozvoljena su dva imena. Primer: Aleksandar Radivoje", "Prezime nije u dobrom formatu. Dozvoljena su dva prezimena. Primer: Pavlović Jovanović", "Email nije u dobrom formatu. Primer: johndoe@email.com"];

let spanObavezno = document.createElement("span");

spanObavezno.setAttribute("id", "obavezno");

spanObavezno.classList.add("col-6");

spanObavezno.classList.add("obavezna");

spanObavezno.innerHTML = "\*obavezna polja";

forma.appendChild(spanObavezno);

for (let i in formaInputId){

    let inputTag = document.createElement(`input`);

    inputTag.setAttribute(`id`, `${formaInputId[i]}`);

    inputTag.setAttribute(`name`, `${formaInputImena[i]}`);

    inputTag.setAttribute(`type`, `${formaInputType[i]}`);

    inputTag.setAttribute(`placeholder`, `${formaInputPlaceholder[i]}`);

    for (let i in inputKlase){

        inputTag.classList.add(`${inputKlase[i]}`);

    }

    inputTag.setAttribute(`required`, `required`);

    forma.appendChild(inputTag);

    let spanTag = document.createElement(`span`);

    for (let i in spanKlase){

        spanTag.classList.add(`${spanKlase[i]}`);

    }

    forma.appendChild(spanTag);

}

let textareaPitanja = document.createElement(`textarea`);

textareaPitanja.setAttribute(`id`, `pitanja`);

textareaPitanja.setAttribute(`name`, `pitanja`);

textareaPitanja.setAttribute(`rows`, `8`);

textareaPitanja.setAttribute(`placeholder`, `Pitanja?`);

textareaPitanja.classList.add(`tm-input`);

forma.appendChild(textareaPitanja);

let radioNaslov = document.createElement(`label`);

radioNaslov.classList.add(`col-12`);

radioNaslov.innerHTML = `Izaberite za šta se prijavljujete\*:`;

forma.appendChild(radioNaslov);

let radioForIId = [`rdKurs`, `rdPred`];

let radioValue = [`K`, `P`];

let radioTekst = [`Kurs`, `Predavanje`];

let radioDiv = document.createElement(`div`);

forma.appendChild(radioDiv)

let spanVrstaPrijave = document.createElement(`span`);

spanVrstaPrijave.setAttribute(`id`, `vrstaPrijave`);

spanVrstaPrijave.classList.add(`coll-11`);

spanVrstaPrijave.classList.add(`aj-sakrij`);

spanVrstaPrijave.innerHTML = `Morate izabrati ili predavanje ili kurs!`

radioDiv.appendChild(spanVrstaPrijave);

for (let i in radioTekst){

    let prviDiv = document.createElement(`div`);

    let drugiDiv = document.createElement(`div`);

    drugiDiv.classList.add(`form-check-inline`);

    drugiDiv.classList.add(`mb-2`);

    prviDiv.appendChild(drugiDiv);

    let radioLabel = document.createElement(`label`);

    radioLabel.classList.add(`form-check-label`);

    radioLabel.setAttribute(`for`, `${radioForIId[i]}`);

    radioLabel.innerHTML = `${radioTekst[i]}`

    let radioInput = document.createElement(`input`);

    radioInput.setAttribute(`type`, `radio`);

    radioInput.setAttribute(`id`, `${radioForIId[i]}`);

    radioInput.setAttribute(`name`, `tipPrijave`);

    radioInput.setAttribute(`value`, `${radioValue[i]}`);

    radioInput.classList.add(`form-check-input`);

    radioInput.innerHTML = `${radioTekst[i]}`

    drugiDiv.appendChild(radioInput);

    drugiDiv.appendChild(radioLabel);

    radioDiv.appendChild(prviDiv);

}

let prviDiv = document.createElement(`div`);

prviDiv.classList.add(`form-check`);

let chkLabel = document.createElement(`label`);

chkLabel.classList.add(`form-check-label`);

chkLabel.setAttribute(`for`, `chbObavest`);

chkLabel.innerHTML = `Obeležiti ukoliko želite da primate obaveštenja o našim predavanjima i kursevima putem e-mail pošte (opciono).`;

let chkInput = document.createElement(`input`);

chkInput.setAttribute(`type`, `checkbox`);

chkInput.setAttribute(`id`, `chbObavest`);

chkInput.setAttribute(`value`, `Obavestenja`);

chkInput.classList.add(`form-check-input`);

prviDiv.appendChild(chkInput);

prviDiv.appendChild(chkLabel);

forma.appendChild(prviDiv);

let tasterPrijava = document.createElement(`button`);

tasterPrijava.setAttribute(`type`, `button`);

tasterPrijava.setAttribute(`id`, `prijava`);

let tasterPrijavaKlase = [`tm-intro-text`, `tm-btn-primary`, `mt-3`, `mb-5`, `col-6`];

for (let i in tasterPrijavaKlase){

    tasterPrijava.classList.add(`${tasterPrijavaKlase[i]}`);

}

tasterPrijava.innerHTML = `Prijavi se`;

forma.appendChild(tasterPrijava);

    //Provera Forme

let prijavaButton = document.querySelector("#prijava");

prijavaButton.addEventListener("click", proveraForma);

let objIme = document.querySelector("#ime");

let objPrezime = document.querySelector("#prezime");

let objEmail = document.querySelector("#email");

let nizTipPrijave = document.getElementsByName("tipPrijave");

let regExImeIPrezime = /^([A-ZŠĐČĆŽ][a-zšđčćž]{2,19})(\s[A-ZŠĐČĆŽ][a-zšđčćž)]{2,19})?$/;

let regExEmail = /^(([^<>()[\]\\.,;:\s@"]+(\.[^<>()[\]\\.,;:\s@"]+)\*)|(".+"))@((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\])|(([a-zA-Z\-0-9]+\.)+[a-zA-Z]{2,}))$/;

blurProvera(objIme, regExImeIPrezime, "Ime nije u dobrom formatu. Dozvoljena su dva imena. Primer: Aleksandar Radivoje");

blurProvera(objPrezime, regExImeIPrezime, "Prezime nije u dobrom formatu. Dozvoljena su dva prezimena. Primer: Pavlović Jovanović");

blurProvera(objEmail, regExEmail, "Email nije u dobrom formatu. Primer: johndoe@email.com");

function proveraForma() {

    proveraRegEx(regExImeIPrezime, objIme, "Ime nije u dobrom formatu. Dozvoljena su dva imena. Primer: Aleksandar Radivoje");

    proveraRegEx(regExImeIPrezime, objPrezime, "Prezime nije u dobrom formatu. Dozvoljena su dva prezimena. Primer: Pavlović Jovanović");

    proveraRegEx(regExEmail, objEmail, "Email nije u dobrom formatu. Primer: johndoe@email.com")

    let tipPrijave = "";

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

        if (nizTipPrijave[i].checked){

            tipPrijave = nizTipPrijave[i].value;

        }

    }

    if (tipPrijave == ""){

        nizTipPrijave[0].parentElement.parentElement.previousElementSibling.classList.remove("aj-sakrij");

        nizTipPrijave[0].parentElement.parentElement.parentElement.classList.add("aj-crveni-okvir");

    }

    if (tipPrijave != ""){

        nizTipPrijave[0].parentElement.parentElement.previousElementSibling.classList.add("aj-sakrij");

        nizTipPrijave[0].parentElement.parentElement.parentElement.classList.remove("aj-crveni-okvir");

    }

}

function blurProvera(obj, regEx, poruka) {

    obj.addEventListener("blur", function(){

        proveraRegEx(regEx, obj, poruka)

    })

}

function proveraRegEx(re, objekat, poruka){

    if(!re.test(objekat.value)){

        objekat.nextElementSibling.classList.remove("aj-sakrij");

        objekat.nextElementSibling.innerHTML = poruka;

        objekat.classList.add("aj-crveni-okvir");

    }

    else{

        objekat.nextElementSibling.classList.add("aj-sakrij");

        objekat.nextElementSibling.innerHTML = "";

        objekat.classList.remove("aj-crveni-okvir");

    }

}

// Forma - kraj