



# NewsBee Forum Web App

Nilay Barde, Jiahao Song, Hao Yuan, Chen Gu

# About our app



NewsBee is a web application for individuals to read news along with peer commentators.

- All users have associated passwords and can log in and out
- Users can create topics for news they are interested in
- Users can share their opinions with other users by posting comments
- Users can browse top and breaking headlines for the USA, specific category in a country, single source, or multiple sources
- All users, comments, and topics are stored in a PostgreSQL database.
- All edits to the comments and topics are live and happen in real-time

# How does it work?



- Register button to register new users (supports local password authentication implemented securely)
- Log in using account details (username and password)
- Users' emails and passwords are stored in a Postgres database, along with some other non-trivial persistent state
- Browse news and click on the news title to go to the source page
- News list is retrieved from the News API using HTTPoison (server to server API requests)
- Discuss articles by creating topics and leaving comments
- Uses an external API that requires authentication using an API key. API access is server <-> server.

# What libraries and tools did you use?



- Redux
- React-Router
- Anime.js
- ElixirScript
- HTTPoison
- Poison

# Challenges and Solutions



## 1. Single Page Application

- Redux and React-Router
- Attributions to Professor's homework 7

## 2. News API request creation and rendering

- At first, our browser Javascript code makes requests directly to the remote API, not to the server
- Spent hours to fix this issue and rewrite the whole API-fetch process in elixir code

## 3. The Integration

- Unsure about how to implement SPA from the beginning
- Causes some level of inconsistency while merging our branches

# API, DB, Additional Features



- News API: <https://newsapi.org/>
- Key: Generated from website
- Database: Postgres
- Additional Features:
  1. Use Phoenix Channels to push realtime updates to users,
  2. triggered either by live data from the external API or actions by other users.
  3. Build our app almost entirely as a Single Page Application with Redux and React-Router.