

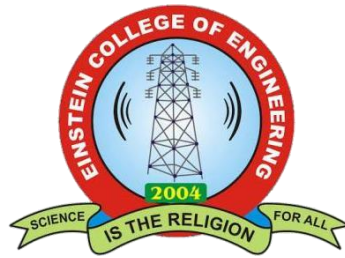
NEWS TRACKER APPPLICATION USING CLOUD

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: PNT2022TMID49885

E.ANANTHI	950619104005
K.ASWIN	950619104010
V.PADMAPRIYA	950619104046
G.PRISCILLA	950619104049



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
EINSTEIN
COLLEGE OF ENGINEERING,
TIRUNELVELI-627 012

BONAFIDE CERTIFICATE

Certified this Report "NEWS TRACKER APPLICATION", for the project, is the bonafide work of Ms E.ANANTHI (950619104005), Mr K.ASWIN (950619104010), Ms V.PADMAPRIYA (950619104046), and Ms G.PRISCILLA (950619104049) who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported here in does not form part of any other thesis or dissertation on the basis of which a degree or award was co-offered on the earlier occasion on this or any other candidate.

SIGNATURE

Dr. T.SURESH THANKRISHNAN
HEAD OF THE DEPARTMENT
COMPUTER SCIENCE AND ENGINEERING
EINSTEIN COLLEGE OF ENGINEERING
TIRUNELVELI-12

SIGNATURE

Dr. T.SURESH THANKRISHNAN
HEAD OF THE DEPARTMENT
COMPUTER SCIENCE AND ENGINEERING
EINSTEIN COLLEGE OF ENGINEERING
TIRUNELVELI-12

ACKNOWLEDGEMENT

We have successfully completed the project with blessings showered on us by god, the almighty, A project of this nature needs co-operation and support from many for successful completion.

We express our heartfelt thanks to **Mr.A.MATHIVANAN, BE., M.sc.,(Agri)**, Managing Trustee of Einstein college of Engineering, Tirunelveli, for his mortal support and device.

Our thanks to **Prof.A.AMUTHAVANAN, BE., M.S (USA).B.L**, Chairman of our college for making necessary arrangements to do this project.

Our heartily thanks to **Prof.EZHILVANAN, MBA.,** Secretary of our college for making necessary arrangements to do this project.

We wish to express our gratitude to **Dr.VELAYUTHAM,M.E,Ph.D.,FIE.** Principal for the support he provided us to carry out this project successfully.

We are very much thankful to **Dr. T.SURESH THANKRISHNAN**, Head of the Department , Electronics and communication Engineering who is always a constant of inspiring us.

We extend our sincere thanks to all teaching and non-teaching staff members and our family members, friends for their help in completing this project.

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

- 1. INTRODUCTION**
 - 1.1 Project Overview
 - 1.2 Purpose
 - 2. LITERATURE SURVEY**
 - 2.1 Existing problem
 - 2.2 References
 - 2.3 Problem Statement Definition
 - 3. IDEATION & PROPOSED SOLUTION**
 - 3.1 Empathy Map Canvas
 - 3.2 Ideation & Brainstorming
 - 3.3 Proposed Solution
 - 3.4 Problem Solution fit
 - 4. REQUIREMENT ANALYSIS**
 - 4.1 Functional requirement
 - 4.2 Non-Functional requirements
 - 5. PROJECT DESIGN**
 - 5.1 Data Flow Diagrams
 - 5.2 Solution & Technical Architecture
 - 5.3 User Stories
 - 6. PROJECT PLANNING & SCHEDULING**
 - 6.1 Sprint Planning & Estimation
 - 6.2 Sprint Delivery Schedule
 - 6.3 Reports from JIRA
 - 7. CODING & SOLUTIONING**
 - 7.1 Feature 1
 - 7.2 Feature 2
 - 7.3 Database Schema (if Applicable)
 - 8. TESTING**
 - 8.1 Test Cases
 - 8.2 User Acceptance Testing
 - 9. RESULTS**
 - 9.1 Performance Metrics
 - 10. ADVANTAGES & DISADVANTAGES**
 - 11. CONCLUSION**
 - 12. FUTURE SCOPE**
 - 13. APPENDIX**
- Source Code GitHub & Project Demo Link

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.

1.1 Project 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.

1.2 Purpose

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/>

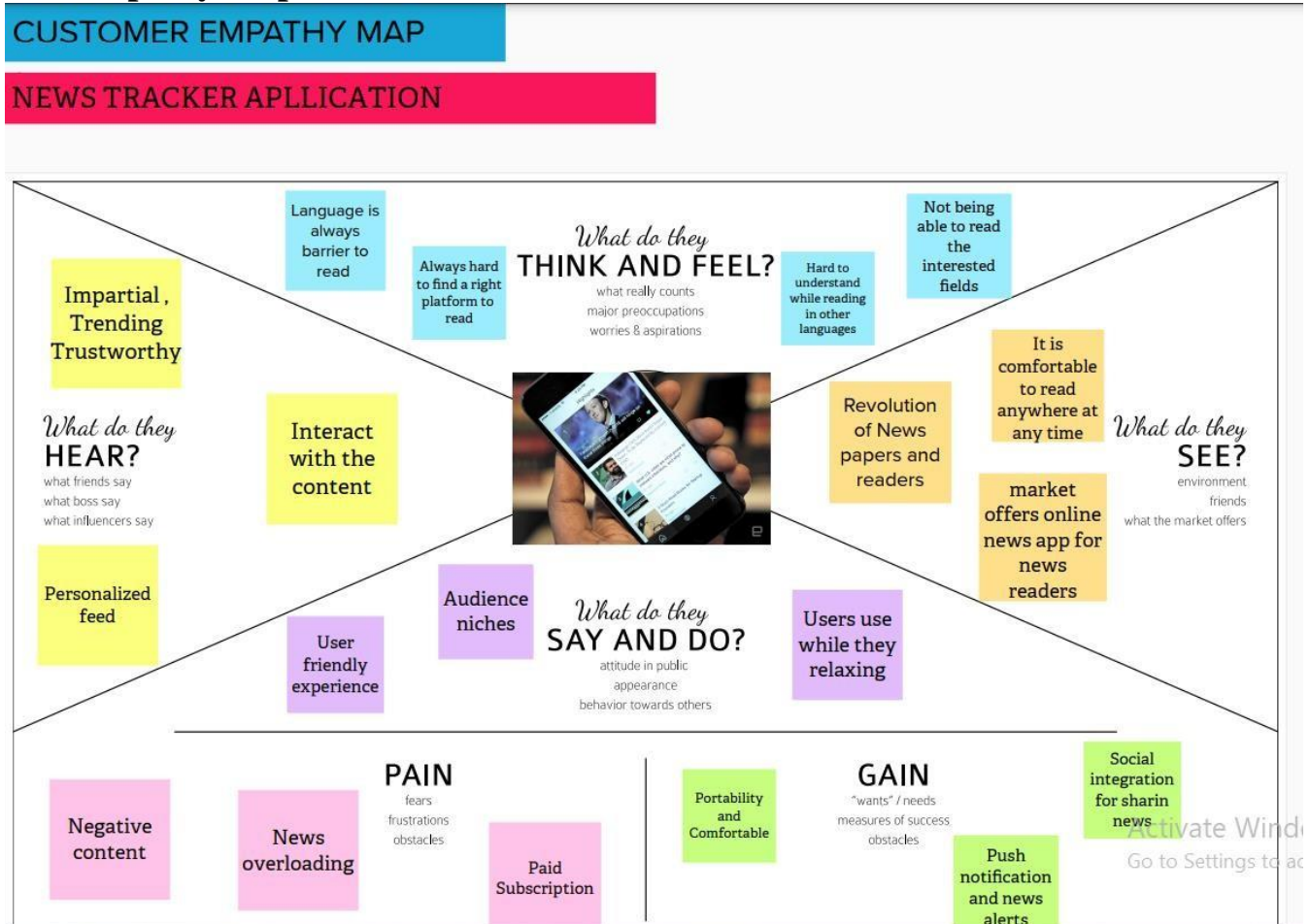
<https://www.ripublication.com/>

2.3 Problem Statement Definition

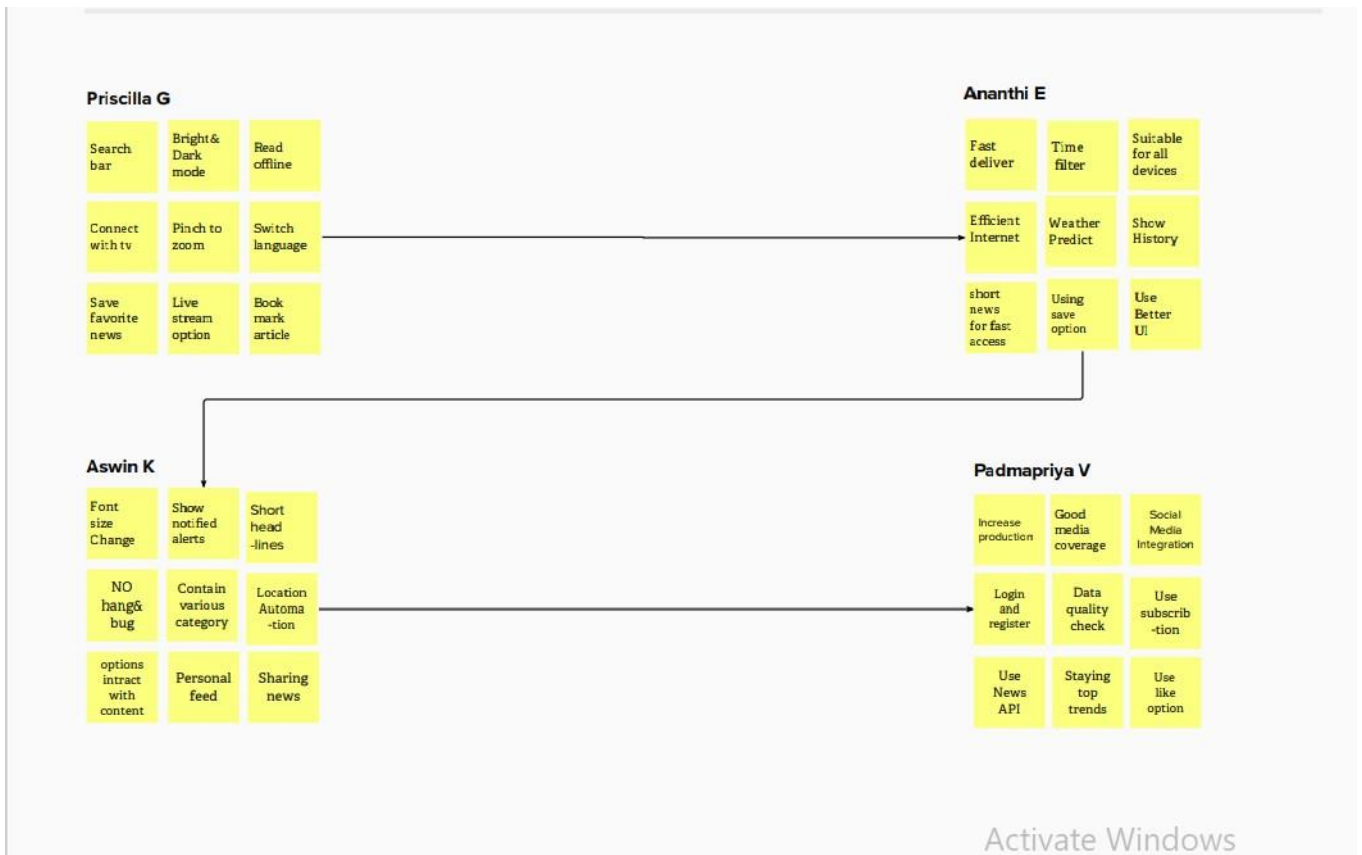
1. The user needs a way to organize the news on daily basis so that he can cope up with daily events in his tight schedule.
2. The user wants to read news only about particular topics so that he can be informed about his interest.
3. The user wants to get informed from only certified news outlets .
4. 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



3.2 Ideation & Brainstorming



3.3 Proposed Solution

S.NO	PARAMETERS	DESCRIPTION
1.	Problem statement (problems to be solved)	To help customer to access NEWS using Cloud Application.
2.	Idea/solution description	Making the customer to access news in few touch, providing authorized news, cloud based app, With tons of features to make it exciting
3.	Novelty/Uniqueness	Profile based account, like and share, download blogs, links to original publishers are provided.
4.	Social Impact/Customer satisfaction	Customer Satisfaction, Customer can follow NEWS publishers, like and share feature.
5.	Business Model (Revenue)	Showing ad, Showing a particular publishers NEWS and blogs.
6.	Scalability of the solution	Enabling the system to scalable according to the customers need

3.4 Problem Solution fit

Project Title: News Tracker Application		Project Design Phase-I - Solution Fit Template		Team ID: PNT2022TMID49885	
Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS Who is your customer? i.e. working parents of 0-5 y.o. kids <div>1. News reader 2. People</div>	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. <div>1. It will consume more time 2. It will consume more cost 3. Network connection</div>	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. <div>People may use either newspaper or social media or youtube channels to know the news</div>	Explore AS, differentiate	
	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. <div>1. People can get simultaneous breaking news 2. We can avoid fake news 3. News received at correct time</div>	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. <div>In a busy world people not have allocate time for reading newspaper and watching news channels</div>	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) <div>People follow youtube channels but this will not possible to know all news. People buy a news paper they don't read all news because of timecons</div>		
Identify strong TR & EM	3. TRIGGERS TR What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news. <div>Reading about a more efficient solution in the news</div>	10. YOUR SOLUTION SL If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour. <div>Making separate space for each category of news. People select news category and know all about the news.</div>	8. CHANNELS of BEHAVIOUR CH 8.1 ONLINE What kind of actions do customers take online? Extract online channels from #7 <div>In online people know news faster through network</div>	Identify strong TR & EM	
	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. <div>People will know the news faster</div>		8.2 OFFLINE What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development. <div>In offline people must allocate time for reading news papers.</div>		

Activate

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Login	User using his /her device to login to the app/website for reading news
FR-4	User search	For getting news he/she must need to search the news in the given search bar, Which will collect the news according to his search from the online sources like blog, e-news paper,Articles etc
FR-5	User Reads	Gathered news will be shown to the user in short lines/ summary manner. Along with news Pictures, hyperlinks for the blog, article Or original publisher of the news is given
FR-6	User features	User can have lot of features like Like and share and Post features.

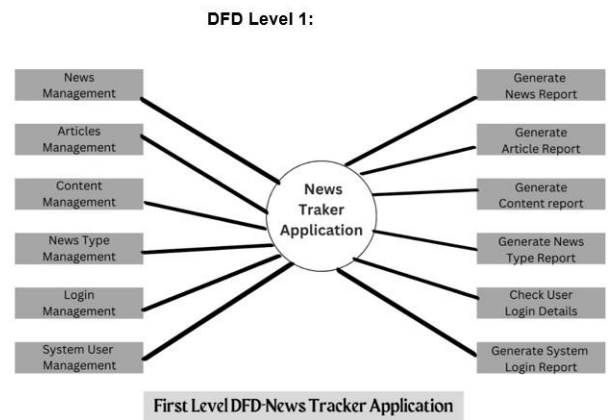
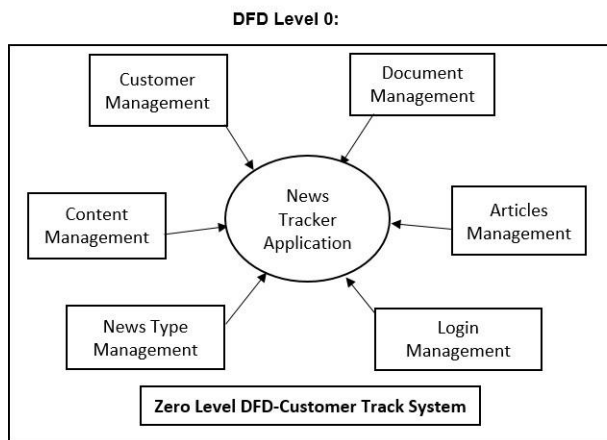
4.2 Non-functional Requirements:

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

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	For getting news he/she must need to search the news in the given search bar, Which will collect the news according to his search from the online sources like blog, e-news paper,Articles etc. Gathered news will be shown to the user in short lines/ summary manner. Along with news Pictures, hyperlinks for the blog, article Or original publisher of the news is given
NFR-2	Security	User's data for the Registration and data collected from his activity are maintained in high security and must be used for only product improvement. Disclosing of his/her detail is considered as a crime hence they can take any legal actions against the service provider.
NFR-3	Reliability	News gathered and presented to the user are from only authorized publishers, and the sources are authentic. User can relay on this app for original news .
NFR-4	Performance	24*7 service with best cloud service, There will be no performance issue to the users.
NFR-5	Availability	Can download from Playstore and online websites.
NFR-6	Scalability	Hence it is cloud based system Scalability will be no issue because Scalability is the prior feature of the cloud based services.

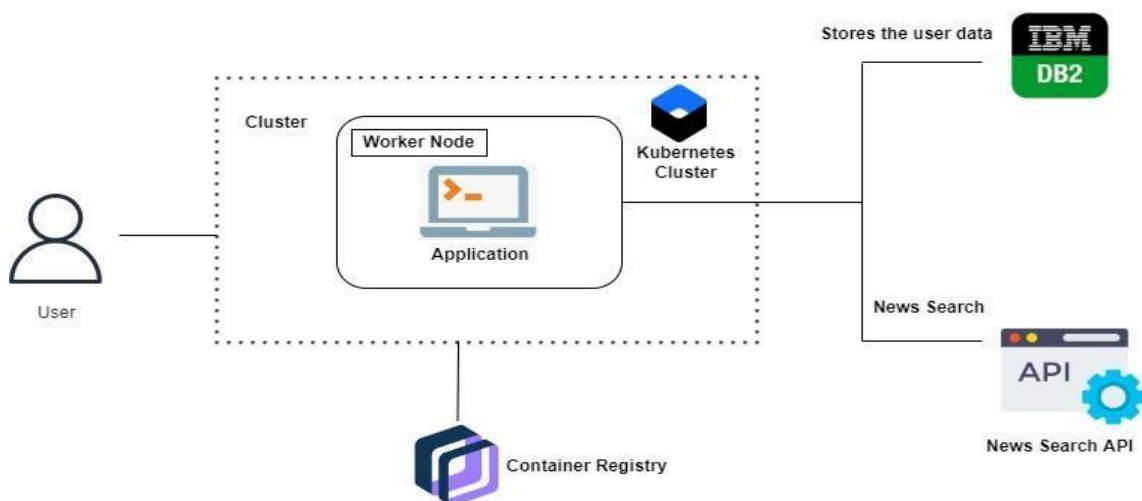
5. PROJECT DESIGN

5.1 Data Flow Diagrams



Activate
Go to Setting

5.2 Solution & Technical Architecture



5.3 User Stories

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 Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail		High	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard					
Customer (Web user)	Log in	USN-6	As a user, I can access the application through website	I can access my account / dashboard	High	
Customer Care Executive	24x7 service	USN-7	As a User, I may need help in using the app/website	Any help need	High	
Administrator	Owner/Chief executive	–	He/she will access the datas in the app	Product behaviour/Improvement purposes	Low	

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	Literature survey on the selected project & gathering information by referring the, technical papers, research publications etc.	2 Sept 2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements	10 Sept 2022

Ideation Brain Storming	List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	15 Sept 2022
Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	23 Sept 2022
Problem Solution Fit	Prepare problem solution fit document	28 Sept 2022
Solution Architecture	Prepare solution architecture document.	5 October 2022
Customer Journey	Prepare the customer journey maps to	7 October 2022

	understand the user interactions & experiences with the application (entry to exit).	
Solution Requirement	Prepare the functional requirement document	8 October 2022
Data Flow Diagrams	Draw the data flow diagrams and submit for review.	10 October 2022
Technology Architecture	Prepare the technology architecture diagram.	15 October 2022
Prepare Milestone & Activity List	Prepare the milestones & activity list of the project.	26 October 2022
Project Development - Delivery of Sprint-1, 2, 3 & 4	Develop & submit the developed code by testing it.	IN PROGRESS..

6.2 Sprint Delivery Schedule

Product Backlog, Sprint Schedule, and Estimation

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 to the Application by using my Email and password.	10	High	Abishek Arun kumar, Ram kumar.
Sprint-1	Confirmation mail	USN-2	Confirmation Mail will be received by the user, that he is registered for the Application.	10	High	Abishek Arun kumar, Ram kumar.
Sprint-1	Login	USN-3	As a user I need to login to my registered mail to access the Application.	10	Low	Abishek Arun kumar, Ram kumar.
Sprint-2	Search box	USN-4	User need search box to search the news	20	Medium	Manikandan, Gokul
Sprint-3	News gathering	USN-5	User need to get result for his search.	20	High	Ram Kumar, Manikandan
Sprint-4	Chatbot (IBM Watson)	USN-6	If user have any doubt in using the Application, Chat bot is available in the Application to help the user.	20	High	Abishek, Arun kumar, Ram kumar, Manikandan, Gokul
Sprint-4	IBM Db2	USN-7	Application is integrated with DB2	10	High	Abishek, Arun kumar.

6.3 Reports from JIRA

7. CODING & SOLUTIONING

Forgot password-

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/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">
```

```
<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-align: 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>
      <button class="navbar-toggler" type="button" data-bs-toggle="offcanvas" data-bs-
target="#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-bs-
dismiss="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>

<main class="form-signin">
    <form action = "{{ url_for('getUser')}}" method="POST">
        
        <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('.bd-example').innerHTML
```

```
    const classes = Array.from(event.target.closest('.bd-code-snippet').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.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(/^\s/gm, ' ')}

  <!-- End Example Code -->

</body>

</html>
```

[illegible]

```

\u0027Placement\u0027 example in docs or StackBlitz\n const toastPlacement =
document.getElementById(\u0027toastPlacement\u0027)\n if (\u0027toastPlacement\u0027) {\n
document.getElementById(\u0027selectToastPlacement\u0027).addEventListener(\u0027
change\u0027, function () {\n      if (\u0027!toastPlacement.dataset.originalClass\u0027) {\n
toastPlacement.dataset.originalClass = toastPlacement.className\n      }\n
toastPlacement.className = `${toastPlacement.dataset.originalClass} ${this.value}`\n
})\n      }\n      \u2219 Instantiate all toasts in a docs page only\n
document.querySelectorAll(\u0027.bd-example .toast\u0027)\n      .forEach((toastNode
=\u003e {\n      const toast = new bootstrap.Toast(toastNode, {\n      autohide: false\n
})\n      toast.show()\n      })\n      \u2219 Instantiate all toasts in a docs page only\n
const toastTrigger = document.getElementById(\u0027liveToastBtn\u0027)\n      const
toastLiveExample = document.getElementById(\u0027liveToast\u0027)\n      if
(\u0027toastTrigger\u0027) {\n      toastTrigger.addEventListener(\u0027click\u0027, () => {\n
const toast = new bootstrap.Toast(toastLiveExample)\n      toast.show()\n      })\n      }\n
\u2219 Alerts\n \u2219 Alerts\n \u2219 Used in \u0027Show live toast\u0027 example in docs or
StackBlitz\n      const alertPlaceholder =
document.getElementById(\u0027liveAlertPlaceholder\u0027)\n      const alertTrigger =
document.getElementById(\u0027liveAlertBtn\u0027)\n      const appendAlert = (message,
type) => {\n      const wrapper = document.createElement(\u0027div\u0027)\n
wrapper.innerHTML = `<div class=\u0022alert alert-${type}\u0022 alert-
dismissible\u0022 role=\u0022alert\u0022\u003e`,\n
\u003cdiv\u003e${message}\u003c/div\u003e`,\n      \u0027\n      \u003cbutton
type=\u0022button\u0022 class=\u0022btn-close\u0022 data-bs-dismiss=\u0022alert\u0022
aria-label=\u0022Close\u0022\u003e\u003cbutton\u003e\u0027,\n
\u0027\u003cdiv\u003e\u0027)\n      ].join(\u0027\u0027)\n
alertPlaceholder.append(wrapper)\n      }\n      if (\u0027alertTrigger\u0027) {\n
alertTrigger.addEventListener(\u0027click\u0027, () => {\n
appendAlert(\u0027Nice, you triggered this alert message!\u0027, \u0027success\u0027)\n
})\n      }\n      \u2219 Checks\n \u2219 Radios\n \u2219 Indeterminate checkbox example in
docs and StackBlitz\n
document.querySelectorAll(\u0027.bd-example-indeterminate
[type=\u0022checkbox\u0022]\u0027)\n      .forEach((checkbox => {\n      if
(\u0027checkbox.id.includes(\u0027Indeterminate\u0027)\u0027) {\n      checkbox.indeterminate =
true\n      }\n      })\n      \u2219 Links\n \u2219 Links\n \u2219 Disable empty links in docs examples
only\n
document.querySelectorAll(\u0027.bd-content [href=\u0022#\u0022]\u0027)\n
.forEach((link => {\n      link.addEventListener(\u0027click\u0027, event => {\n
event.preventDefault()\n      })\n      })\n      \u2219 Modal\n \u2219 Modal\n \u0027Varying modal content\u0027 example in docs and StackBlitz\n
const exampleModal = document.getElementById(\u0027exampleModal\u0027)\n      if
(\u0027exampleModal\u0027) {\n      exampleModal.addEventListener(\u0027show.bs.modal\u0027, event => {\n
\u2219 Button that triggered the modal\n      const button = event.relatedTarget\n      \u2219 Extract info
from data-bs-* attributes\n      const recipient = button.getAttribute(\u0027data-bs-
whatever\u0027)\n      \u2219 Update the modal\u0027s content\n      const modalTitle =

```


8. TESTING

8.1 Test Cases

Test case

| 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 | - | Abishek |
| Sign up | Functional | Login page | Verify user has the option to sign up | Can visible | Yes visible | pass | Successful | - | Abishek |
| Forgot password | Functional | Login page | Verify user has the option to forgot password | Yes the option is available | Option is available | pass | Successful | - | Manikandan |

| | | | | | | | | | |
|--|------------|------------|------------------------------|------------------------------|------------------------------|------|--------------|-------------------------|-----------------|
| Fetch news | Functional | Home page | Verify user can get the news | News will be feed to the app | 404 error | Fail | unsuccessful | App integration problem | Arun,Ram |
| 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 | - | Gokul, Abishek. |

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 Advertisement 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-99de-440d-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 = 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">

    <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" 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-bs-
dismiss="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>
```

```
<div class="container div2">
```

```
  <div class="row">
```

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

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

```
        
```

```
      </a>
```

</div>

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

<h1>Home page</h1>

<p>

<ul class="content">

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

</p>

</div>

</div>

</div>

<div class="container div2">

<div class="row">

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

<h1>About Page</h1>

<p>

<ul class="content">


```
<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">
```

```

```

```
</a>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="container div2">
```

```
<div class="row">
```

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

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

```

```

```
</a>
```

```
</div>
```

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

<h1>Sign Up Page</h1>

<p>

<ul class="content">

<li class="content">Create an account for yourself and
enjoy the privileges given to you

</p>

</div>

</div>

</div>

<div class="container div2">

<div class="row">

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

<h1>Sign In Page</h1>

<p>

<ul class="content">

<li class="content">Log In to your existing account if
you have one

</p>

</div>

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

</div>

</div>

</div>

<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('.bd-example').innerHTML

        const classes = Array.from(event.target.closest('.bd-code-
snippet').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>

```



```

global bootstrap: false
use strict

// Tooltips
// Instantiate all tooltips in a docs or StackBlitz page
document.querySelectorAll('[data-bs-toggle="tooltip"]')
  .forEach((tooltip) => {
    new bootstrap.Tooltip(tooltip)
  })

// Popovers
// Instantiate all popovers in a docs or StackBlitz page
document.querySelectorAll('[data-bs-toggle="popover"]')
  .forEach((popover) => {
    new bootstrap.Popover(popover)
  })

// Toasts
// Used by Placement example in docs or StackBlitz
const toastPlacement =
document.getElementById('toastPlacement')
if (toastPlacement) {
  document.getElementById('selectToastPlacement')
    .addEventListener('change', function () {
      if (!toastPlacement.dataset.originalClass) {
        toastPlacement.dataset.originalClass = toastPlacement.className
        toastPlacement.className = `${toastPlacement.dataset.originalClass} ${this.value}`
      }
    })
  // Instantiate all toasts in a docs page only
  document.querySelectorAll('.bd-example .toast')
    .forEach((toastNode) => {
      const toast = new

```

```

bootstrap.Toast(toastNode, {\n      autohide: false\n    })\n\ntoast.show()\n\n // Instantiate all toasts in a docs page only\nconst toastTrigger = document.getElementById('liveToastBtn')\n\nconst toastLiveExample =\n\ndocument.getElementById('liveToast')\nif (toastTrigger) {\n  toastTrigger.addEventListener('click', () => {\n    const toast = new bootstrap.Toast(toastLiveExample)\n    toast.show()\n  })\n}\n\n // Alerts\n\n // Used in 'Show\nlive toast' example in docs or StackBlitz\nconst alertPlaceholder =\n\ndocument.getElementById('liveAlertPlaceholder')\nconst alertTrigger = document.getElementById('liveAlertBtn')\n\nconst appendAlert = (message, type) => {\n  const wrapper =\n\ndocument.createElement('div')\n  wrapper.innerHTML = [\n    `<div class="alert alert-${type}" alert-dismissable`\n    role="alert">`,\n    `<div>${message}</div>`,\n    `<button` +\n    type="button" class="btn-close" data-bs-dismiss=` +\n    dismiss="alert" aria-label=` +\n    label="Close"></button>`,\n  ].join('\\n')\n\n  alertPlaceholder.append(wrapper)\n  if (alertTrigger) {\n
```

```

alertTrigger\addEventListener(\u0027click\u0027, \() =\u003e {\n
appendAlert(\u0027Nice, you triggered this alert message!\u0027,
\u0027success\u0027)\n  }})\n }\n\n // |---|---|---|---|---|---|---|---|---|---|
|---|---|---|---|---|---|---|---|---|---|
// Checks \u0026 Radios\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 // |---|---|---|---|---|---|---|---|---|---|
|---|---|---|---|---|---|---|---|---|---|
// Links\n // |---|---|---|---|---|---|---|---|---|---|
|---|---|---|---|---|---|---|---|---|---|
// Disable empty links in docs examples only\n
document.querySelectorAll(\u0027.bd-content
[href#\u0022]\u0027)\n  .forEach((link =\u003e {\n
link.addEventListener(\u0027click\u0027, event =\u003e {\n
event.preventDefault()\n    })\n  })\n\n // |---|---|---|---|---|---|---|---|---|---|
|---|---|---|---|---|---|---|---|---|---|
// Modal\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\n\nrecipient = button.getAttribute(\\u0027data-bs-whatever\\u0027\\)\n\nUpdate the modal's content.\n    const modalTitle =\n\nexampleModal.querySelector(\\u0027.modal-title\\u0027)\n    const\n\nmodalBodyInput = exampleModal.querySelector(\\u0027.modal-body\ninput\\u0027\\)\n\n    modalTitle.textContent = `New message to\n\n\\${{recipient}}`\n    modalBodyInput.value = recipient\n    \\})\n    \\}\n\n    \\/\n\n-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|\n    \\ Offcanvas\n    \\|-|-|-|-|-|\n\n-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|\n    \\ Offcanvas components\\u0027\n\nexample in docs only\nconst myOffcanvas =\n\ndocument.querySelectorAll(\\u0027.bd-example-offcanvas\n.offcanvas\\u0027\\)\nif (myOffcanvas) {\n\nmyOffcanvas.forEach(offcanvas => {\n\noffcanvas.addEventListener(\\u0027show.bs.offcanvas\\u0027, event =>\n{\n    event.preventDefault()\n    }, false)\n    })\n    })\n    ' : null\n\nconst project = {\n\nfiles: {\n\n    'index.html': markup,\n\n    'index.js': jsSnippetContent\n\n},
```

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>

<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">
```

```
<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-align: 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>  
  
            <button class="navbar-toggler" type="button" data-bs-  
toggle="offcanvas" data-bs-target="#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-bs-
dismiss="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">
```

[About Us](about.html)

[Sign In](signin.html)

[Sign Up](signup.html)

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

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

</div>

<br>

<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-code-  
snippet').querySelector('.bd-example').innerHTML
```

```
    const classes = Array.from(event.target.closest('.bd-code-  
snippet').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 -->

`

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\\\\\\

[illegible]

```
document.querySelector('[data-bs-toggle=popover][data-bs-trigger=\'hover\']').forEach((popover) => {  
  new bootstrap.Popover(popover)  
})  
  
// Toasts  
// Used by Placement example in docs or StackBlitz  
  
const toastPlacement = document.getElementById('toastPlacement')  
if (toastPlacement) {  
  document.getElementById('selectToastPlacement').addEventListener('change', function () {  
    if (!toastPlacement.dataset.originalClass) {  
      toastPlacement.dataset.originalClass = toastPlacement.className  
      toastPlacement.className = `${toastPlacement.dataset.originalClass} ${this.value}`  
    }  
  })  
}  
  
// Instantiate all toasts in a docs page only  
document.querySelectorAll('.bd-example .toast').forEach((toastNode) => {  
  const toast = new bootstrap.Toast(toastNode, { autohide: false })  
  toast.show()  
})  
  
// Instantiate all toasts in a docs page only  
const toastTrigger = document.getElementById('liveToastBtn')  
const toastLiveExample = document.getElementById('liveToast')  
if (toastTrigger) {  
  toastTrigger.addEventListener('click', () => {  
    const
```

```
toast = new bootstrap.Toast(toastLiveExample) toast.show()

})

// Alerts

// Used in Show
live toast example in docs or StackBlitz
const alertPlaceholder =
document.getElementById('liveAlertPlaceholder')
const alertTrigger = document.getElementById('liveAlertBtn')

const appendAlert = (message, type) => {
  const wrapper =
document.createElement('div')
  wrapper.innerHTML = [
`<div class="alert alert-${type} alert-dismissible"
role="alert">`,
`<div>${message}</div>`,
`<button
type="button" class="btn-close" data-bs-dismiss="alert"
aria-label="Close"></button>`,
].join('')

alertPlaceholder.append(wrapper)

if (alertTrigger) {
  alertTrigger.addEventListener('click', () => {
    appendAlert('Nice, you triggered this alert message!',
'success')
  })
}

// Checks & Radios

// Indeterminate checkbox example in docs and StackBlitz
document.querySelectorAll('bd-example-indeterminate
```

```
[type=\u0022checkbox\u0022]\u0027)\n    \\.forEach\(\checkbox =\u003e {\n\nif \(\checkbox\\.id\\.includes\(\u0027Indeterminate\u0027\)) {\n\ncheckbox\\.indeterminate = true\n    }\n    })\n\n\\/\n-|-|-|-|-|-|-|-|-|-|-|-|\n-|-|-|-|-|-|-|-|-|-|-|\n\\/\nLinks\n\\/\n-|-|-|-|-|-|-|-|-|-|-|-|\n-|-|-|-|-|\n\\/\nDisable empty links in docs examples only\n\ndocument\\.querySelectorAll\(\u0027\\.bd\\-content\n\n[\\href=\u0022#\u0022]\u0027)\n    \\.forEach\(\link =\u003e {\n\nlink\\.addEventListener\(\u0027click\u0027, event =\u003e {\n\nevent\\.preventDefault\()\n    })\n    })\n\n\\/\n-|-|-|-|-|-|-|-|-|-|-|-|\n-|-|-|-|-|-|-|-|-|-|-|\n\\/\nModal\n\\/\n-|-|-|-|-|-|-|-|-|-|-|-|\n-|-|-|-|\n\\/\nModal \u0027Varying modal content\u0027 example in docs and\nStackBlitz\nconst exampleModal =\n\ndocument\\.getElementById\(\u0027exampleModal\u0027)\nif\n\n(exampleModal) {\n\nexampleModal\\.addEventListener\(\u0027show\\.bs\\.modal\u0027, event\n\n=\u003e {\n    \\/\nButton that triggered the modal\n    const button =\n\n    event\\.relatedTarget\n        \\/\nExtract info from data-bs-* attributes\n    const\n\nrecipient = button\\.getAttribute\(\u0027data-bs-whatever\u0027)\n\n    \\/\n\nUpdate the modal's content.\n    const modalTitle =\n\nexampleModal\\.querySelector\(\u0027\\.modal\\-title\u0027)\n    const\n\nmodalBodyInput = exampleModal\\.querySelector\(\u0027\\.modal\\-body\n\ninput\u0027)\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.querySelector(\`#\u0027.bd-example-offcanvas  
\.offcanvas`\n)\n if (myOffcanvas) {\n  
myOffcanvas.forEach((offcanvas => {\n  
offcanvas.addEventListener(\`#\u0027show.bs.offcanvas`, event =>{\n  
\{\n      event.preventDefault()\n    }, false)\n  })\n})\n}) : null  
  
const project = {\n  
  files: {\n  
    'index.html': markup,\n  
    '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="">  
  
  <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">
```

```
<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-align: 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>
            <button class="navbar-toggler" type="button" data-bs-
toggle="offcanvas" data-bs-target="#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-bs-
dismiss="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">

      <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>

  </ul>

</div>

</div>

</div>

</nav>
```

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

  <form action="{ { url_for("login") } }">

    <div class="form-floating">

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

    </div>

    <br>

    <div class="form-floating">

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

    </div>

    <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+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-code-
snippet').querySelector('.bd-example').innerHTML

    const classes = Array.from(event.target.closest('.bd-code-
snippet').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>

<${'script'}
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js"></${'script'}>

</head>

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

  <!-- Example Code -->

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

  <!-- End Example Code -->

</body>

</html>
```

[illegible]

```

global bootstrap: false
use strict

//
// Tooltips
//
// Instantiate all tooltips in a
// docs or StackBlitz page
document.querySelectorAll([data-bs-
toggle='tooltip']
).forEach((tooltip) => {
  new bootstrap.Tooltip(tooltip)
})

//
// Popovers
//
// Instantiate all popovers in a docs or StackBlitz page
document.querySelectorAll([data-bs-
toggle='popover']
).forEach((popover => {
  new bootstrap.Popover(popover)
})

//
// Toasts
//
// Used by Placement example in docs or StackBlitz
const toastPlacement =
document.getElementById('toastPlacement')
if (toastPlacement) {
  document.getElementById('selectToastPlacement')
    .addEventListener('change', function () {
    if (!toastPlacement.dataset.originalClass) {
      toastPlacement.dataset.originalClass = toastPlacement.className
      toastPlacement.className = ` ${toastPlacement.dataset.originalClass}
${this.value}`
    }
  })
  // Instantiate all toasts in a docs page only
  document.querySelectorAll('bd-example .toast')
    .forEach((toastNode => {
      const toast = new
bootstrap.Toast(toastNode, {
        autohide: false
      })
      toast.show()
    })
  // Instantiate all toasts in a docs page only
  const toastTrigger = document.getElementById('liveToastBtn')
  const toastLiveExample =
document.getElementById('liveToast')
  if (toastTrigger) {
    toastTrigger.addEventListener('click', () => {
      const
toast = new bootstrap.Toast(toastLiveExample)
      toast.show()
    })
  }
}

```

```

})\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) => {\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 \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   alertPlaceholder.append(wrapper)\n   if (alertTrigger) {\n     alertTrigger.addEventListener(\u0027click\u0027, () => {\n       appendAlert(\u0027Nice, you triggered this alert message!\u0027,\n         \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).forEach((checkbox => {\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).forEach((link => {\n   link.addEventListener(\u0027click\u0027, event => {\n
```


[illegible]

```
    },  
    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/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/TYkiZhlZB 6+
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">
```

```
<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-align: 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>

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

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

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

```
        
```

```
        <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-code-
snippet').querySelector('.bd-example').innerHTML

const classes = Array.from(event.target.closest('.bd-code-
snippet').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'}>

```



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

```
const toastLiveExample =

document.getElementById(liveToast)
if (toastTrigger) {
  toastTrigger.addEventListener(click, () => {
    const toast = new bootstrap.Toast(toastLiveExample)
    toast.show()
  })
}

// Alerts

// Used in Show live toast example in docs or StackBlitz
const alertPlaceholder =

document.getElementById(liveAlertPlaceholder)
const alertTrigger = document.getElementById(liveAlertBtn)

const appendAlert = (message, type) => {
  const wrapper =

document.createElement(div)
  wrapper.innerHTML = [
`<div class="alert alert-${type}" alert-dismissable`
role="alert">`,
`<div>${message}</div>`,
`<button`
type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>`,
].join(liveToast)

alertPlaceholder.append(wrapper)
if (alertTrigger) {
  alertTrigger.addEventListener(click, () => {
    appendAlert(Nice, you triggered this alert message!,
      success)
  })
}

//
```

\\ Checks \u0026 Radios\\ \\

\\ Indeterminate checkbox example in docs and StackBlitz\\

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 \\ \\

\\ Links\\ \\

\\ Disable empty links in docs examples only\\

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 \\ \\

\\ Modal\\ \\

\\ Modal \u0027Varying modal content\u0027 example in docs and

StackBlitz\\ const exampleModal =

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

(exampleModal) {\n

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

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

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

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

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

```
exampleModal.querySelector(\u0027.modal-title\u0027)\n    const\nmodalBodyInput = exampleModal.querySelector(\u0027.modal-body\ninput\u0027)\n\n    modalTitle.textContent = `New message to\n\n${recipient}`\n    modalBodyInput.value = recipient\n  })\n}\n\n \/\ \/-\n-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|\n \/\ Offcanvas\n \/\ -|-|-|-|-|-|\n-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|\n \/\ \u0027Offcanvas components\u0027\n\nexample in docs only\nconst myOffcanvas =\ndocument.querySelectorAll(\u0027.bd-example-offcanvas\n.offcanvas\u0027)\nif (myOffcanvas) {\nmyOffcanvas.forEach(offcanvas => {\noffcanvas.addEventListener(\u0027show.bs.offcanvas\u0027, event =>{\n\n    event.preventDefault()\n    }, false)\n  })\n})\n\n': null\n\nconst project = {\n\n  files: {\n\n    'index.html': markup,\n\n    'index.js': jsSnippetContent\n  },\n\n  title: 'Bootstrap Example',\n\n  description: `Official example from ${window.location.href}`, \n\n  template: jsSnippet ? 'javascript' : 'html',\n\n  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/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">
```

```
<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-align: 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>

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

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

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

```
        
```

```
        <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>
```

```
<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 </b></p>
```

```
</form>
```

```
</main>
```

```
<script
```

```
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.bundle.min.js" 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-code-snippet').querySelector('.bd-example').innerHTML
```

```
    const classes = Array.from(event.target.closest('.bd-code-snippet').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>
```



```
tener(() => change(), function () {
  if (!toastPlacement.dataset.originalClass) {
    toastPlacement.dataset.originalClass = toastPlacement.className
    toastPlacement.className = ` ${toastPlacement.dataset.originalClass}
    ${this.value}`
  })
  // Instantiate all toasts in a docs page only
  document.querySelectorAll('.bd-example .toast')
    .forEach((toastNode) => {
      const toast = new bootstrap.Toast(toastNode, {
        autohide: false
      })
      toast.show()
    })
  // Instantiate all toasts in a docs page only
  const toastTrigger = document.getElementById('liveToastBtn')
  const toastLiveExample = document.getElementById('liveToast')
  if (toastTrigger) {
    toastTrigger.addEventListener('click', () => {
      const toast = new bootstrap.Toast(toastLiveExample)
      toast.show()
    })
  }
  // Alerts
  // Used in 'Show live toast' example in docs or StackBlitz
  const alertPlaceholder = document.getElementById('liveAlertPlaceholder')
  const alertTrigger = document.getElementById('liveAlertBtn')
  const appendAlert = (message, type) => {
    const wrapper = document.createElement('div')
    wrapper.innerHTML = [
      `<div class="alert alert-${type}" alert-dismissable`
      `role="alert">`,
      `<div>${message}</div>`,
      `<button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>`,
    ].join('')
    alertPlaceholder.append(wrapper)
    if (alertTrigger) {

```

[illegible]


```
<!-- Offcanvas -->  
<!-- Offcanvas components -->  
example in docs only  
const myOffcanvas =  
document.querySelector(`.bd-example-offcanvas  
.offcanvas`)  
if (myOffcanvas) {  
myOffcanvas.forEach(offcanvas => {  
offcanvas.addEventListener(show.bs.offcanvas, event => {  
event.preventDefault()  
}, false)  
})  
}  
  
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-
1BmE4kWBq78iYhFIdvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ
6jIW3" crossorigin="anonymous">

    <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=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-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">
```

```
<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>
```

```
    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;  
  
    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">

  <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" 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-bs-
dismiss="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>

<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">

    <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>

      <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 text-  
black">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.js" 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-code-
snippet').querySelector('.bd-example').innerHTML

    const classes = Array.from(event.target.closest('.bd-code-
snippet').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>

```



```

new bootstrap.Tooltip((tooltip))\n  })\n\n  \/\ \-|-|-|-|-|-|\n  \/\ Popovers\n  \/\ \-|-|-|-|-|-|\n  \/\ Instantiate all popovers in a docs or StackBlitz page\n  document.querySelector(\u0027[data-bs-\n  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 =\n  document.getElementById(\u0027toastPlacement\u0027)\n  if\n  (\n    toastPlacement\n  ) {\n  document.getElementById(\u0027selectToastPlacement\u0027)\n    .addEventListener(\n      \u0027change\u0027, function () {\n        if\n        (\n          !toastPlacement.dataset.originalClass\n        ) {\n          toastPlacement.dataset.originalClass = toastPlacement.className\n          toastPlacement.className = `${toastPlacement.dataset.originalClass}\n          ${this.value}`\n        }\n      }\n    )\n  }\n  \/\ Instantiate all toasts in a docs page only\n  document.querySelector(\u0027.bd-example .toast\u0027)\n    .forEach((toastNode =\u003e {\n      const toast = new\n      bootstrap.Toast(toastNode, {\n        autohide: false\n      })\n      toast.show()\n    })\n  )\n  \/\ Instantiate all toasts in a docs page only\n  const\n  toastTrigger = document.getElementById(\u0027liveToastBtn\u0027)\n  const toastLiveExample =\n  document.getElementById(\u0027liveToast\u0027)\n  if (\n    toastTrigger\n  ) {\n    toastTrigger.addEventListener(\u0027click\u0027, () =\u003e {\n      const\n      toast = new bootstrap.Toast(toastLiveExample)\n      toast.show()\n    })\n  }\n  \/\ \-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|\n  \/\ Alerts\n  \/\ \-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|-|\n  \/\ Used in \u0027Show\n  live toast\u0027 example in docs or StackBlitz\n  const alertPlaceholder =\n  document.getElementById(\u0027liveAlertPlaceholder\u0027)\n  const

```

```

alertTrigger = document.getElementById('liveAlertBtn')\n\n
const appendAlert = (message, type) => {\n  const wrapper =
document.createElement('div')\n  wrapper.innerHTML = [\n
`<div class="alert alert-${type} alert-dismissible">
role="alert">
<div>${message}</div>
<button
type="button" class="btn-close" data-bs-dismiss="
label="Close"></button>
`.join('\n')\n\n
alertPlaceholder.append(wrapper)\n\n
if (alertTrigger) {\n
alertTrigger.addEventListener('click', () => {\n
appendAlert('Nice, you triggered this alert message!',
'success')\n
})\n\n
// - - - - -
// Checks & Radios\n
// - - - - -
// Indeterminate checkbox example in docs and StackBlitz\n
document.querySelectorAll('.bd-example-indeterminate')
.forEach((checkbox => {\n
if (checkbox.id.includes('Indeterminate')) {\n
checkbox.indeterminate = true\n
})\n\n
// - - - - -
// Links\n
// - - - - -
// Disable empty links in docs examples only\n
document.querySelectorAll('.bd-content')
.forEach((link => {\n
link.addEventListener('click', event => {\n
event.preventDefault()\n
})\n\n
// - - - - -
// Modal\n
// - - - - -
// Modal & Varying modal content example in docs and
StackBlitz\n
const exampleModal =

```



```
template: jsSnippet ? 'javascript' : 'html',  
tags: ['bootstrap']  
}  
  
StackBlitzSDK.openProject(project, { openFile: 'index.html' })  
  
}  
  
</script>  
  
</body>  
  
</html>
```

GitHub

<https://github.com/IBM-EPBL/IBM-Project-47656-1660800898>

Project demo link

Project demo link (youtube) : <https://youtu.be/zbzQLS5v4xY>