NEWS TRACKER APPPLICATION

USING CLOUD

A Project report submitted in partial fulfilment of 7th semester in degree
of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

Team ID: PNT2022TMID40663
SIVA. A   610919104082
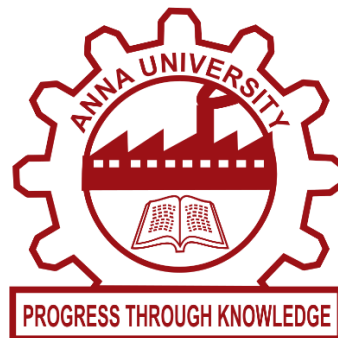SAKTHI.V   610919104070
RAJ GANESH V   610919104069

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
JAYALAKSHMI INSTITUTE OF TECHNOLOGY, THOPPUR

ANNA UNIVERSITY: CHENNAI 600025

NOV-2022

JAYALAKSHMI INSTITUTE OF TECHNOLOGY, THOPPUR
(A Constituent College of Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report "**NEWS TRACKER APPLICATION"** is the bonafide record work done by **SIVA.A**(**610919104082**),**SAKTHI.V** (**610919104070**),**RAJGANESH.V**(**610919104069**),for **IBM- NALAIYATHIRAN** in **VII** semester of **B.E.,** degree course **in Computer Science and Engineering** branch during the academic year of 2022 - 2023.

Staff-In charge   **Evaluator**
 A.SUMATHI .ME   **G.GEETHA.ME**

Head of the Department
MR SATHYA.ME

ACKNOWLEDGEMENT

We express our breathless thanks to our **Dr.VENKATESAN.ME., Ph.D.,** the PRINCIPAL(i\c), JAYALAKSHMI INSTITUTE OF TECHNOLOGY, THOPPUR, for giving constant motivation in succeeding in our goal.

We acknowledge our sincere thanks to Head of the Department (i/c)

MR SATHYA.ME for giving us valuable suggestion and help towards us throughout this Project.

We are highly grateful to thank our Project coordinator **A SUMATHI.ME.,**   and our Project Evaluator **G GEETHA M.E.,** Department of Computer Science and Engineering, JAYALAKSHMI INSTITUTE OF TECHNOLOGY , THOPPUR, for the coordinating us throughout this Project.

We are very much indebted to thank all the faculty members of Department of Computer science and Engineering in our Institute, for their excellent moral support and suggestions to complete our Project work successfully.

Finally our acknowledgment does our parents, sisters and friends those who had extended their excellent support and ideas to make our Project a pledge one.

<div align="right">

SIVA A
SAKTHI V
RAJGANESH V

</div>

ABSTRACT

Everyone has the right to freedom of speech. However, this right is being misused to differentiate and attack others, physically or verbally, in the name of free speech. This discrimination is known as hate speech. Hate speech can be well-defined as language used to express hate towards a person or a group of people based on characteristics such as race, religion, ethnicity, gender, nationality, disability and sexual orientation. The increasing usage of social sites and information sharing has specified major benefits to humanity. However, this has also assumed rise to a variety of challenges including the spreading and sharing of hate speech messages. Thus, to solve this emerging issue in social media sites, recent studies employed a variety of machine learning and deep learning algorithms with text mining algorithm to automatically detect the hate speech messages on real time datasets. Hence, the aim of this Project is to analyses the comments on social networks using Natural Language processing technique (NLP) and Deep learning algorithm named as Back propagation neural network algorithm. Using NLP technique, can extract the keywords from user generated content and implement Back Propagation neural network to classify the text whether it is positive or negative. If it is negative means, automatically block the comments as per user wish and also block the friends based on pre-defined threshold values. Experimental results shows that the proposed framework implemented in real time social network site with improved notification .

TABLE OF FIGURES

News Tracker Application

1.INTRODUCTION

Mobile app ecosystems are transforming patterns of news consumption. Until quite recently, reading the news was a niche use for smartphones [12], mostly for when users were 'on the go'; now however, two in every three users of mobile devices in the US regularly access news and as many as one in five read in-depth news articles daily [2]; a similar picture is found in the UK [1]. This growth in mobile news access continues the migration of news consumers to the Internet.

1.1Project overview

As news is increasingly accessed on smartphones and tablets, the need for personalising news app interactions is apparent. We report a series of three studies addressing key issues in the development of adaptive news app interfaces. We first surveyed users' news reading preferences and behaviours; analysis revealed three primary types of reader. We then implemented and deployed an Android news app that logs users' interactions with the app. We used the logs to train a classifier and showed that it is able to reliably recognise a user according to their reader type. Finally we evaluated alternative, adaptive user interfaces for each reader type. The evaluation demonstrates the differential benefit of the adaptation for different users of the news app and the feasibility of adaptive interfaces for news apps.

1.2Purpose

As our lives are very busy these days, we often feel we need more than 24 hrs. a day to cope up with everything we have in our schedule. Well, that's not possible but reducing the time by changing the conventional method of reading news can help. Just tell us what market news you're interested in and get a quick peek for the day. Only read what you feel is relevant and save your time. This app helps you to query for all information about Indices, Commodities, Currencies, Future Rates, Bonds, etc.… as on official websites.

2. LITERATURE SURVEY

2.1 Existing problem

we often feel we need more than 24 hrs. a day to cope up with everything we have in our schedule. Well, that's not possible but reducing the time by changing the conventional method of reading news can help.

2.2 References

Like articles, websites, blogs. https://www.researchgate.net/.

https://techvidvan.com/tutorials/android-news-app-project-source-code/

https://www.ripublication.com/

2.3 Problem Statement Definition

1.The user needs a way to organize the news on daily basis so that he can cope up with daily events in his tight schedule.

2.The user wants to read news only about particular topics so that he can be informed about his interest.

3.The user wants to get informed from only certified news outlets .

4.The user needs a way to search about the news on topics he wants to.

5.The user wants to know the news about his surrounding using GPS location.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

Empathy Map

What does he THINK AND FEEL?
- what really counts
- major preoccupations
- worries & aspirations

What does he HEAR?
- what friends say
- what boss say
- what influencers say

What does he SEE?
- enviroment
- friends
- what the market offers

What does he SAY AND DO?
- attitude in public
- appearance
- behavior towards others

Pain
- fears
- frustrations
- obstacles

Gain
- "wants" / needs
- measures of success
- obstacles

Business Model **Toolbox**

3.2 Ideation & Brainstorming
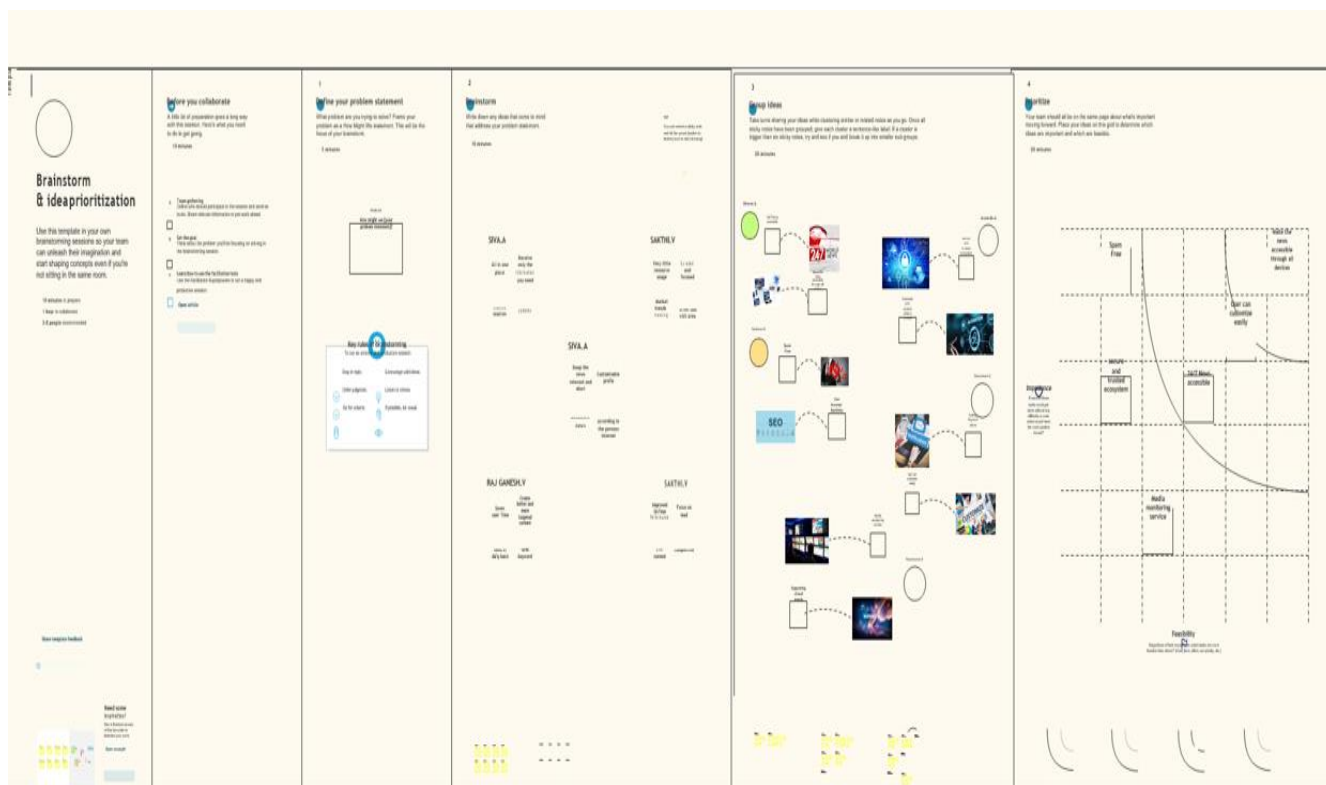
Proposed Solution :

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Most people don't like to carry a newspaper with them. Some people want them to be updated only in the area they are interested in.
2.	Idea / Solution description	An application needs to be developed in which users can read news whenever they want and they will be able to customize their area of interest. So that they will be notified, if any new news is updated in their interested areas.
3.	Novelty / Uniqueness	1. A user can read news only from their interested fields rather than reading all the news. 2. This application provides users with a trusted and secured ecosystem. News shared through the application is original and spam free.
4.	Social Impact / Customer Satisfaction	This application encourages its users to provide feedback. Based on that feedback, developments were made eventually.

5.	Business Model (Revenue Model)	1. Add advertisements to the application, so that we can get revenue from those advertisement-sponsored organizations. More advertisements may irritate the user. 2. Add premium subscription, users who subscribe for premium won't get advertisements.
6.	Scalability of the Solution	As it was an application-based project, correct ideation and execution can develop an application with no bugs and errors, so that the user might like our application and some might suggest and share it to their surroundings, resulting in an increase in our application insights.

PROBLEM SOLUTION FIT

Problem-Solution fit canvas 2.0

Purpose / Vision

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) — CS
Who is your customer?
i.e. working parents of 0-5 y.o. kids

6. CUSTOMER CONSTRAINTS — CC
What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

5. AVAILABLE SOLUTIONS — AS
Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS — J&P
Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

9. PROBLEM ROOT CAUSE — RC
What is the real reason that this problem exists?
What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

7. BEHAVIOUR — BE
What does your customer do to address the problem and get the job done? i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

Focus on J&P, tap into BE, understand RC

Identify strong TR &

the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

Making separate space for each category of news.people select the news category and know all news about that.

In online people know news faster through network

4. EMOTIONS: BEFORE / AFTER — EM
How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure > confident, in control - use it in your communication strategy & design.

Reading about a more efficient solution in the news

People will know the news in faster

8.2 OFFLINE
What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

In offline people must allocate time for reading newspaper

online & offline CH of BE

4. REQUIREMENT ANALYSIS

Functional Requirements:

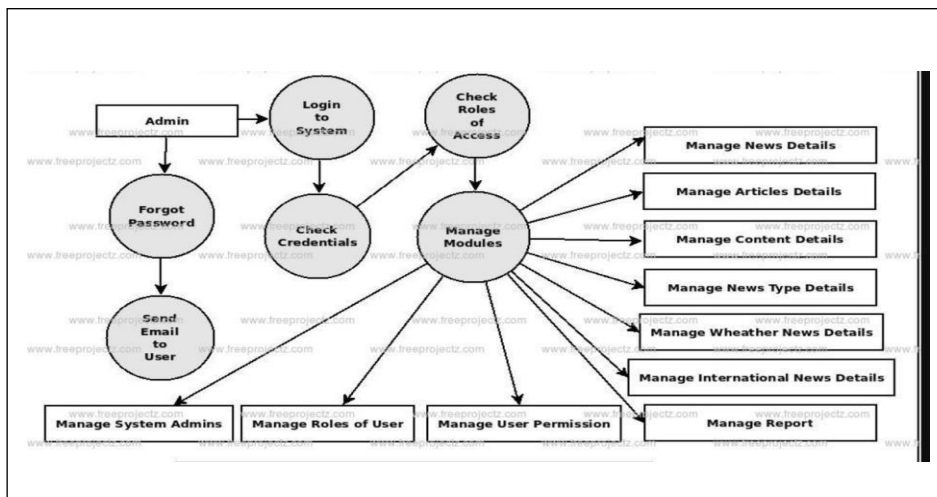| FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through online application Registration through Gmail Registration through website
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User login	Login through browser directly by entering username and password Login through Login through email
FR-4	User interaction	Done through user interface between client and server View the related news by subscripted or requested page

4.1 Non-functional Requirements:

5. Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	End users can receive push updates for new content on a site by subscribing to the site's news feed
NFR-2	**Security**	How well are the system and its data protected against attacks
NFR-3	**Reliability**	How often does the system experience critical failures? How much time does it take to fix the issue when it arises ?And how is user availability time compared to downtime?
NFR-4	**Performance**	Performance is the core non-functional requirements no system can do without.It defines how fast a software system or a particular piece of it responds to certain users actions under a certain workload. In most cases, this metric explains how long a user must wait before the target operation happens (the page renders, a transaction is processed, etc.) given the overall number of users at the moment. But it's not always like that. Performance requirements may describe background

		processes invisible to users, e.g. backup. But let's focus on user-centric performance.
NFR-5	**Availability**	Availability describes how likely the system is accessible to a user at a given point in time. While it can be expressed as an expected percentage of successful requests, you may also define it as a percentage of time the system is accessible for operation during some time period. For instance, the system may be available 98 percent of the time during a month. Availability is perhaps the most [business-critical requirement,](#) but to define it, you also must have estimations for reliability and maintainability.
NFR-6	**Scalability**	Scalability assesses the highest workloads under which the system will still meet the performance requirements. There are two ways to enable your system scale as the workloads get higher: horizontal and vertical scaling.

6. PROJECT DESIGN

6.1 Data Flow Diagrams

6.2 Solution & Technical Architecture
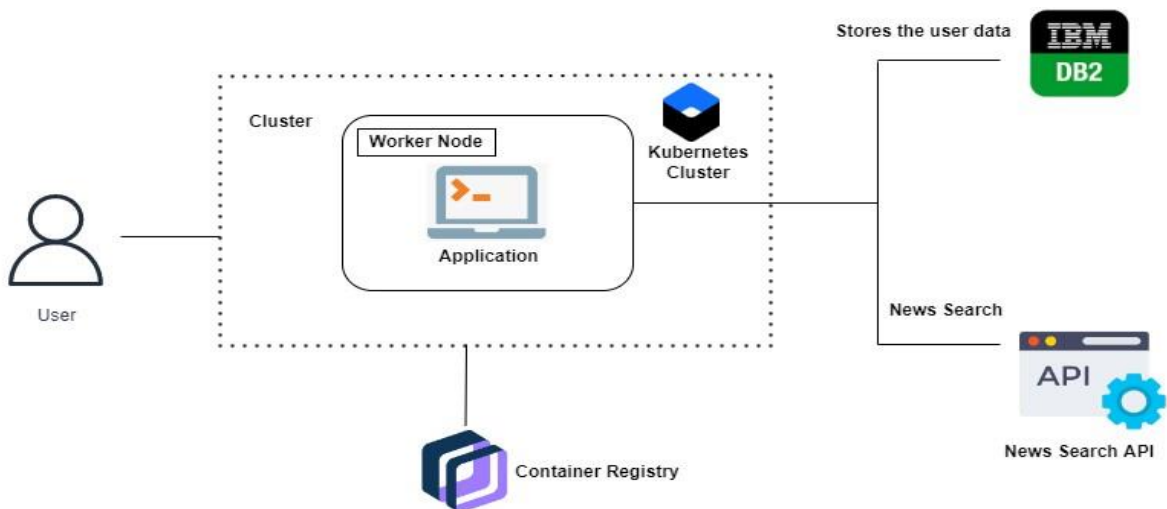
6.3 User Stories

User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through QR code	I can register & access the dashboard by scanning the QR code	Low	Sprint-2
		USN-4	As a user, I can register for the application through google	I can create a new password	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	If I forgot the password, then I will receive a mail for creating another password or alternate login credentials.	High	Sprint-1
	Dashboard					
Customer (Web user)						
Customer Care Executive						
Administrator						

7. PROJECT PLANNING & SCHEDULING

7.1 Sprint Planning & Estimation

Project Planning Phase

Project Planning Template (Product Backlog, Sprint Planning, Stories, Story points)

Date	15 November 2022
Team ID	PNT2022TMID40663
Project Name	Project – News Tracker Application
Maximum Marks	8 Marks

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Use the below template to create product backlog and sprint schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Siva,sakthi
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Raj ganesh,sakthi
Sprint-1		USN-3	As a user, I can register for the application through Facebook	2	Low	siva
Sprint-1		USN-4	As a user, I can register for the application through Gmail	2	Medium	Raj ganesh
Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	1	High	Siva,sakthi
Sprint-2	Dashboard	USN-6	As a user I should be able to navigate and access all the features hassle free	2	High	Siva,sakthi
Sprint-2	Layout	USN-7	As a user I should be able to access the portal with different devices with the same comfort	2	High	Siva, Raj ganesh
Sprint-3	Data Store and retrieval	USN-8	Get Data from API and store as JSON in DB2	3	High	siva,Sakthi,Raj ganesh
Sprint-3		USN-9	Get bin data from API and store in DFS	2	High	Siva,Raj ganesh

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-4	User Segregation and data access	USN-10	As a CC executive I should be able to uniquely identify the customer and offer help	1	Low	sakthi
Sprint-4	Change code	USN-11	As a administrator I should be able to modify code according to the future requirements.	2	Medium	Siva,Sakthi
Sprint-4	Monitor the system	USN-12	As a administrator I should be able to monitor the cloud system and fix errors before customer.	1	High	Siva,Sakthi,Raj ganesh

Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	8	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	4	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	5	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	4	19 Nov 2022

Reports from JIRA

----

2. CODING & SOLUTIONING

Forgot password-

Forpass.html

```html
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap contributors">
    <meta name="generator" content="Hugo 0.84.0">
    <title>Sign In</title>


    <link rel="canonical" href="https://getbootstrap.com/docs/5.0/examples/sign-in/">


    <link href="https://getbootstrap.com/docs/5.0/assets/css/docs.css" rel="stylesheet">


    <!-- Bootstrap core CSS -->
    <link           href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+fzT"
crossorigin="anonymous">
    <!-- Favicons -->
<link     rel="apple-touch-icon"     href="/docs/5.0/assets/img/favicons/apple-touch-icon.png"
sizes="180x180">
<link    rel="icon"    href="/docs/5.0/assets/img/favicons/favicon-32x32.png"    sizes="32x32"
type="image/png">
<link    rel="icon"    href="/docs/5.0/assets/img/favicons/favicon-16x16.png"    sizes="16x16"
type="image/png">
<link rel="manifest" href="/docs/5.0/assets/img/favicons/manifest.json">
```

```html
<link          rel="mask-icon"          href="/docs/5.0/assets/img/favicons/safari-pinned-tab.svg"
color="#7952b3">
<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon.ico">
<meta name="theme-color" content="#7952b3">


    <style>
      .bd-placeholder-img {
font-size: 1.125rem;          text-
anchor: middle;
        -webkit-user-select: none;          -
moz-user-select: none;          user-select:
none;
      }

      @media (min-width: 768px) {
.bd-placeholder-img-lg {          font-
size: 3.5rem;
        }
      }
    </style>


    <!-- Custom styles for this template -->
    <link       href="static/sheets/signin.css"       rel="stylesheet">
<link href="static/sheets/colors.css" rel="stylesheet">
  </head>
  <body class="text-center">
      <nav class="navbar navbar-dark bg-dark fixed-top">
        <div class="container-fluid">
          <a class="navbar-brand" href="#">News Tracker</a>
```

```html
<button class="navbar-toggler" type="button" data-bs-toggle="offcanvas" data-bstarget="#offcanvasDarkNavbar" aria-controls="offcanvasDarkNavbar">
    <span class="navbar-toggler-icon"></span>
</button>
<div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1" id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">
    <div class="offcanvas-header">
        <h5 class="offcanvas-title" id="offcanvasDarkNavbarLabel">Profile</h5>
        <button type="button" class="btn-close btn-close-white" data-bsdismiss="offcanvas" aria-label="Close"></button>
    </div>
    <div class="offcanvas-body">
        <ul class="navbar-nav justify-content-end flex-grow-1 pe-3">
            <li class="nav-item">
                <a class="nav-link active" aria-current="page" href="home.html">Home</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="base.html">Fetch News</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="about.html">About Us</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="signin.html">Sign In</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="signup.html">Sign Up</a>
            </li>
        </div>
    </div>
</div>
</nav>
```

```html
<main class="form-signin ">
 <form action = "{{ url_for("getUser")}}" method="POST">
   <img    class="mb-4"    src="static/Images/house-user-solid.svg"    alt=""    width="72"
height="57">
   <h1 class="h3 mb-3 fw-normal white">Change your password</h1>


   <div class="form-floating">
    <input    type="email" class="form-control" id="floatingInput"    name =
"uname" placeholder="name@example.com">
    <label for="floatingInput">Email address</label>
   </div>
   <br>
   <button class="w-100 btn btn-lg btn-warning btn-outline-warning" type="submit"><a
href="static/templates/changepass.html">Change Password</a></button><br><br>
 </form>
</main>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-
u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8"
crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/@docsearch/js@3"></script>

<script src="https://cdn.jsdelivr.net/npm/@stackblitz/sdk@1/bundles/sdk.umd.js"></script>


<script src="/docs/5.2/assets/js/docs.min.js"></script>


<script>


 document.querySelectorAll('.btn-edit').forEach(btn                =>                {

btn.addEventListener('click', event => {

   const            htmlSnippet            =            event.target.closest('.bd-code-
snippet').querySelector('.bdexample').innerHTML
```

```
    const              classes              =              Array.from(event.target.closest('.bd-code-
snippet').querySelector('.bdexample').classList).join(' ')


    const       jsSnippet       =       event.target.closest('.bd-code-
snippet').querySelector('.btnedit').getAttribute('data-sb-js-snippet')

    StackBlitzSDK.openBootstrapSnippet(htmlSnippet, jsSnippet, classes)

  })

 })


 StackBlitzSDK.openBootstrapSnippet = (htmlSnippet, jsSnippet, classes) => {

const markup = `<!doctype html>

<html lang="en">

  <head>

    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width, initial-scale=1">

    <link    href="https:\/\/cdn.jsdelivr.net\/npm\/bootstrap@5.2.1\/dist\/css\/bootstrap.min.css"
rel="stylesheet">

    <link href="https://getbootstrap.com/docs/5.2/assets/css/docs.css" rel="stylesheet">

    <title>Bootstrap Example</title>

    <${'script'}
src="https:\/\/cdn\.jsdelivr\.net\/npm\/bootstrap@5\.2\.1\/dist\/js\/bootstrap\.bundle\.min\.js">
</${'script'}>

  </head>

  <body class="p-3 m-0 border-0 ${classes}">


    <!-- Example Code -->

${htmlSnippet.replace(/^/gm, '    ')}

    <!-- End Example Code -->

  </body>

</html>`
```

const jsSnippetContent = jsSnippet ? '\/\/ NOTICE!!! Initially embedded in our docs this JavaScript\n\/\/ file contains elements that can help you create reproducible\n\/\/ use cases in StackBlitz for instance\.\n\/\/ In a real project please adapt this content to your needs\.\n\/\/ \u002b\n\n\/*!\n * JavaScript for Bootstrap\u0027s docs \(https:\/\/getbootstrap\.com\/\)\n * Copyright 2011\-2022 The Bootstrap Authors\n * Copyright 2011\-2022 Twitter, Inc\.\n * Licensed under the Creative Commons Attribution 3\.0 Unported License\.\n * For details, see https:\/\/creativecommons\.org\/licenses\/by\/3\.0\/\.\n *\/\n\n\/* global bootstrap: false *\/\n\n\(\(\) =\u003e \{\n \u0027use strict\u0027\n\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Tooltips\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Instantiate all tooltips in a docs or StackBlitz page\n document\.querySelectorAll\(\u0027\[data\-bs\-toggle=\u0022tooltip\u0022\]\u0027\)\n \.forEach\(tooltip =\u003e \{\n new bootstrap\.Tooltip\(tooltip\)\n \}\)\n\n \/\/ \-\-\-\-\-\-\-\-\\-\n \/\/ Popovers\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Instantiate all popovers in a docs or StackBlitz page\n document\.querySelectorAll\(\u0027\[data\-bs\-toggle=\u0022popover\u0022\]\u0027\)\n \.forEach\(popover =\u003e \{\n new bootstrap\.Popover\(popover\)\n \}\)\n\n \/\/ \-\\n \/\/ Toasts\n \/\/ \-\n \/\/ Used by \u0027Placement\u0027 example in docs or StackBlitz\n const toastPlacement = document\.getElementById\(\u0027toastPlacement\u0027\)\n if \(toastPlacement\) \{\n document\.getElementById\(\u0027selectToastPlacement\u0027\)\.addEventListener\(\u0027change\u0027, function \(\) \{\n if \(!toastPlacement\.dataset\.originalClass\) \{\n toastPlacement\.dataset\.originalClass = toastPlacement\.className\n \}\n\n toastPlacement\.className = `\$\{toastPlacement\.dataset\.originalClass\} \$\{this\.value\}`\n \}\)\n \}\n\n \/\/ Instantiate all toasts in a docs page only\n document\.querySelectorAll\(\u0027\.bd\-example \.toast\u0027\)\n \.forEach\(toastNode =\u003e \{\n const toast = new bootstrap\.Toast\(toastNode, \{\n autohide: false\n \}\)\n\n toast\.show\(\)\n \}\)\n\n \/\/ Instantiate all toasts in a docs page only\n const toastTrigger = document\.getElementById\(\u0027liveToastBtn\u0027\)\n const toastLiveExample = document\.getElementById\(\u0027liveToast\u0027\)\n if \(toastTrigger\) \{\n toastTrigger\.addEventListener\(\u0027click\u0027, \(\) =\u003e \{\n const toast = new bootstrap\.Toast\(toastLiveExample\)\n\n toast\.show\(\)\n \}\)\n \}\n\n \/\/ \-\n \/\/ Alerts\n \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n \/\/ Used in \u0027Show live toast\u0027 example in docs or StackBlitz\n const alertPlaceholder = document\.getElementById\(\u0027liveAlertPlaceholder\u0027\)\n const alertTrigger = document\.getElementById\(\u0027liveAlertBtn\u0027\)\n\n const appendAlert = \(message, type\) =\u003e \{\n const wrapper = document\.createElement\(\u0027div\u0027\)\n wrapper\.innerHTML = \[\n `\u003cdiv class=\u0022alert alert\-\$\{type\} alert\-dismissible\u0022 role=\u0022alert\u0022\u003e`,\n ` \u003cdiv\u003e\$\{message\}\u003c\/div\u003e`,\n \u0027 \u003cbutton type=\u0022button\u0022 class=\u0022btn\-close\u0022 data\-bs\-dismiss=\u0022alert\u0022 aria\-label=\u0022Close\u0022\u003e\u003c\/button\u003e\u0027,\n

\u0027\u003c\/div\u003e\u0027\n \]\.join\(\u0027\u0027\)\n\n alertPlaceholder\.append\(wrapper\)\n \}\n\n if \(alertTrigger\) \{\n alertTrigger\.addEventListener\(\u0027click\u0027, \(\) =\u003e \{\n appendAlert\(\u0027Nice, you triggered this alert message!\u0027, \u0027success\u0027\)\n \}\)\n \}\n\n \/\/ \-\n \/\/ Checks \u0026 Radios\n \/\/ \-\n \/\/ Indeterminate checkbox example in docs and StackBlitz\n document\.querySelectorAll\(\u0027\.bd\-example\-indeterminate \[type=\u0022checkbox\u0022\]\u0027\)\n \.forEach\(checkbox =\u003e \{\n if \(checkbox\.id\.includes\(\u0027Indeterminate\u0027\)\) \{\n checkbox\.indeterminate = true\n \}\n \}\)\n\n \/\/ \-\n \/\/ Links\n \/\/ \-\n \/\/ Disable empty links in docs examples only\n document\.querySelectorAll\(\u0027\.bd\-content \[href=\u0022#\u0022\]\u0027\)\n \.forEach\(link =\u003e \{\n link\.addEventListener\(\u0027click\u0027, event =\u003e \{\n event\.preventDefault\(\)\n \}\)\n \}\)\n\n \/\/ \-\\-\-\-\n \/\/ Modal\n \/\/ \-\n \/\/ Modal \u0027Varying modal content\u0027 example in docs and StackBlitz\n const exampleModal = document\.getElementById\(\u0027exampleModal\u0027\)\n if \(exampleModal\) \{\n exampleModal\.addEventListener\(\u0027show\.bs\.modal\u0027, event =\u003e \{\n \/\/ Button that triggered the modal\n const button = event\.relatedTarget\n \/\/ Extract info from data\-bs\-* attributes\n const recipient = button\.getAttribute\(\u0027data\-bs\whatever\u0027\)\n\n \/\/ Update the modal\u0027s content\.\n const modalTitle = exampleModal\.querySelector\(\u0027\.modal\-title\u0027\)\n const modalBodyInput = exampleModal\.querySelector\(\u0027\.modal\-body input\u0027\)\n\n modalTitle\.textContent = `New message to \$\{recipient\}`\n modalBodyInput\.value = recipient\n \}\)\n \}\n\n \/\/ \-\n \/\/ Offcanvas\n \/\/ \-\n \/\/ \u0027Offcanvas components\u0027 example in docs only\n const myOffcanvas = document\.querySelectorAll\(\u0027\.bd\-example\-offcanvas \.offcanvas\u0027\)\n if \(myOffcanvas\) \{\n myOffcanvas\.forEach\(offcanvas =\u003e \{\n offcanvas\.addEventListener\(\u0027show\.bs\.offcanvas\u0027, event =\u003e \{\n event\.preventDefault\(\)\n \}, false\)\n \}\)\n \}\n\}\)\(\)\n' : null const project = {

files: {

   'index.html': markup,

   'index.js': jsSnippetContent

  },

  title: 'Bootstrap Example',

  description: `Official example from ${window.location.href}`,

template: jsSnippet ? 'javascript' : 'html',   tags: ['bootstrap']

  }

```
    StackBlitzSDK.openProject(project, { openFile: 'index.html' })

  }

</script>

  </body>

</html>
```

8. TESTING

Test case	feature	component	Test scenario	Expected result	Actual result	status	comments	bug	Executed by
Sign in	Functional	Login page	Verify user can see the_sign in option	can visible	Yes visible	pass	successful	-	SIVA
Sign up	Functional	Login_page	Verify user has the option to sign up	Can visible	Yes visible	pass	Successful	-	SAKTHI

Test case									
Fetch news	Functional	Home page	Verify user can get the news	News will be feed to the app	404 error	Fail	unsuccessful	App integeration problem	SIVA, SAKTHI
Types of news available in the fetch news page	Functional	Fetch news	Types of news available	Weather, Sport, Economy.	Hover buttons will be shown.	Yes	successful	-	SIVA,SAKTHI, RAJ GANESH.

Result:

News tracker application using cloud is developed and executed at the level of completed progress .

10. ADVANTAGES & DISADVANTAGES

Advantages:-

- ✠ In this app news is already categorization.
- ✠ Easily accessible and portable.
- ✠ Better user experience.
- ✠ Apps help you convert visitors to loyal readers.
- ✠ Minute by minute updates of news.
- ✠ This app helps you to get local news updates instantly.
- ✠ To explore and discover trending news and topics.
- ✠ News feeds for you based on your interest.

Disadvantages:-

- ✠ Some apps demand premium subscription from user.

- ✠ Occurance of Advertisment disturb the user.
- ✠ Sometimes the news gives brief information.
- ✠ Prevailance of fake and uncertain news can confuse the user lead to misconception.
- ✠ Fake news may mislead the readers.

11. CONCLUSION

We explored the feasibility of recognising patterns of news reading interactions and evaluated three adaptive interface designs for different news reader types. We show that from their interaction log, a specific user can be recognised as one of three kinds. The reader types emerging from the online survey are well defined and distinct. The evaluation of the three variant interfaces suggests that different news reader types need different user interfaces. We have demonstrated a method for monitoring users' news reading behaviour and inferring news reader type from it. In the future we will further explore the design of adaptive interfaces, in order to be in a position to demonstrate a complete adaptive mobile news framework providing automatic personalisation of news apps.

12.FUTURE SCOPE

In the future we will further explore the design of adaptive interfaces, in order to be in a position to demonstrate a complete adaptive mobile news framework providing automatic personalisation of news apps.

13. APPENDIX

Source Code App.py

```
from glob import escape from flask import Flask, render_template,
request, redirect, url_for, session import ibm_db app = Flask(__name__)
#conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=824dfd4d-
99de440d-9991-
```

```
629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=30119;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=pss83307;PWD=Lab47l4rodOvB2WE","","")

@app.route('/') def

login():

    return render_template('signin.html')

@app.route('/signin.html',methods = ['POST'])

def getUser():    if request.method == 'POST':

user = request.form['uname']

        password = request.form['upwd']        sql = "SELECT *

FROM CUSTOMERS where Email = ?"        stmt =

ibm_db.prepare(conn, sql)        email = user

        # Explicitly bind parameters

ibm_db.bind_param(stmt, 1,user)

ibm_db.execute(stmt)        dictionary =

ibm_db.fetch_assoc(stmt)        pwd =

dictionary["PASSWORD"]        if

password != pwd:

            return render_template('error.html')

return render_template('base.html')

@app.route('/signup.html') def

putUser():

    return render_template('signup.html')

@app.route('/signup.html',methods = ['POST']) def

storedUser():

    if request.method == 'POST':        fname =

request.form['fname']        lname =
```

```python
request.form['lname']        mail =
request.form['mail']      npwd =
request.form['npwd']       cpwd =
request.form['cpwd']        res = fname + lname +
mail + npwd + cpwd        if npwd != cpwd:
        return render_template('signup.html')
    sql = "INSERT INTO customers
(FirstName,LastName,Email,password,confirmpassword) VALUES(?,?,?,?,?);"
stmt = ibm_db.prepare(conn, sql)        ibm_db.bind_param(stmt, 1, fname)
ibm_db.bind_param(stmt, 2, lname)        ibm_db.bind_param(stmt, 3, mail)
ibm_db.bind_param(stmt, 4, npwd)        ibm_db.bind_param(stmt, 5, cpwd)
ibm_db.execute(stmt)    return render_template('signin.html') if __name__ ==
'__main__':
   # run() method of Flask class runs the application
# on the local development server.
    app.run(debug=True)
```

Templates

About.html:

```
{% extends "base.html" %}

{% block head %}

<title>About Page</title>

{% endblock %}

{% block body %}

<h1>Welcome to About Page</h1>

<p> This is a paragraph</p>

{% endblock %}
```

Base.html:

```
<!DOCTYPE html>

<html lang="en">

<head>

{% block head %}

    <meta charset="utf-8">
```

```html
    <meta name="viewport" content="width=device-width , initial-scale=1">

<!--Bootstrap CSS-->

<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+
fzT" crossorigin="anonymous">

<link rel="stylesheet" type="text/css" href="static/sheets/styles1.css">

<!--Google Fonts-->

<link rel="preconnect" href="https://fonts.gstatic.com">

<link
href="https://fonts.googleapis.com/css2?family=Noto+Sans+JP:wght@300&dis
play=swap" rel="stylesheet">

<link rel="preconnect" href="https://fonts.gstatic.com">

<link
href="https://fonts.googleapis.com/css2?family=Nunito+Sans:ital@1&display=
swap" rel="stylesheet">

<link rel="preconnect" href="https://fonts.gstatic.com">

<link
href="https://fonts.googleapis.com/css2?family=Oxygen:wght@300&display=s
wap" rel="stylesheet">

<link rel="preconnect" href="https://fonts.gstatic.com">

<link
href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@500&
display=swap" rel="stylesheet">
```

```
{% endblock %}

</head>

<body>

    <nav class="navbar navbar-dark bg-dark fixed-top">

        <div class="container-fluid">

        <a class="navbar-brand" href="#">News Tracker</a>

        <button class="navbar-toggler" type="button" data-bs-toggle="offcanvas"
data-bs-target="#offcanvasDarkNavbar" ariacontrols="offcanvasDarkNavbar">

            <span class="navbar-toggler-icon"></span>

        </button>

        <div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1"
id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">

            <div class="offcanvas-header">

            <h5 class="offcanvas-title"
id="offcanvasDarkNavbarLabel">Profile</h5>

                <button type="button" class="btn-close btn-close-white" data-
bsdismiss="offcanvas" aria-label="Close"></button>

            </div>

            <div class="offcanvas-body">

            <ul class="navbar-nav justify-content-end flex-grow-1 pe-3">

                <li class="nav-item">

                <a class="nav-link active" aria-current="page"
href="home.html">Home</a>
```

```html
        </li>

        <li class="nav-item">

          <a class="nav-link" href="base.html">Fetch News</a>

        </li>

        <li class="nav-item">

          <a class="nav-link" href="about.html">About Us</a>

        </li>

        <li class="nav-item">

          <a class="nav-link" href="signin.html">Sign In</a>

        </li>

        <li class="nav-item">

          <a class="nav-link" href="signup.html">Sign Up</a>

        </li>

      </div>

    </div>

  </div>

</nav>

<div class="container div2">

  <div class="row">

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

      <a href="home.html" alt="Make A Transaction">

        <img src="static/Images/images.png" width="410" height="390">
```

```html
        </a>

      </div>

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

        <h1>Home page</h1>

        <p>

          <span>

            <ul class="content">

              <li class="content"><b>U can see what's in the home page using this link</b></li>

            </ul>

          </span>

        </p>

      </div>

    </div>

  </div>

  <div class="container div2">

    <div class="row">

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

        <h1>About Page</h1>

        <p>

          <span>

            <ul class="content">
```

```html
      <li class="content"><b>U can see what's in the about page using this link</b></li>
    </ul>
   </span>
  </p>
 </div>
 <div class="col-md-5">
  <a href="about.html" alt="Transaction History">
  <img src="static/Images/download.png" width="410" height="390">
 </a>
 </div>
 </div>
</div>
<div class="container div2">
 <div class="row">
  <div class="col-sm-5">
   <a href="signup.html" alt="Make A Transaction">
    <img src="static/Images/sign up.svg" width="410" height="390">
   </a>
  </div>
  <div class="col-sm-5">
    <h1>Sign Up Page</h1>
```

```html
        <p>
          <span>
            <ul class="content">
              <li class="content"><b>Create an account for yourself and enjoy the priviliges given to you</b></li>
            </ul>
          </span>
        </p>
        </div>
      </div>
    </div>
    <div class="container div2">
      <div class="row">
        <div class="col-md-5">
          <h1>Sign In Page</h1>
          <p>
            <span>
              <ul class="content">
                <li class="content"><b>Log In to your existing account if you have one</b></li>
              </ul>
            </span>
```

```
                </p>

              </div>

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

                <a href="signin.html" alt="Transaction History">

                  <img src="static/Images/sign in.svg" width="410"
height="390">

                </a>

              </div>

            </div>

          </div>

    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.j
s" integrity="sha384-
u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOm
Mi466C8" crossorigin="anonymous"></script>

    <script src="https://cdn.jsdelivr.net/npm/@docsearch/js@3"></script>

    <script
src="https://cdn.jsdelivr.net/npm/@stackblitz/sdk@1/bundles/sdk.umd.js"></sc
ript>

    <script src="/docs/5.2/assets/js/docs.min.js"></script>

    <script>      document.querySelectorAll('.btn-
edit').forEach(btn => {          btn.addEventListener('click',
event => {          const htmlSnippet =
event.target.closest('.bd-codesnippet').querySelector('.bd-
```

```
example').innerHTML        const classes =
Array.from(event.target.closest('.bd-
codesnippet').querySelector('.bd-example').classList).join(' ')
const jsSnippet = event.target.closest('.bd-
codesnippet').querySelector('.btn-edit').getAttribute('data-sb-js-
snippet')

      StackBlitzSDK.openBootstrapSnippet(htmlSnippet, jsSnippet, classes)

    })

  })

    StackBlitzSDK.openBootstrapSnippet = (htmlSnippet, jsSnippet, classes)
=> {        const markup = `<!doctype
html>

  <html lang="en">

    <head>

      <meta charset="utf-8">

      <meta name="viewport" content="width=device-width, initial-scale=1">

      <link
href="https:\/\/cdn.jsdelivr.net\/npm\/bootstrap@5.2.1\/dist\/css\/bootstrap.min.c
ss" rel="stylesheet">

      <link href="https://getbootstrap.com/docs/5.2/assets/css/docs.css"
rel="stylesheet">

      <title>Bootstrap Example</title>
```

```
      <${'script'}
src="https:\/\/cdn\.jsdelivr\.net\/npm\/bootstrap@5\.2\.1\/dist\/js\/bootstrap\.bun
dle\.min\.js"></${'script'}>

    </head>

    <body class="p-3 m-0 border-0 ${classes}">

      <!-- Example Code -->

    ${htmlSnippet.replace(/^/gm, '    ')}

      <!-- End Example Code -->

    '</body>         </html>`          const jsSnippetContent = jsSnippet ? '\/\/
NOTICE!!! Initially embedded in our docs this JavaScript\n\/\/ file contains
elements that can help you create reproducible\n\/\/ use cases in StackBlitz for
instance\.\n\/\/ In a real project please adapt this content to your needs\.\n\/\/
\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u0
02b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b
\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u0
02b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\n\n\/\*!\n \* JavaScript for
Bootstrap\u0027s docs \(https:\/\/getbootstrap\.com\/\)\n \* Copyright 2011\-
2022 The Bootstrap Authors\n \* Copyright 2011\-2022 Twitter, Inc\.\n \*
Licensed under the Creative Commons Attribution 3\.0 Unported License\.\n \*
For details, see https:\/\/creativecommons\.org\/licenses\/by\/3\.0\/\.\n \*\/\n\n\/\*
global bootstrap: false \*\/\n\n\(\(\) =\u003e \{\n  \u0027use strict\u0027\n\n  \/\/
\/-\/-\/-\/-\/-\/-\/-\n  \/\/ Tooltips\n  \/\/ \/-\/-\/-\/-\/-\/-\/-\n  \/\/ Instantiate all tooltips in a
docs    or    StackBlitz    page\n        document\.querySelectorAll\(\u0027\[data\-
```

```
bs\toggle=\u0022tooltip\u0022\]\u0027\)\n        \.forEach\(tooltip =\u003e \{\n
new bootstrap\.Tooltip\(tooltip\)\n    \}\)\n\n  \/\/ \-\-\-\-\-\-\-\-\n  \/\/ Popovers\n
\/\/ \-\-\-\-\-\-\-\-\n  \/\/ Instantiate all popovers in a docs or StackBlitz page\n
document\.querySelectorAll\(\u0027\[data\-
bs\toggle=\u0022popover\u0022\]\u0027\)\n    \.forEach\(popover =\u003e \{\n
new bootstrap\.Popover\(popover\)\n    \}\)\n\n  \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-
\\-\-\-\-\-\-\-\-\-\-\-\-\n  \/\/ Toasts\n  \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\\-
\-\-\-\n  \/\/ Used by \u0027Placement\u0027 example in docs or StackBlitz\n
const                    toastPlacement                    =
document\.getElementById\(\u0027toastPlacement\u0027\)\n  if
\(toastPlacement\) \{\n
document\.getElementById\(\u0027selectToastPlacement\u0027\)\.addEventLis
tener\(\u0027change\u0027, function \(\) \{\n      if
\(!toastPlacement\.dataset\.originalClass\) \{\n
toastPlacement\.dataset\.originalClass = toastPlacement\.className\n      \}\n\n
toastPlacement\.className = `\$\{toastPlacement\.dataset\.originalClass\}
\$\{this\.value\}`\n    \}\)\n  \}\n\n  \/\/ Instantiate all toasts in a docs page only\n
document\.querySelectorAll\(\u0027\.bd\-example \.toast\u0027\)\n
\.forEach\(toastNode =\u003e \{\n      const toast = new
bootstrap\.Toast\(toastNode, \{\n        autohide: false\n      \}\)\n\n
toast\.show\(\)\n    \}\)\n\n  \/\/ Instantiate all toasts in a docs page only\n  const
toastTrigger = document\.getElementById\(\u0027liveToastBtn\u0027\)\n
```

const toastLiveExample =

document\.getElementById\(\u0027liveToast\u0027\)\n if \(toastTrigger\) \{\n

toastTrigger\.addEventListener\(\u0027click\u0027, \(\) =\u003e \{\n   const

toast = new bootstrap\.Toast\(toastLiveExample\)\n\n   toast\.show\(\)\n

\}\)\n \}\n\n \/\/ \-\n \/\/ Alerts\n

\/\/ \-\n \/\/ Used in \u0027Show

live toast\u0027 example in docs or StackBlitz\n const alertPlaceholder =

document\.getElementById\(\u0027liveAlertPlaceholder\u0027\)\n   const

alertTrigger = document\.getElementById\(\u0027liveAlertBtn\u0027\)\n\n

const appendAlert = \(message, type\) =\u003e \{\n   const wrapper =

document\.createElement\(\u0027div\u0027\)\n   wrapper\.innerHTML = \[\n

`\u003cdiv class=\u0022alert alert\-\$\{type\} alert\-dismissible\u0022

role=\u0022alert\u0022\u003e`,\n      `

\u003cdiv\u003e\$\{message\}\u003c\/div\u003e`,\n   \u0027 \u003cbutton

type=\u0022button\u0022   class=\u0022btn\-close\u0022   data\-

bs\dismiss=\u0022alert\u0022

aria\label=\u0022Close\u0022\u003e\u003c\/button\u003e\u0027,\n

\u0027\u003c\/div\u003e\u0027\n   \]\.join\(\u0027\u0027\)\n\n

alertPlaceholder\.append\(wrapper\)\n   \}\n\n   if \(alertTrigger\) \{\n

alertTrigger\.addEventListener\(\u0027click\u0027,   \(\)   =\u003e   \{\n

appendAlert\(\u0027Nice, you triggered this alert message!\u0027,

\u0027success\u0027\)\n   \}\)\n \}\n\n \/\/ \-\\-\-\-\-\-\-\-\-\-\n \/\/ Checks \u0026 Radios\n \/\/ \-\\-\-\-\-\-\-\-\-\-\-\n \/\/ Indeterminate checkbox example in docs and StackBlitz\n document\.querySelectorAll\(\u0027\.bd\-example\-indeterminate \[type=\u0022checkbox\u0022\]\u0027\)\n   \.forEach\(checkbox =\u003e \{\n   if \(checkbox\.id\.includes\(\u0027Indeterminate\u0027\)\) \{\n   checkbox\.indeterminate = true\n   \}\n   \}\)\n\n \/\/ \-\\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n \/\/ Links\n \/\/ \-\n \/\/ Disable empty links in docs examples only\n document\.querySelectorAll\(\u0027\.bd\-content \[href=\u0022#\u0022\]\u0027\)\n   \.forEach\(link =\u003e \{\n   link\.addEventListener\(\u0027click\u0027, event =\u003e \{\n   event\.preventDefault\(\)\n   \}\)\n   \}\)\n\n \/\/ \-\\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n \/\/ Modal\n \/\/ \-\n \/\/ Modal \u0027Varying modal content\u0027 example in docs and StackBlitz\n const exampleModal = document\.getElementById\(\u0027exampleModal\u0027\)\n if \(exampleModal\) \{\n   exampleModal\.addEventListener\(\u0027show\.bs\.modal\u0027, event =\u003e \{\n   \/\/ Button that triggered the modal\n   const button = event\.relatedTarget\n   \/\/ Extract info from data\-bs\-* attributes\n   const recipient = button\.getAttribute\(\u0027data\-bs\-whatever\u0027\)\n\n   \/\/

Update the modal\u0027s content\.\n   const modalTitle = exampleModal\.querySelector\(\(\u0027\.modal\-title\u0027\)\)\n   const modalBodyInput = exampleModal\.querySelector\(\(\u0027\.modal\-body input\u0027\)\)\n\n   modalTitle\.textContent = `New message to \$\{recipient\}`\n   modalBodyInput\.value = recipient\n   \}\)\n \}\n\n \/\/ \\-\n \/\/ Offcanvas\n \/\/ \-\-\-\-\-\-\-\\-\n   \/\/ \u0027Offcanvas components\u0027 example in docs only\n   const myOffcanvas = document\.querySelectorAll\(\(\u0027\.bd\-example\-offcanvas \.offcanvas\u0027\)\n if \(myOffcanvas\) \{\n   myOffcanvas\.forEach\(offcanvas =\u003e \{\n   offcanvas\.addEventListener\(\(\u0027show\.bs\.offcanvas\u0027, event =\u003e \{\n   event\.preventDefault\(\)\n   \}, false\)\n   \}\)\n \}\n\}\)\(\)\n' : null

const project = {   files: {

      'index.html': markup,

      'index.js': jsSnippetContent

    },

    title: 'Bootstrap Example',   description: `Official example from ${window.location.href}`,   template: jsSnippet ?

'javascript' : 'html',   tags: ['bootstrap']

```
        }

        StackBlitzSDK.openProject(project, { openFile: 'index.html' })

      }

    </script>

  </body>

    </html>
```

change   pass.html:

```
<!doctype html>

<html lang="en">

  <head>

    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width, initial-scale=1">
<meta name="description" content="">

    <meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap

contributors">

    <meta name="generator" content="Hugo 0.84.0">

    <title>Sign In</title>
```

```html
    <link rel="canonical"
href="https://getbootstrap.com/docs/5.0/examples/signin/">

    <link href="https://getbootstrap.com/docs/5.0/assets/css/docs.css"
rel="stylesheet">

    <!-- Bootstrap core CSS -->

    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+
fzT" crossorigin="anonymous">

    <!-- Favicons -->

<link rel="apple-touch-icon" href="/docs/5.0/assets/img/favicons/apple-
touchicon.png" sizes="180x180">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-32x32.png"
sizes="32x32" type="image/png">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-16x16.png"
sizes="16x16" type="image/png">

<link rel="manifest" href="/docs/5.0/assets/img/favicons/manifest.json">
```

```html
<link rel="mask-icon" href="/docs/5.0/assets/img/favicons/safari-pinnedtab.svg" color="#7952b3">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon.ico">

<meta name="theme-color" content="#7952b3">

  <style>

    .bd-placeholder-img {

font-size:        1.125rem;

text-anchor: middle;

      -webkit-user-select: none;

-moz-user-select:          none;

user-select: none;

    }

    @media (min-width: 768px) {

      .bd-placeholder-img-lg {

font-size: 3.5rem;

      }

    }

  </style>
```

```html
    <!-- Custom styles for this template -->

    <link href="static/sheets/signin.css" rel="stylesheet">

<link href="static/sheets/colors.css" rel="stylesheet">

  </head>

  <body class="text-center">

    <nav class="navbar navbar-dark bg-dark fixed-top">

      <div class="container-fluid">

        <a class="navbar-brand" href="#">News Tracker</a>

        <button class="navbar-toggler" type="button" data-
bstoggle="offcanvas" data-bs-target="#offcanvasDarkNavbar"
ariacontrols="offcanvasDarkNavbar">

          <span class="navbar-toggler-icon"></span>

        </button>

        <div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1"
id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">

          <div class="offcanvas-header">

            <h5 class="offcanvas-title"
id="offcanvasDarkNavbarLabel">Profile</h5>
```

```html
        <button type="button" class="btn-close btn-close-white" data-bsdismiss="offcanvas" aria-label="Close"></button>

      </div>

      <div class="offcanvas-body">

        <ul class="navbar-nav justify-content-end flex-grow-1 pe-3">

          <li class="nav-item">

            <a class="nav-link active" aria-current="page" href="home.html">Home</a>

          </li>

          <li class="nav-item">

            <a class="nav-link" href="base.html">Fetch News</a>

          </li>

          <li class="nav-item">

            <a class="nav-link" href="about.html">About Us</a>

          </li>

          <li class="nav-item">

            <a class="nav-link" href="signin.html">Sign In</a>

          </li>
```

```html
        <li class="nav-item">

          <a class="nav-link" href="signup.html">Sign Up</a>

        </li>

      </div>

    </div>

  </nav>
<main class="form-signin  ">

  <form action = "{{ url_for("getUser")}}" method="POST">

    <img class="mb-4" src="static/Images/house-user-solid.svg" alt=""
width="72" height="57">

    <h1 class="h3 mb-3 fw-normal white">Change Password</h1>

    <div class="form-floating">

      <input type="email" class="form-control" id="floatingInput" name =
"uname" placeholder="name@example.com">

      <label for="floatingPassword">Password</label>

    </div>

    <br>
```

```html
    <div class="form-floating">

      <input type="password" class="form-control" id="floatingPassword" name
= "upwd" placeholder="Password">

      <label for="floatingPassword">Confirm Password</label>

    </div>

    <button class="w-100 btn btn-lg bd-green-500 btn-outline-warning"
type="submit">Change</button><br><br>

  </form>

</main>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.j
s" integrity="sha384-
u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOm
Mi466C8" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/@docsearch/js@3"></script>

<script
src="https://cdn.jsdelivr.net/npm/@stackblitz/sdk@1/bundles/sdk.umd.js"></sc
ript>

<script src="/docs/5.2/assets/js/docs.min.js"></script>
```

```
<script>  document.querySelectorAll('.btn-edit').forEach(btn => {

btn.addEventListener('click', event => {       const htmlSnippet =

event.target.closest('.bd-codesnippet').querySelector('.bd-example').innerHTML

const classes = Array.from(event.target.closest('.bd-

codesnippet').querySelector('.bd-example').classList).join(' ')       const jsSnippet

= event.target.closest('.bd-code-

snippet').querySelector('.btnedit').getAttribute('data-sb-js-snippet')

    StackBlitzSDK.openBootstrapSnippet(htmlSnippet, jsSnippet, classes)

  })

 })

  StackBlitzSDK.openBootstrapSnippet = (htmlSnippet, jsSnippet, classes) => {
const markup = `<!doctype html>

<html lang="en">

 <head>

  <meta charset="utf-8">

  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link

href="https:\/\/cdn.jsdelivr.net\/npm\/bootstrap@5.2.1\/dist\/css\/bootstrap.min.

c ss" rel="stylesheet">
```

```
    <link href="https://getbootstrap.com/docs/5.2/assets/css/docs.css"
rel="stylesheet">

    <title>Bootstrap Example</title>

    <${'script'}
src="https:\/\/cdn\.jsdelivr\.net\/npm\/bootstrap@5\.2\.1\/dist\/js\/bootstrap\.bun
dle\.min\.js"></${'script'}>

  </head>

  <body class="p-3 m-0 border-0 ${classes}">

    <!-- Example Code -->
${htmlSnippet.replace(/^/gm, '    ')}
    <!-- End Example Code -->

  </body> </html>`    const jsSnippetContent = jsSnippet ? '\/\/ NOTICE!!!
Initially embedded in our docs this JavaScript\n\/\/ file contains elements that
can help you create reproducible\n\/\/ use cases in StackBlitz for
instance\.\n\/\/ In a real project please adapt this content to your needs\.\n\/\/
\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u0
02b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b
\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u0
02b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\n\n\/\*!\n \* JavaScript for
Bootstrap\u0027s docs \(https:\/\/getbootstrap\.com\/\)\n \* Copyright 2011\-
```

global bootstrap: false *\/\n\n\(\(\) =\u003e \{\n \u0027use strict\u0027\n\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Tooltips\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Instantiate all tooltips in a docs or StackBlitz page\n   document\.querySelectorAll\(\(\u0027\[data\-bs\toggle=\u0022tooltip\u0022\]\u0027\)\)\n   \.forEach\(\(tooltip =\u003e \{\n   new bootstrap\.Tooltip\(tooltip\)\n   \}\)\)\n\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Popovers\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Instantiate all popovers in a docs or StackBlitz page\n   document\.querySelectorAll\(\(\u0027\[data\-bs\toggle=\u0022popover\u0022\]\u0027\)\)\n   \.forEach\(\(popover =\u003e \{\n   new bootstrap\.Popover\(popover\)\n   \}\)\)\n\n \/\/ \-\n\\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n \/\/ Toasts\n \/\/ \-\n\\-\-\-\-\-\n \/\/ Used by \u0027Placement\u0027 example in docs or StackBlitz\n const   toastPlacement   = document\.getElementById\(\(\u0027toastPlacement\u0027\)\)\n if \(toastPlacement\) \{\n   document\.getElementById\(\(\u0027selectToastPlacement\u0027\)\)\.addEventListener\(\(\u0027change\u0027, function \(\) \{\n   if \(!toastPlacement\.dataset\.originalClass\) \{\n   toastPlacement\.dataset\.originalClass = toastPlacement\.className\n   \}\n\n   toastPlacement\.className = `\$\{toastPlacement\.dataset\.originalClass\}

\$\{this\.value\}`\n   \}\)\)\n \}\n\n \/\/ Instantiate all toasts in a docs page only\n document\.querySelectorAll\(\(\u0027\.bd\-example \.toast\u0027\)\)\n \.forEach\(toastNode =\u003e \{\n   const toast = new bootstrap\.Toast\(toastNode, \{\n   autohide: false\n   \}\)\n\n toast\.show\(\)\n   \}\)\)\n\n \/\/ Instantiate all toasts in a docs page only\n const toastTrigger = document\.getElementById\(\(\u0027liveToastBtn\u0027\)\)\n const toastLiveExample = document\.getElementById\(\(\u0027liveToast\u0027\)\)\n if \(toastTrigger\) \{\n toastTrigger\.addEventListener\(\(\u0027click\u0027, \(\) =\u003e \{\n   const toast = new bootstrap\.Toast\(toastLiveExample\)\n\n   toast\.show\(\)\n

\}\)\n \}\n\n \/\/ \-\n \/\/ Alerts\n \/\/ \-\n \/\/ Used in \u0027Show live toast\u0027 example in docs or StackBlitz\n const alertPlaceholder = document\.getElementById\(\(\u0027liveAlertPlaceholder\u0027\)\)\n const alertTrigger = document\.getElementById\(\(\u0027liveAlertBtn\u0027\)\)\n\n const appendAlert = \(message, type\) =\u003e \{\n   const wrapper = document\.createElement\(\(\u0027div\u0027\)\n   wrapper\.innerHTML = \[\n `\u003cdiv class=\u0022alert alert\-\$\{type\} alert\-dismissible\u0022 role=\u0022alert\u0022\u003e`,\n   `   \u003cdiv\u003e\$\{message\}\u003c\/div\u003e`,\n   \u0027 \u003cbutton type=\u0022button\u0022 class=\u0022btn\-close\u0022 data\-

bs\dismiss=\u0022alert\u0022

aria\label=\u0022Close\u0022\u003e\u003c\/button\u003e\u0027,\n

\u0027\u003c\/div\u003e\u0027\n   \]\.join\(\u0027\u0027\)\)\n\n

alertPlaceholder\.append\(wrapper\)\n   \}\n\n   if \(alertTrigger\) \{\n

alertTrigger\.addEventListener\(\u0027click\u0027,   \(\)   =\u003e   \{\n

appendAlert\(\u0027Nice, you triggered this alert message!\u0027,

\u0027success\u0027\)\n   \}\)\n \}\n\n \/\/ \-\\-

\-\-\-\-\-\-\-\-\-\n \/\/ Checks \u0026 Radios\n \/\/ \-\\-\-

\-\-\-\-\-\-\-\-\-\n   \/\/ Indeterminate checkbox example in docs and StackBlitz\n

document\.querySelectorAll\(\u0027\.bd\-example\-indeterminate

\[type=\u0022checkbox\u0022\]\u0027\)\n   \.forEach\(checkbox =\u003e \{\n

if   \(checkbox\.id\.includes\(\u0027Indeterminate\u0027\)\)   \{\n

checkbox\.indeterminate = true\n   \}\n   \}\)\n\n \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\\-

\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n \/\/ Links\n \/\/ \-

\-\-\-\-\-\-\-\n \/\/ Disable empty links in docs examples only\n

document\.querySelectorAll\(\u0027\.bd\-content

\[href=\u0022#\u0022\]\u0027\)\n   \.forEach\(link   =\u003e   \{\n

link\.addEventListener\(\u0027click\u0027,   event   =\u003e   \{\n

event\.preventDefault\(\)\n   \}\)\n   \}\)\n\n \/\/ \-\\-

\-\-\-\-\-\-\-\-\-\-\-\-\-\n \/\/ Modal\n \/\/ \-

```
\-\-\-\-\n  \/\/ Modal \u0027Varying modal content\u0027 example in docs and StackBlitz\n    const exampleModal = document\.getElementById\(\u0027exampleModal\u0027\)\n  if \(exampleModal\)                                                    \{\n    exampleModal\.addEventListener\(\u0027show\.bs\.modal\u0027,    event =\u003e \{\n      \/\/ Button that triggered the modal\n      const button = event\.relatedTarget\n    \/\/ Extract info from data\-bs\-\* attributes\n      const recipient = button\.getAttribute\(\u0027data\-bs\-whatever\u0027\)\n\n      \/\/ Update the modal\u0027s content\.\n            const modalTitle = exampleModal\.querySelector\(\u0027\.modal\-title\u0027\)\n            const modalBodyInput = exampleModal\.querySelector\(\u0027\.modal\-body input\u0027\)\n\n      modalTitle\.textContent = `New message to \$\{recipient\}`\n      modalBodyInput\.value = recipient\n    \}\)\n  \}\n\n \/\/ \\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n  \/\/ Offcanvas\n  \/\/ \-\-\-\-\-\-\-\\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n  \/\/ \u0027Offcanvas components\u0027 example in docs only\n      const myOffcanvas = document\.querySelectorAll\(\u0027\.bd\-example\-offcanvas \.offcanvas\u0027\)\n  if \(myOffcanvas\) \{\n    myOffcanvas\.forEach\(offcanvas =\u003e \{\n    offcanvas\.addEventListener\(\u0027show\.bs\.offcanvas\u0027, event =\u003e \{\n      event\.preventDefault\(\)\n      \}, false\)\n    \}\)\n  \}\n\}\)\(\)\n' : null

const project = {
```

```
    files: {

      'index.html': markup,

      'index.js': jsSnippetContent

    },

    title: 'Bootstrap Example',      description: `Official example

from ${window.location.href}`,      template: jsSnippet ?

'javascript' : 'html',      tags: ['bootstrap']

    }

    StackBlitzSDK.openProject(project, { openFile: 'index.html' })

  }

</script>

  </body>

</html>
```

Error.html:

```
<!doctype html>

<html lang="en">

  <head>

    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width, initial-scale=1">

    <meta name="description" content="">
```

```html
<meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap
contributors">

<meta name="generator" content="Hugo 0.84.0">

<title>Sign In</title>

<link rel="canonical"
href="https://getbootstrap.com/docs/5.0/examples/signin/">

<link href="https://getbootstrap.com/docs/5.0/assets/css/docs.css"
rel="stylesheet">

<!-- Bootstrap core CSS -->

<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+
fzT" crossorigin="anonymous">

<!-- Favicons -->
<link rel="apple-touch-icon" href="/docs/5.0/assets/img/favicons/apple-
touchicon.png" sizes="180x180">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-32x32.png"
sizes="32x32" type="image/png">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-16x16.png"
sizes="16x16" type="image/png">

<link rel="manifest" href="/docs/5.0/assets/img/favicons/manifest.json">

<link rel="mask-icon" href="/docs/5.0/assets/img/favicons/safari-
pinnedtab.svg" color="#7952b3">
```

```html
<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon.ico">

<meta name="theme-color" content="#7952b3">

    <style>

    .bd-placeholder-img {
font-size:         1.125rem;
text-anchor: middle;
      -webkit-user-select: none;
-moz-user-select:          none;
user-select: none;
    }
    @media (min-width: 768px) {
     .bd-placeholder-img-lg {

      font-size: 3.5rem;

     }

    }
   </style>
   <!-- Custom styles for this template -->
   <link href="signin.css" rel="stylesheet">
   <link href="colors.css" rel="stylesheet">
  </head>
  <body class="text-center">
     <nav class="navbar navbar-dark bg-dark fixed-top">
       <div class="container-fluid">
         <a class="navbar-brand" href="#">About Me</a>
```

```html
        <button class="navbar-toggler" type="button" data-
bstoggle="offcanvas" data-bs-target="#offcanvasDarkNavbar"
ariacontrols="offcanvasDarkNavbar">

          <span class="navbar-toggler-icon"></span>

        </button>

        <div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1"
id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">

          <div class="offcanvas-header">

            <h5 class="offcanvas-title"
id="offcanvasDarkNavbarLabel">Profile</h5>

            <button type="button" class="btn-close btn-close-white" data-
bsdismiss="offcanvas" aria-label="Close"></button>

          </div>

          <div class="offcanvas-body">

           <ul class="navbar-nav justify-content-end flex-grow-1 pe-3">

             <li class="nav-item">

               <a class="nav-link active" aria-current="page"
href="home.html">Home</a>

             </li>

             <li class="nav-item">

               <a class="nav-link" href="base.html">Main</a>

             </li>

             <li class="nav-item">
```

```html
          <a class="nav-link" href="about.html">About</a>

        </li>

        <li class="nav-item">

          <a class="nav-link" href="signin.html">Sign In</a>

        </li>

        <li class="nav-item">

          <a class="nav-link" href="signup.html">Sign Up</a>

        </li>

      </div>

    </div>

   </div>

  </nav>
<main class="form-signin">
 <form action="{{ url_for("login")}}">

  <div class="form-floating">

    <h1 class="pink div2">Sorry for the inconvience caused.</h1><br>

  </div>

  <br>

  <div class="form-floating">

    <h3 class="red div2">You have entered the wrong credentials.</h3><br>

</div>
```

```html
<button class="w-100 btn btn-lg bd-purple-500 btn-outline-warning center" type="submit">Return To Home Page</button><br><br>

  </form>

  </main>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js" integrity="sha384-u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/@docsearch/js@3"></script>

<script src="https://cdn.jsdelivr.net/npm/@stackblitz/sdk@1/bundles/sdk.umd.js"></script>

<script src="/docs/5.2/assets/js/docs.min.js"></script>

  <script>    document.querySelectorAll('.btn-edit').forEach(btn => {        btn.addEventListener('click', event => {        const htmlSnippet = event.target.closest('.bd-codesnippet').querySelector('.bd-example').innerHTML
const classes = Array.from(event.target.closest('.bd-codesnippet').querySelector('.bd-example').classList).join(' ')


      const jsSnippet = event.target.closest('.bd-code-snippet').querySelector('.btn-edit').getAttribute('data-sb-js-snippet')

      StackBlitzSDK.openBootstrapSnippet(htmlSnippet, jsSnippet, classes)
```

```
  })

})

StackBlitzSDK.openBootstrapSnippet = (htmlSnippet, jsSnippet, classes) =>
{     const markup = `<!doctype

html>

  <html lang="en">

  <head>

    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width, initial-scale=1">

    <link
href="https:\/\/cdn.jsdelivr.net\/npm\/bootstrap@5.2.1\/dist\/css\/bootstrap.min.c
ss" rel="stylesheet">

    <link href="https://getbootstrap.com/docs/5.2/assets/css/docs.css"
rel="stylesheet">

    <title>Bootstrap Example</title>

    <${'script'}
src="https:\/\/cdn\.jsdelivr\.net\/npm\/bootstrap@5\.2\.1\/dist\/js\/bootstrap\.bun
dle\.min\.js"></${'script'}>

  </head>

  <body class="p-3 m-0 border-0 ${classes}">

    <!-- Example Code -->

  ${htmlSnippet.replace(/^/gm, '    ')}

    <!-- End Example Code -->
```

```
</body>    </html>`     const jsSnippetContent = jsSnippet ? '\/\/ NOTICE!!! Initially embedded in our docs this JavaScript\n\/\/ file contains elements that can help you create reproducible\n\/\/ use cases in StackBlitz for instance\.\n\/\/ In a real project please adapt this content to your needs\.\n\/\/ \u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\n\n\/\*!\n \* JavaScript for Bootstrap\u0027s docs \(https:\/\/getbootstrap\.com\/\)\n \* Copyright 2011\-2022 The Bootstrap Authors\n \* Copyright 2011\-2022 Twitter, Inc\.\n \* Licensed under the Creative Commons Attribution 3\.0 Unported License\.\n \* For details, see https:\/\/creativecommons\.org\/licenses\/by\/3\.0\/\.\n \*\/\n\n\/\* global bootstrap: false \*\/\n\n\(\(\) =\u003e \{\n  \u0027use strict\u0027\n\n  \/\/ \-\-\-\-\-\-\-\-\-\n  \/\/ Tooltips\n  \/\/ \-\-\-\-\-\-\-\-\-\n  \/\/ Instantiate all tooltips in a docs or StackBlitz page\n  document\.querySelectorAll\(\u0027\[data\-bs\toggle=\u0022tooltip\u0022\]\u0027\)\n    \.forEach\(tooltip =\u003e \{\n  new bootstrap\.Tooltip\(tooltip\)\n    \}\)\n\n  \/\/ \-\-\-\-\-\-\-\-\-\n  \/\/ Popovers\n  \/\/ \-\-\-\-\-\-\-\-\-\n  \/\/ Instantiate all popovers in a docs or StackBlitz page\n  document\.querySelectorAll\(\u0027\[data\-bs\toggle=\u0022popover\u0022\]\u0027\)\n    \.forEach\(popover =\u003e \{\n  new bootstrap\.Popover\(popover\)\n    \}\)\n\n  \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n  \/\/ Toasts\n  \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n  \/\/ Used by \u0027Placement\u0027 example in docs or StackBlitz\n  const toastPlacement = document\.getElementById\(\u0027toastPlacement\u0027\)\n  if \(toastPlacement\) \{\n    document\.getElementById\(\u0027selectToastPlacement\u0027\)\.addEventListener\(\u0027change\u0027, function \(\) \{\n      if \(!toastPlacement\.dataset\.originalClass\) \{\n
```

```
toastPlacement\.dataset\.originalClass = toastPlacement\.className\n    \}\n\n
toastPlacement\.className = `\$\{toastPlacement\.dataset\.originalClass\}
\$\{this\.value\}`\n    \}\)\n  \}\n\n  \/\/ Instantiate all toasts in a docs page only\n
document\.querySelectorAll\(\(\u0027\.bd\-example \.toast\u0027\)\)\n
\.forEach\(toastNode =\u003e \{\n                const toast = new
bootstrap\.Toast\(toastNode, \{\n           autohide: false\n           \}\)\n\n
toast\.show\(\)\n    \}\)\n\n  \/\/ Instantiate all toasts in a docs page only\n  const
toastTrigger = document\.getElementById\(\(\u0027liveToastBtn\u0027\)\)\n
const                        toastLiveExample                        =
document\.getElementById\(\(\u0027liveToast\u0027\)\)\n  if \(toastTrigger\) \{\n
toastTrigger\.addEventListener\(\(\u0027click\u0027, \(\) =\u003e \{\n     const
toast = new bootstrap\.Toast\(toastLiveExample\)\n\n         toast\.show\(\)\n
\}\)\n  \}\n\n  \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n  \/\/ Alerts\n
\/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n  \/\/ Used in \u0027Show
live toast\u0027 example in docs or StackBlitz\n  const alertPlaceholder =
document\.getElementById\(\(\u0027liveAlertPlaceholder\u0027\)\)\n        const
alertTrigger = document\.getElementById\(\(\u0027liveAlertBtn\u0027\)\)\n\n
const appendAlert = \(message, type\) =\u003e \{\n      const wrapper =
document\.createElement\(\(\u0027div\u0027\)\)\n    wrapper\.innerHTML = \[\n
`\u003cdiv class=\u0022alert alert\-\$\{type\} alert\-dismissible\u0022
role=\u0022alert\u0022\u003e`,\n      `
\u003cdiv\u003e\$\{message\}\u003c\/div\u003e`,\n      \u0027   \u003cbutton
type=\u0022button\u0022        class=\u0022btn\-close\u0022        data\-
bs\dismiss=\u0022alert\u0022
aria\label=\u0022Close\u0022\u003e\u003c\/button\u003e\u0027,\n
\u0027\u003c\/div\u003e\u0027\n                  \]\.join\(\(\u0027\u0027\)\)\n\n
alertPlaceholder\.append\(wrapper\)\n    \}\n\n    if \(alertTrigger\) \{\n
alertTrigger\.addEventListener\(\(\u0027click\u0027,    \(\)    =\u003e    \{\n
appendAlert\(\(\u0027Nice, you triggered this alert message!\u0027,
```

\u0027success\u0027\)\n   \}\)\n \}\n\n // \-\\-\-\-\-\-\-\-\-\n // Checks \u0026 Radios\n // \-\\-\-\-\-\-\-\-\-\-\n // Indeterminate checkbox example in docs and StackBlitz\n document\.querySelectorAll\(\u0027\.bd\-example\-indeterminate \[type=\u0022checkbox\u0022\]\u0027\)\n   \.forEach\(checkbox =\u003e \{\n if   \(checkbox\.id\.includes\(\u0027Indeterminate\u0027\)\)   \{\n checkbox\.indeterminate = true\n   \}\n   \}\)\n\n // \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n // Links\n // \-\n // Disable empty links in docs examples only\n document\.querySelectorAll\(\u0027\.bd\-content \[href=\u0022#\u0022\]\u0027\)\n   \.forEach\(link =\u003e \{\n link\.addEventListener\(\u0027click\u0027, event =\u003e \{\n event\.preventDefault\(\)\n   \}\)\n   \}\)\n\n // \-\\-\-\-\-\-\-\-\-\-\-\-\n // Modal\n // \-\n // Modal \u0027Varying modal content\u0027 example in docs and StackBlitz\n   const   exampleModal   = document\.getElementById\(\u0027exampleModal\u0027\)\n if \(exampleModal\)   \{\n exampleModal\.addEventListener\(\u0027show\.bs\.modal\u0027,   event =\u003e \{\n   // Button that triggered the modal\n   const button = event\.relatedTarget\n   // Extract info from data\-bs\-* attributes\n   const recipient = button\.getAttribute\(\u0027data\-bs\-whatever\u0027\)\n\n   // Update the modal\u0027s content\.\n   const modalTitle = exampleModal\.querySelector\(\u0027\.modal\-title\u0027\)\n   const modalBodyInput   =   exampleModal\.querySelector\(\u0027\.modal\-body input\u0027\)\n\n   modalTitle\.textContent = `New message to \$\{recipient\}`\n   modalBodyInput\.value = recipient\n   \}\)\n \}\n\n // \\-\n // Offcanvas\n // \-\-\-\-\-\-\-\\-\n   // \u0027Offcanvas components\u0027

example in docs only\n const myOffcanvas = document\.querySelectorAll\(\(\u0027\.bd\-example\-offcanvas \.offcanvas\u0027\)\)\n if \(myOffcanvas\) \{\n myOffcanvas\.forEach\(offcanvas =\u003e \{\n offcanvas\.addEventListener\(\(\u0027show\.bs\.offcanvas\u0027, event =\u003e \{\n event\.preventDefault\(\)\n \}, false\)\n \}\)\n \}\n\}\)\(\)\n' : null

```
const project = {        files: {

        'index.html': markup,

        'index.js': jsSnippetContent

    },

    title: 'Bootstrap Example',        description: `Official example

from ${window.location.href}`,        template: jsSnippet ?

'javascript' : 'html',        tags: ['bootstrap']

    }

    StackBlitzSDK.openProject(project, { openFile: 'index.html' })

  }

  </script>

 </body>

</html>
```

Forpass.html

```
<!doctype html>

<html lang="en">

 <head>

  <meta charset="utf-8">

  <meta name="viewport" content="width=device-width, initial-scale=1">
<meta name="description" content="">

  <meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap
contributors">

  <meta name="generator" content="Hugo 0.84.0">

  <title>Sign In</title>

  <link rel="canonical"
href="https://getbootstrap.com/docs/5.0/examples/signin/">

  <link href="https://getbootstrap.com/docs/5.0/assets/css/docs.css"
rel="stylesheet">

  <!-- Bootstrap core CSS -->

  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+
fzT" crossorigin="anonymous">

  <!-- Favicons -->
```

```html
<link rel="apple-touch-icon" href="/docs/5.0/assets/img/favicons/apple-touchicon.png" sizes="180x180">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-32x32.png" sizes="32x32" type="image/png">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-16x16.png" sizes="16x16" type="image/png">

<link rel="manifest" href="/docs/5.0/assets/img/favicons/manifest.json">

<link rel="mask-icon" href="/docs/5.0/assets/img/favicons/safari-pinnedtab.svg" color="#7952b3">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon.ico">

<meta name="theme-color" content="#7952b3">

    <style>

    .bd-placeholder-img {

font-size:          1.125rem;

text-anchor: middle;

    -webkit-user-select: none;

-moz-user-select:          none;

user-select: none;

    }

    @media (min-width: 768px) {

.bd-placeholder-img-lg {          font-

size: 3.5rem;

    }
```

```html
      }

    </style>

    <!-- Custom styles for this template -->

    <link href="static/sheets/signin.css" rel="stylesheet">

<link href="static/sheets/colors.css" rel="stylesheet">

  </head>

  <body class="text-center">

      <nav class="navbar navbar-dark bg-dark fixed-top">

        <div class="container-fluid">

          <a class="navbar-brand" href="#">News Tracker</a>

          <button class="navbar-toggler" type="button" data-
bstoggle="offcanvas" data-bs-target="#offcanvasDarkNavbar"
ariacontrols="offcanvasDarkNavbar">

            <span class="navbar-toggler-icon"></span>

          </button>

          <div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1"
id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">

            <div class="offcanvas-header">

              <h5 class="offcanvas-title"
id="offcanvasDarkNavbarLabel">Profile</h5>

              <button type="button" class="btn-close btn-close-white" data-
bsdismiss="offcanvas" aria-label="Close"></button>

            </div>
```

```html
<div class="offcanvas-body">
  <ul class="navbar-nav justify-content-end flex-grow-1 pe-3">
    <li class="nav-item">
      <a class="nav-link active" aria-current="page" href="home.html">Home</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="base.html">Fetch News</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="about.html">About Us</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="signin.html">Sign In</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="signup.html">Sign Up</a>
    </li>
  </div>
</div>
</nav>
```

```html
<main class="form-signin  ">

  <form action = "{{ url_for("getUser")}}" method="POST">

    <img class="mb-4" src="static/Images/house-user-solid.svg" alt=""
width="72" height="57">

    <h1 class="h3 mb-3 fw-normal white">Change your password</h1>

    <div class="form-floating">

      <input type="email" class="form-control" id="floatingInput" name =
"uname" placeholder="name@example.com">

      <label for="floatingInput">Email address</label>

    </div>

    <br>

    <button class="w-100 btn btn-lg btn-warning btn-outline-warning"
type="submit"><a   href="static/templates/changepass.html">Change
Password</a></button><br><br>

  </form>

</main>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.j
s" integrity="sha384-
u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOm
Mi466C8" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/@docsearch/js@3"></script>

<script
src="https://cdn.jsdelivr.net/npm/@stackblitz/sdk@1/bundles/sdk.umd.js">
```

```
</script>

<script src="/docs/5.2/assets/js/docs.min.js"></script>

<script>  document.querySelectorAll('.btn-
edit').forEach(btn => {    btn.addEventListener('click',
event => {

    const htmlSnippet = event.target.closest('.bd-
codesnippet').querySelector('.bd-example').innerHTML      const classes =
Array.from(event.target.closest('.bd-codesnippet').querySelector('.bd-
example').classList).join(' ')      const jsSnippet = event.target.closest('.bd-code-
snippet').querySelector('.btnedit').getAttribute('data-sb-js-snippet')

    StackBlitzSDK.openBootstrapSnippet(htmlSnippet, jsSnippet, classes)

  })

  })

  StackBlitzSDK.openBootstrapSnippet = (htmlSnippet, jsSnippet, classes) => {
const markup = `<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link
href="https:\/\/cdn.jsdelivr.net\/npm\/bootstrap@5.2.1\/dist\/css\/bootstrap.min.
c ss" rel="stylesheet">
```

```
    <link href="https://getbootstrap.com/docs/5.2/assets/css/docs.css" rel="stylesheet">

    <title>Bootstrap Example</title>

    <${'script'} src="https:\/\/cdn\.jsdelivr\.net\/npm\/bootstrap@5\.2\.1\/dist\/js\/bootstrap\.bundle\.min\.js"></${'script'}>

  </head>

  <body class="p-3 m-0 border-0 ${classes}">

    <!-- Example Code -->
${htmlSnippet.replace(/^/gm, '    ')}

    <!-- End Example Code -->

  </body> </html>`

    const jsSnippetContent = jsSnippet ? '\/\/ NOTICE!!! Initially embedded in our docs this JavaScript\n\/\/ file contains elements that can help you create reproducible\n\/\/ use cases in StackBlitz for instance\.\n\/\/ In a real project please adapt this content to your needs\.\n\/\/ \u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\n\n\/\*!\n \* JavaScript for Bootstrap\u0027s docs \(https:\/\/getbootstrap\.com\/\)\n \* Copyright 2011\-2022 The Bootstrap Authors\n \* Copyright 2011\-2022 Twitter, Inc\.\n \*
```

global bootstrap: false *\/\n\n\(\(\() =\u003e \{\n \u0027use strict\u0027\n\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Tooltips\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Instantiate all tooltips in a docs or StackBlitz page\n document\.querySelectorAll\(\(\u0027\[data\-bs\toggle=\u0022tooltip\u0022\]\u0027\)\)\n   \.forEach\(\(tooltip =\u003e \{\n   new bootstrap\.Tooltip\(tooltip\)\n   \}\)\)\n\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Popovers\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Instantiate all popovers in a docs or StackBlitz page\n document\.querySelectorAll\(\(\u0027\[data\-bs\toggle=\u0022popover\u0022\]\u0027\)\)\n   \.forEach\(\(popover =\u003e \{\n   new bootstrap\.Popover\(popover\)\n   \}\)\)\n\n \/\/ \-\\-\-\-\-\-\-\-\-\-\-\-\-\-\n \/\/ Toasts\n \/\/ \-\\-\-\-\-\n \/\/ Used by \u0027Placement\u0027 example in docs or StackBlitz\n const toastPlacement = document\.getElementById\(\(\u0027toastPlacement\u0027\)\)\n if \(toastPlacement\) \{\n   document\.getElementById\(\(\u0027selectToastPlacement\u0027\)\.addEventListener\(\u0027change\u0027, function \(\() \{\n   if \(!toastPlacement\.dataset\.originalClass\) \{\n   toastPlacement\.dataset\.originalClass = toastPlacement\.className\n   \}\n\n   toastPlacement\.className = `\$\{toastPlacement\.dataset\.originalClass\}

\$\{this\.value\}`\n   \}\)\n \}\n\n \/\/ Instantiate all toasts in a docs page only\n document\.querySelectorAll\(\(\u0027\.bd\-example \.toast\u0027\)\)\n   \.forEach\(toastNode =\u003e \{\n   const toast = new bootstrap\.Toast\(toastNode, \{\n   autohide: false\n   \}\)\n\n   toast\.show\(\)\n   \}\)\n\n \/\/ Instantiate all toasts in a docs page only\n const toastTrigger = document\.getElementById\(\(\u0027liveToastBtn\u0027\)\)\n const toastLiveExample = document\.getElementById\(\(\u0027liveToast\u0027\)\)\n if \(toastTrigger\) \{\n   toastTrigger\.addEventListener\(\u0027click\u0027, \(\) =\u003e \{\n   const toast = new bootstrap\.Toast\(toastLiveExample\)\n\n   toast\.show\(\)\n   \}\)\n \}\n\n \/\/ \-\n \/\/ Alerts\n \/\/ \-\n \/\/ Used in \u0027Show live toast\u0027 example in docs or StackBlitz\n const alertPlaceholder = document\.getElementById\(\(\u0027liveAlertPlaceholder\u0027\)\)\n   const alertTrigger = document\.getElementById\(\(\u0027liveAlertBtn\u0027\)\)\n\n const appendAlert = \(message, type\) =\u003e \{\n   const wrapper = document\.createElement\(\u0027div\u0027\)\n   wrapper\.innerHTML = \[\n `\u003cdiv class=\u0022alert alert\-\$\{type\} alert\-dismissible\u0022 role=\u0022alert\u0022\u003e`,\n   `   \u003cdiv\u003e\$\{message\}\u003c\/div\u003e`,\n   \u0027   \u003cbutton type=\u0022button\u0022 class=\u0022btn\-close\u0022 data\-bs\dismiss=\u0022alert\u0022

```
aria\label=\u0022Close\u0022\u003e\u003c\/button\u003e\u0027,\n

\u0027\u003c\/div\u003e\u0027\n                              \]\.join\(\u0027\u0027\)\n\n

alertPlaceholder\.append\(wrapper\)\n      \}\n\n     if  \(alertTrigger\)  \{\n

alertTrigger\.addEventListener\(\u0027click\u0027,     \(\)     =\u003e     \{\n

appendAlert\(\u0027Nice, you triggered this alert message!\u0027,

\u0027success\u0027\)\n    \}\)\n  \}\n\n  \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\\-
\-\-\-\-\-\-\-\-\n  \/\/ Checks \u0026 Radios\n  \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\\-\-
\-\-\-\-\-\-\-\-\n  \/\/ Indeterminate checkbox example in docs and StackBlitz\n

document\.querySelectorAll\(\u0027\.bd\-example\-indeterminate

\[type=\u0022checkbox\u0022\]\u0027\)\n    \.forEach\(checkbox =\u003e \{\n

if          \(checkbox\.id\.includes\(\u0027Indeterminate\u0027\)\)            \{\n

checkbox\.indeterminate = true\n    \}\n   \}\)\n\n  \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\\-
\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n  \/\/ Links\n  \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-
\-\-\-\-\-\-\-\n  \/\/ Disable empty links in docs examples only\n

document\.querySelectorAll\(\u0027\.bd\-content

\[href=\u0022#\u0022\]\u0027\)\n              \.forEach\(link   =\u003e   \{\n

link\.addEventListener\(\u0027click\u0027,        event        =\u003e         \{\n

event\.preventDefault\(\)\n    \}\)\n   \}\)\n\n  \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\\-
\-\-\-\-\-\-\-\-\-\-\-\-\n  \/\/ Modal\n  \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-
\-\-\-\-\-\n  \/\/ Modal \u0027Varying modal content\u0027 example in docs and

StackBlitz\n                   const        exampleModal           =

document\.getElementById\(\u0027exampleModal\u0027\)\n  if
```

```
\(exampleModal\) \{\n

exampleModal\.addEventListener\(\u0027show\.bs\.modal\u0027, event

=\u003e \{\n    \/\/ Button that triggered the modal\n    const button =

event\.relatedTarget\n    \/\/ Extract info from data\-bs\-\* attributes\n    const

recipient = button\.getAttribute\(\u0027data\-bs\-whatever\u0027\)\n\n    \/\/

Update the modal\u0027s content\.\n    const modalTitle =

exampleModal\.querySelector\(\u0027\.modal\-title\u0027\)\n    const

modalBodyInput = exampleModal\.querySelector\(\u0027\.modal\-body

input\u0027\)\n\n    modalTitle\.textContent = `New message to

\$\{recipient\}`\n    modalBodyInput\.value = recipient\n  \}\)\n \}\n\n \/\/ \\-

\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n \/\/ Offcanvas\n \/\/ \-\-\-\-\-\-\-\\-\-

\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n   \/\/ \u0027Offcanvas components\u0027

example in docs only\n    const myOffcanvas =

document\.querySelectorAll\(\u0027\.bd\-example\-offcanvas

\.offcanvas\u0027\)\n  if \(myOffcanvas\) \{\n

myOffcanvas\.forEach\(offcanvas =\u003e \{\n

offcanvas\.addEventListener\(\u0027show\.bs\.offcanvas\u0027, event =\u003e

\{\n      event\.preventDefault\(\)\n    \}, false\)\n  \}\)\n \}\n\}\)\(\)\n' : null

const project = {

  files: {

    'index.html': markup,
```

```
      'index.js': jsSnippetContent

    },

    title: 'Bootstrap Example',     description: `Official example

from ${window.location.href}`,     template: jsSnippet ?

'javascript' : 'html',     tags: ['bootstrap']


  }

  StackBlitzSDK.openProject(project, { openFile: 'index.html' })

 }

</script>

 </body>

</html>
```

Home.html

```
  {% extends "base.html" %}

{% block head %}

<title>Home Page</title>

{% endblock %}

{% block body %}

<h1>Welcome to Home Page</h1>

<p> This is a paragraph</p>

{% endblock %}
```

Signin.html

```
<!doctype html>

<html lang="en">

  <head>

    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width, initial-scale=1">
<meta name="description" content="">

    <meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap
contributors">

    <meta name="generator" content="Hugo 0.84.0">

    <title>Sign In</title>

    <link rel="canonical"
href="https://getbootstrap.com/docs/5.0/examples/signin/">

    <link href="https://getbootstrap.com/docs/5.0/assets/css/docs.css"
rel="stylesheet">

    <!-- Bootstrap core CSS -->

    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+
fzT" crossorigin="anonymous">

    <!-- Favicons -->

<link rel="apple-touch-icon" href="/docs/5.0/assets/img/favicons/apple-
touchicon.png" sizes="180x180">
```

```html
<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-32x32.png" sizes="32x32" type="image/png">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-16x16.png" sizes="16x16" type="image/png">

<link rel="manifest" href="/docs/5.0/assets/img/favicons/manifest.json">

<link rel="mask-icon" href="/docs/5.0/assets/img/favicons/safari-pinnedtab.svg" color="#7952b3">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon.ico">

<meta name="theme-color" content="#7952b3">

  <style>

    .bd-placeholder-img {

font-size:        1.125rem;

text-anchor: middle;

    -webkit-user-select: none;

-moz-user-select:         none;

user-select: none;

    }

    @media (min-width: 768px) {

.bd-placeholder-img-lg {        font-size: 3.5rem;

    }

    }

  </style>
```

```html
<!-- Custom styles for this template -->

<link href="static/sheets/signin.css" rel="stylesheet">

<link href="static/sheets/colors.css" rel="stylesheet">

</head>

<body class="text-center">

    <nav class="navbar navbar-dark bg-dark fixed-top">

      <div class="container-fluid">

        <a class="navbar-brand" href="#">News Tracker</a>

        <button class="navbar-toggler" type="button" data-
bstoggle="offcanvas" data-bs-target="#offcanvasDarkNavbar"
ariacontrols="offcanvasDarkNavbar">

          <span class="navbar-toggler-icon"></span>

        </button>

        <div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1"
id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">

          <div class="offcanvas-header">

            <h5 class="offcanvas-title"
id="offcanvasDarkNavbarLabel">Profile</h5>

            <button type="button" class="btn-close btn-close-white" data-
bsdismiss="offcanvas" aria-label="Close"></button>

          </div>

          <div class="offcanvas-body">

            <ul class="navbar-nav justify-content-end flex-grow-1 pe-3">
```

```html
            <li class="nav-item">

              <a class="nav-link active" aria-current="page"
href="home.html">Home</a>

            </li>

            <li class="nav-item">

              <a class="nav-link" href="base.html">Fetch News</a>

            </li>

            <li class="nav-item">

              <a class="nav-link" href="about.html">About Us</a>

            </li>

            <li class="nav-item">

              <a class="nav-link" href="signin.html">Sign In</a>

            </li>

            <li class="nav-item">

              <a class="nav-link" href="signup.html">Sign Up</a>

            </li>

          </div>

        </div>

      </div>

    </nav>

<main class="form-signin  ">

  <form action = "{{ url_for("getUser")}}" method="POST">
```

```html
<img class="mb-4" src="static/images/sign in.svg.png" alt="" width="72" height="57">

<h1 class="h3 mb-3 fw-normal white">Please sign in</h1>

<div class="form-floating">

  <input type="email" class="form-control" id="floatingInput" name = "uname" placeholder="name@example.com">

  <label for="floatingInput">Email address</label>

</div>

<br>

<div class="form-floating">

  <input type="password" class="form-control" id="floatingPassword" name = "upwd" placeholder="Password">

  <label for="floatingPassword">Password</label>

</div>

<div class="checkbox mb-3">

  <label class="white">

    <input type="checkbox" value="remember-me"> Remember me

  </label>

</div>

<button class="w-100 btn btn-lg bd-purple-500 btn-outline-warning" type="submit">Sign in</button><br><br>

<p class="white"><b>Forgot Password? <a href="forpass.html">Change here</b></a></p>
```

```html
<p class="white"><b>Don't have an account? <a href="signup.html">Sign Up</b></a></p>

<p class="mt-5 mb-3 text-muted"><b>&copy; 2022–2023</b></p>

<p class="white"><b>Or</b></p>

<p class="white"><b>Sign in with <img src="static/images/golo..png" alt="" width="40" height="40"></b></p>

  </form>
</main>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js" integrity="sha384-u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOmMi466C8" crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/@docsearch/js@3"></script>

<script src="https://cdn.jsdelivr.net/npm/@stackblitz/sdk@1/bundles/sdk.umd.js">

</script>

<script src="/docs/5.2/assets/js/docs.min.js"></script>

<script>   document.querySelectorAll('.btn-edit').forEach(btn => {

btn.addEventListener('click', event => {       const htmlSnippet =

event.target.closest('.bd-codesnippet').querySelector('.bd-example').innerHTML

const classes = Array.from(event.target.closest('.bd-codesnippet').querySelector('.bd-example').classList).join(' ')      const jsSnippet
```
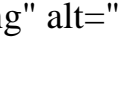
```
= event.target.closest('.bd-code-
snippet').querySelector('.btnedit').getAttribute('data-sb-js-snippet')

      StackBlitzSDK.openBootstrapSnippet(htmlSnippet, jsSnippet, classes)

    })

  })

  StackBlitzSDK.openBootstrapSnippet = (htmlSnippet, jsSnippet, classes) => {
const markup = `<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link
href="https:\/\/cdn.jsdelivr.net\/npm\/bootstrap@5.2.1\/dist\/css\/bootstrap.min.
c ss" rel="stylesheet">
    <link href="https://getbootstrap.com/docs/5.2/assets/css/docs.css"
rel="stylesheet">
    <title>Bootstrap Example</title>
    <${'script'}
src="https:\/\/cdn\.jsdelivr\.net\/npm\/bootstrap@5\.2\.1\/dist\/js\/bootstrap\.bun
dle\.min\.js"></${'script'}>
  </head>
  <body class="p-3 m-0 border-0 ${classes}">
    <!-- Example Code -->
```

```
${htmlSnippet.replace(/^/gm, '    ')}

    <!-- End Example Code -->

  </body>

</html>`    const jsSnippetContent = jsSnippet ? '\/\/ NOTICE!!! Initially embedded in our docs this JavaScript\n\/\/ file contains elements that can help you create reproducible\n\/\/ use cases in StackBlitz for instance\.\n\/\/ In a real project please adapt this content to your needs\.\n\/\/ \u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\u002b\n\n\/\*!\n \* JavaScript for Bootstrap\u0027s docs \(https:\/\/getbootstrap\.com\/\)\n \* Copyright 2011\-2022 The Bootstrap Authors\n \* Copyright 2011\-2022 Twitter, Inc\.\n \* Licensed under the Creative Commons Attribution 3\.0 Unported License\.\n \* For details, see https:\/\/creativecommons\.org\/licenses\/by\/3\.0\/\.\n \*\/\n\n\/\* global bootstrap: false \*\/\n\n\(\(\) =\u003e \{\n  \u0027use strict\u0027\n\n  \/\/ \-\-\-\-\-\-\-\-\-\n  \/\/ Tooltips\n  \/\/ \-\-\-\-\-\-\-\-\-\n  \/\/ Instantiate all tooltips in a docs or StackBlitz page\n    document\.querySelectorAll\(\u0027\[data\-bs\toggle=\u0022tooltip\u0022\]\u0027\)\n    \.forEach\(tooltip =\u003e \{\n    new bootstrap\.Tooltip\(tooltip\)\n    \}\)\n\n  \/\/ \-\-\-\-\-\-\-\-\-\n  \/\/ Popovers\n  \/\/ \-\-\-\-\-\-\-\-\-\n  \/\/ Instantiate all popovers in a docs or StackBlitz page\n    document\.querySelectorAll\(\u0027\[data\-bs\toggle=\u0022popover\u0022\]\u0027\)\n    \.forEach\(popover =\u003e \{\n    new bootstrap\.Popover\(popover\)\n    \}\)\n\n  \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n  \/\/ Toasts\n  \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n  \/\/ Used by \u0027Placement\u0027 example in docs or StackBlitz\n  const toastPlacement = document\.getElementById\(\u0027toastPlacement\u0027\)\n  if
```

\(toastPlacement\) \{\n   document\.getElementById\(\(\u0027selectToastPlacement\u0027\)\)\.addEventListener\(\u0027change\u0027, function \(\) \{\n   if \(!toastPlacement\.dataset\.originalClass\) \{\n   toastPlacement\.dataset\.originalClass = toastPlacement\.className\n   \}\n\n   toastPlacement\.className = `\$\{toastPlacement\.dataset\.originalClass\} \$\{this\.value\}`\n   \}\)\n \}\n\n \/\/ Instantiate all toasts in a docs page only\n document\.querySelectorAll\(\(\u0027\.bd\-example \.toast\u0027\)\)\n   \.forEach\(toastNode =\u003e \{\n   const toast = new bootstrap\.Toast\(toastNode, \{\n   autohide: false\n   \}\)\n\n   toast\.show\(\)\n   \}\)\n\n \/\/ Instantiate all toasts in a docs page only\n const toastTrigger = document\.getElementById\(\(\u0027liveToastBtn\u0027\)\)\n const toastLiveExample = document\.getElementById\(\(\u0027liveToast\u0027\)\)\n if \(toastTrigger\) \{\n   toastTrigger\.addEventListener\(\(\u0027click\u0027, \(\) =\u003e \{\n   const toast = new bootstrap\.Toast\(toastLiveExample\)\n\n   toast\.show\(\)\n   \}\)\n \}\n\n \/\/ \-\n \/\/ Alerts\n \/\/ \-\n \/\/ Used in \u0027Show live toast\u0027 example in docs or StackBlitz\n const alertPlaceholder = document\.getElementById\(\(\u0027liveAlertPlaceholder\u0027\)\)\n const alertTrigger = document\.getElementById\(\(\u0027liveAlertBtn\u0027\)\)\n\n const appendAlert = \(message, type\) =\u003e \{\n   const wrapper = document\.createElement\(\(\u0027div\u0027\)\)\n   wrapper\.innerHTML = \[\n   `\u003cdiv class=\u0022alert alert\-\$\{type\} alert\-dismissible\u0022 role=\u0022alert\u0022\u003e`,\n   `   \u003cdiv\u003e\$\{message\}\u003c\/div\u003e`,\n   \u0027   \u003cbutton type=\u0022button\u0022 class=\u0022btn\-close\u0022 data\-bs\dismiss=\u0022alert\u0022 aria\label=\u0022Close\u0022\u003e\u003c\/button\u003e\u0027,\n

\u0027\u003c\/div\u003e\u0027\n   \]\.join\(\u0027\u0027\)\n\n   alertPlaceholder\.append\(wrapper\)\n   \}\n\n   if \(alertTrigger\) \{\n   alertTrigger\.addEventListener\(\u0027click\u0027,   \(\)   =\u003e   \{\n   appendAlert\(\u0027Nice, you triggered this alert message!\u0027, \u0027success\u0027\)\n   \}\)\n \}\n\n \/\/ \-\\-\-\-\-\-\-\-\-\n \/\/ Checks \u0026 Radios\n \/\/ \-\\-\-\-\-\-\-\-\-\-\-\-\-\n \/\/ Indeterminate checkbox example in docs and StackBlitz\n document\.querySelectorAll\(\u0027\.bd\-example\-indeterminate \[type=\u0022checkbox\u0022\]\u0027\)\n   \.forEach\(checkbox =\u003e \{\n   if   \(checkbox\.id\.includes\(\u0027Indeterminate\u0027\)\)   \{\n   checkbox\.indeterminate = true\n   \}\n   \}\)\n\n \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n \/\/ Links\n \/\/ \-\n \/\/ Disable empty links in docs examples only\n document\.querySelectorAll\(\u0027\.bd\-content \[href=\u0022#\u0022\]\u0027\)\n   \.forEach\(link   =\u003e   \{\n   link\.addEventListener\(\u0027click\u0027,   event   =\u003e   \{\n   event\.preventDefault\(\)\n   \}\)\n   \}\)\n\n \/\/ \-\\-\-\-\-\-\-\-\-\-\-\-\-\-\n \/\/ Modal\n \/\/ \-\n \/\/ Modal \u0027Varying modal content\u0027 example in docs and StackBlitz\n   const   exampleModal   = document\.getElementById\(\u0027exampleModal\u0027\)\n if \(exampleModal\)   \{\n exampleModal\.addEventListener\(\u0027show\.bs\.modal\u0027,   event =\u003e \{\n   \/\/ Button that triggered the modal\n   const button = event\.relatedTarget\n   \/\/ Extract info from data\-bs\-* attributes\n   const recipient = button\.getAttribute\(\u0027data\-bs\-whatever\u0027\)\n\n   \/\/ Update   the   modal\u0027s   content\.\n   const modalTitle = exampleModal\.querySelector\(\u0027\.modal\-title\u0027\)\n   const

```
modalBodyInput = exampleModal\.querySelector\(\u0027\.modal\-body input\u0027\)\)\n\n    modalTitle\.textContent = `New message to \$\{recipient\}`\n    modalBodyInput\.value = recipient\n    \})\)\n  \}\n\n  \/\/ \\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n  \/\/ Offcanvas\n  \/\/ \-\-\-\-\-\-\-\\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n  \/\/ \u0027Offcanvas components\u0027 example in docs only\n  const myOffcanvas = document\.querySelectorAll\(\u0027\.bd\-example\-offcanvas \.offcanvas\u0027\)\n  if \(myOffcanvas\) \{\n    myOffcanvas\.forEach\(offcanvas =\u003e \{\n      offcanvas\.addEventListener\(\u0027show\.bs\.offcanvas\u0027, event =\u003e \{\n        event\.preventDefault\(\)\n      \}, false\)\n    \}\)\n  \}\n\}\)\(\)\n' : null

const project = {

    files: {

      'index.html': markup,

      'index.js': jsSnippetContent

    },

    title: 'Bootstrap Example',    description: `Official example from ${window.location.href}`,    template: jsSnippet ? 'javascript' : 'html',    tags: ['bootstrap']

  }

  StackBlitzSDK.openProject(project, { openFile: 'index.html' })

 }
</script>
  </body>
```

</html>

Signup.html

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="Mark Otto, Jacob Thornton, and Bootstrap contributors">
    <meta name="generator" content="Hugo 0.84.0">
    <title>Sign Up</title>
    <link rel="stylesheet" type="text/css" href="sign in.css">
    <link rel="canonical" href="https://getbootstrap.com/docs/5.0/examples/checkout/">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">
    <link rel="preconnect" href="https://fonts.gstatic.com">
      <link
```

```html
href="https://fonts.googleapis.com/css2?family=Noto+Sans+JP:wght@300&display=swap" rel="stylesheet">

    <link rel="preconnect" href="https://fonts.gstatic.com">

    <link
href="https://fonts.googleapis.com/css2?family=Nunito+Sans:ital@1&display=swap" rel="stylesheet">

    <link rel="preconnect" href="https://fonts.gstatic.com">

    <link
href="https://fonts.googleapis.com/css2?family=Oxygen:wght@300&display=swap" rel="stylesheet">

    <link rel="preconnect" href="https://fonts.gstatic.com">

    <link
href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@500&display=swap" rel="stylesheet">

  <!-- Bootstrap core CSS -->

  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
iYQeCzEYFbKjA/T2uDLTpkwGzCiq6soy8tYaI1GyVh/UjpbCx/TYkiZhlZB6+
fzT" crossorigin="anonymous">

  <!-- Favicons -->

<link rel="apple-touch-icon" href="/docs/5.0/assets/img/favicons/apple-touchicon.png" sizes="180x180">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-32x32.png" sizes="32x32" type="image/png">
```

```html
<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon-16x16.png" sizes="16x16" type="image/png">

<link rel="manifest" href="/docs/5.0/assets/img/favicons/manifest.json">

<link rel="mask-icon" href="/docs/5.0/assets/img/favicons/safari-pinnedtab.svg" color="#7952b3">

<link rel="icon" href="/docs/5.0/assets/img/favicons/favicon.ico">

<meta name="theme-color" content="#7952b3">

    <style>

        table{              text-align:
center;            border:3px solid
black;          border-collapse:
collapse;           width: 100%;
height: 90px;
        }           th{
border-collapse: collapse;
border: 2px solid black;
font-family: 'Oxygen', serif;
font-weight: 35px;              font-
size: 25px;             height: 45px;
width: 60px;            color: black;

        }           td{
border-collapse: collapse;
```

```css
border: 2px solid white;
width: 60px;           height: 45px;
font-size:20px;            font-
weight: 67px;          font-family:
'Oxygen', serif;
background-color: white;
color: black;
    }
    .tabrow
    {
        background-color: #00ff80;
    }
    .tabdat{
background-color: black;
color: white;          font-
weight: bold;
    }
</style>
<link rel="stylesheet" href="colors.css">
    <!-- Custom styles for this template -->
  </head>
<body class="bd-yellow-600">
```

```html
<nav class="navbar navbar-dark bg-dark fixed-top">

  <div class="container-fluid">

    <a class="navbar-brand" href="#">News Tracker</a>

    <button class="navbar-toggler" type="button" data-bs-toggle="offcanvas"
data-bs-target="#offcanvasDarkNavbar" ariacontrols="offcanvasDarkNavbar">

      <span class="navbar-toggler-icon"></span>

    </button>

    <div class="offcanvas offcanvas-end text-bg-dark" tabindex="-1"
id="offcanvasDarkNavbar" aria-labelledby="offcanvasDarkNavbarLabel">

      <div class="offcanvas-header">

        <h5 class="offcanvas-title"
id="offcanvasDarkNavbarLabel">Profile</h5>

        <button type="button" class="btn-close btn-close-white" data-
bsdismiss="offcanvas" aria-label="Close"></button>

      </div>

      <div class="offcanvas-body">

        <ul class="navbar-nav justify-content-end flex-grow-1 pe-3">

          <li class="nav-item">

            <a class="nav-link active" aria-current="page"
href="home.html">Home</a>

          </li>

          <li class="nav-item">

            <a class="nav-link" href="base.html">Fetch News</a>
```

```html
        </li>

        <li class="nav-item">

          <a class="nav-link" href="about.html">About Us</a>

        </li>

        <li class="nav-item">

          <a class="nav-link" href="signin.html">Sign In</a>

        </li>

        <li class="nav-item">

          <a class="nav-link" href="signup.html">Sign Up</a>

        </li>

      </div>

    </div>

  </div>

</nav>

<div class="container div2">

  <main>

    <div class="py-5 text-center">

      <h2 class="text-white div2">Add User</h2>

    </div>

    <div class="col-md-7 col-lg-8">

      <h4 class="mb-3 text-black">Account Details</h4>

      <form action = "{{ url_for("storedUser")}}" method="POST">
```

```html
<div class="row g-3">

  <!--<div class="col-sm-6">-->

    <!--<label for="firstName" class="form-label text-white">Account Number</label>-->

      <!--<input type="text" class="form-control" id="acc_no" name="acc_no" placeholder="" value="" required>-->

    <!--</div>-->

  <div class="row g-3">

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

      <label for="firstName" class="form-label text-black">First name</label>

        <input type="text" class="form-control" id="firstName" name = "fname" placeholder="" value="" required>

          <div class="invalid-feedback">

            Valid first name is required.

          </div>

        </div>

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

        <label for="lastName" class="form-label text-black">Last name</label>

          <input type="text" class="form-control" id="lastName" name = "lname" placeholder="" value="" required>

            <div class="invalid-feedback">
```

Valid last name is required.

```
        </div>

      </div>

    <div class="col-12">

      <label for="floatingInput" class="form-label
textblack">Email</label>

        <input type="email" class="form-control" id="email"  name = "mail"
placeholder="you@example.com">

      </div>

      <div class="col-12">

        <label for="pass" class="form-label text-black">Password</label>

        <input type="password" class="form-control" id="pass"  name =
"npwd" placeholder="Enter a Valid Password" required>

      </div>

      <div class="col-12">

        <label for="confpass" class="form-label text-black">Confirm
Password</label>

        <input type="password" class="form-control" id="confpass"
name="cpwd" placeholder="Enter a Valid Password" required>

       </div>

      <hr class="my-4">

      <a href="#" alt="didnt connect"><button class="w-100 btn bd-teal-700
btn-outline-dark btn-lg" type="submit">Add Account</button></a>
```

```
    </form>

  </div>

  </div>

</main>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.j
s" integrity="sha384-
u1OknCvxWvY5kfmNBILK2hRnQC3Pr17a+RTT6rIHI7NnikvbZlHgTPOOm
Mi466C8" crossorigin="anonymous"></script>

  <script src="https://cdn.jsdelivr.net/npm/@docsearch/js@3"></script>

  <script
src="https://cdn.jsdelivr.net/npm/@stackblitz/sdk@1/bundles/sdk.umd.js">
</script>

    <script src="/docs/5.2/assets/js/docs.min.js"></script>

  <script>      document.querySelectorAll('.btn-
edit').forEach(btn => {        btn.addEventListener('click',
event => {

      const htmlSnippet = event.target.closest('.bd-
codesnippet').querySelector('.bd-example').innerHTML
const classes = Array.from(event.target.closest('.bd-
codesnippet').querySelector('.bd-example').classList).join(' ')
const jsSnippet = event.target.closest('.bd-code-
snippet').querySelector('.btn-edit').getAttribute('data-sb-js-snippet')
```

```
      StackBlitzSDK.openBootstrapSnippet(htmlSnippet, jsSnippet, classes)

  })

  })

  StackBlitzSDK.openBootstrapSnippet = (htmlSnippet, jsSnippet, classes)
=> {        const markup = `<!doctype

html>

  <html lang="en">

  <head>

    <meta charset="utf-8">

    <meta name="viewport" content="width=device-width, initial-scale=1">

    <link
href="https:\/\/cdn.jsdelivr.net\/npm\/bootstrap@5.2.1\/dist\/css\/bootstrap.min.c
ss" rel="stylesheet">

    <link href="https://getbootstrap.com/docs/5.2/assets/css/docs.css"
rel="stylesheet">

    <title>Bootstrap Example</title>

    <${'script'}
src="https:\/\/cdn\.jsdelivr\.net\/npm\/bootstrap@5\.2\.1\/dist\/js\/bootstrap\.bun
dle\.min\.js"></${'script'}>

  </head>

  <body class="p-3 m-0 border-0 ${classes}">

    <!-- Example Code -->

  ${htmlSnippet.replace(/^/gm, '    ')}
```

<!-- End Example Code -->

</body>   </html>`   const jsSnippetContent = jsSnippet ? '\/\/ NOTICE!!! Initially embedded in our docs this JavaScript\n\/\/ file contains elements that can help you create reproducible\n\/\/ use cases in StackBlitz for instance\.\n\/\/ In a real project please adapt this content to your needs\.\n\/\/ \u002b \u002b\n\n\/\/*!\n * JavaScript for Bootstrap\u0027s docs \(https:\/\/getbootstrap\.com\/\)\n * Copyright 2011\-2022 The Bootstrap Authors\n * Copyright 2011\-2022 Twitter, Inc\.\n * Licensed under the Creative Commons Attribution 3\.0 Unported License\.\n * For details, see https:\/\/creativecommons\.org\/licenses\/by\/3\.0\/\.\n *\/\n\n\/* global bootstrap: false *\/\n\n\(\(\) =\u003e \{\n \u0027use strict\u0027\n\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Tooltips\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Instantiate all tooltips in a docs or StackBlitz page\n document\.querySelectorAll\(\u0027\[data\-bs\toggle=\u0022tooltip\u0022\]\u0027\)\n   \.forEach\(tooltip =\u003e \{\n   new bootstrap\.Tooltip\(tooltip\)\n   \}\)\n\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Popovers\n \/\/ \-\-\-\-\-\-\-\-\-\n \/\/ Instantiate all popovers in a docs or StackBlitz page\n document\.querySelectorAll\(\u0027\[data\-bs\toggle=\u0022popover\u0022\]\u0027\)\n   \.forEach\(popover =\u003e \{\n   new bootstrap\.Popover\(popover\)\n   \}\)\n\n \/\/ \-\n \/\/ Toasts\n \/\/ \-\n \/\/ Used by \u0027Placement\u0027 example in docs or StackBlitz\n const toastPlacement = document\.getElementById\(\u0027toastPlacement\u0027\)\n if \(toastPlacement\) \{\n   document\.getElementById\(\u0027selectToastPlacement\u0027\)\.addEventLis

tener\(\u0027change\u0027, function \(\) \{\n   if \(!toastPlacement\.dataset\.originalClass\) \{\n   toastPlacement\.dataset\.originalClass = toastPlacement\.className\n   \}\n\n   toastPlacement\.className = `\$\{toastPlacement\.dataset\.originalClass\} \$\{this\.value\}`\n   \}\)\n \}\n\n \/\/ Instantiate all toasts in a docs page only\n document\.querySelectorAll\(\u0027\.bd\-example \.toast\u0027\)\n   \.forEach\(toastNode =\u003e \{\n   const toast = new bootstrap\.Toast\(toastNode, \{\n   autohide: false\n   \}\)\n\n   toast\.show\(\)\n   \}\)\n\n \/\/ Instantiate all toasts in a docs page only\n const toastTrigger = document\.getElementById\(\u0027liveToastBtn\u0027\)\n const toastLiveExample = document\.getElementById\(\u0027liveToast\u0027\)\n if \(toastTrigger\) \{\n   toastTrigger\.addEventListener\(\u0027click\u0027, \(\) =\u003e \{\n   const toast = new bootstrap\.Toast\(toastLiveExample\)\n\n   toast\.show\(\)\n   \}\)\n \}\n\n \/\/ \-\n \/\/ Alerts\n \/\/ \-\n \/\/ Used in \u0027Show live toast\u0027 example in docs or StackBlitz\n const alertPlaceholder = document\.getElementById\(\u0027liveAlertPlaceholder\u0027\)\n const alertTrigger = document\.getElementById\(\u0027liveAlertBtn\u0027\)\n\n const appendAlert = \(message, type\) =\u003e \{\n   const wrapper = document\.createElement\(\u0027div\u0027\)\n   wrapper\.innerHTML = \[\n   `\u003cdiv class=\u0022alert alert\-\$\{type\} alert\-dismissible\u0022 role=\u0022alert\u0022\u003e`,\n   `  \u003cdiv\u003e\$\{message\}\u003c\/div\u003e`,\n   \u0027 \u003cbutton type=\u0022button\u0022 class=\u0022btn\-close\u0022 data\-bs\dismiss=\u0022alert\u0022 aria\label=\u0022Close\u0022\u003e\u003c\/button\u003e\u0027,\n   \u0027\u003c\/div\u003e\u0027\n   \]\.join\(\u0027\u0027\)\n\n   alertPlaceholder\.append\(wrapper\)\n   \}\n\n   if \(alertTrigger\) \{\n

```
alertTrigger\.addEventListener\(\u0027click\u0027, \(\) =\u003e \{\n
appendAlert\(\u0027Nice, you triggered this alert message!\u0027,
\u0027success\u0027\)\n \}\)\n \}\n\n \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\\-
\-\-\-\-\-\-\-\n \/\/ Checks \u0026 Radios\n \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\\-
\-\-\-\-\-\-\-\n \/\/ Indeterminate checkbox example in docs and StackBlitz\n
document\.querySelectorAll\(\u0027\.bd\-example\-indeterminate
\[type=\u0022checkbox\u0022\]\u0027\)\n \.forEach\(checkbox =\u003e \{\n
if \(checkbox\.id\.includes\(\u0027Indeterminate\u0027\)\) \{\n
checkbox\.indeterminate = true\n \}\n \}\)\n\n \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\\-
\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\n \/\/ Links\n \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-
\-\-\-\-\-\-\-\n \/\/ Disable empty links in docs examples only\n
document\.querySelectorAll\(\u0027\.bd\-content
\[href=\u0022#\u0022\]\u0027\)\n \.forEach\(link =\u003e \{\n
link\.addEventListener\(\u0027click\u0027, event =\u003e \{\n
event\.preventDefault\(\)\n \}\)\n \}\)\n\n \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\\-
\-\-\-\-\-\-\-\-\-\-\-\n \/\/ Modal\n \/\/ \-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-\-
\-\-\-\-\-\n \/\/ Modal \u0027Varying modal content\u0027 example in docs and
StackBlitz\n const exampleModal =
document\.getElementById\(\u0027exampleModal\u0027\)\n if
\(exampleModal\) \{\n
exampleModal\.addEventListener\(\u0027show\.bs\.modal\u0027, event
=\u003e \{\n \/\/ Button that triggered the modal\n const button =
event\.relatedTarget\n \/\/ Extract info from data\-bs\-\* attributes\n const
recipient = button\.getAttribute\(\u0027data\-bs\-whatever\u0027\)\n\n \/\/
Update the modal\u0027s content\.\n const modalTitle =
exampleModal\.querySelector\(\u0027\.modal\-title\u0027\)\n const
modalBodyInput = exampleModal\.querySelector\(\u0027\.modal\-body
input\u0027\)\n\n modalTitle\.textContent = `New message to
```

\$\{recipient\}`\n   modalBodyInput\.value = recipient\n   \}\)\n \}\n\n \/\/ \\-\n \/\/ Offcanvas\n \/\/ \-\-\-\-\-\-\-\\-\n   \/\/ \u0027Offcanvas components\u0027 example in docs only\n   const myOffcanvas = document\.querySelectorAll\(\(\u0027\.bd\-example\-offcanvas \.offcanvas\u0027\)\n if \(myOffcanvas\) \{\n   myOffcanvas\.forEach\(offcanvas =\u003e \{\n   offcanvas\.addEventListener\(\(\u0027show\.bs\.offcanvas\u0027, event =\u003e \{\n   event\.preventDefault\(\(\)\n   \}, false\)\n   \}\)\n \}\n\}\)\(\)\n' : null

const project = {   files: {

     'index.html': markup,

     'index.js': jsSnippetContent

    },

   title: 'Bootstrap Example',   description: `Official example from ${window.location.href}`,   template: jsSnippet ? 'javascript' : 'html',   tags: ['bootstrap']

   }

   StackBlitzSDK.openProject(project, { openFile: 'index.html' })

  }

  </script>

  </body>

  </html>

GitHub IBM-EPBL/**IBM-Project-52299-1660994829**

Project demo link

As of now we couldn't able to complete the project on due time ,

Lots of works are in under process ,so we are not able provide the demo link.