

Group 205 - Daniel, Nick, Noah, Thomas

CS4530 Final Project: *AdOverflow!*

Our Features

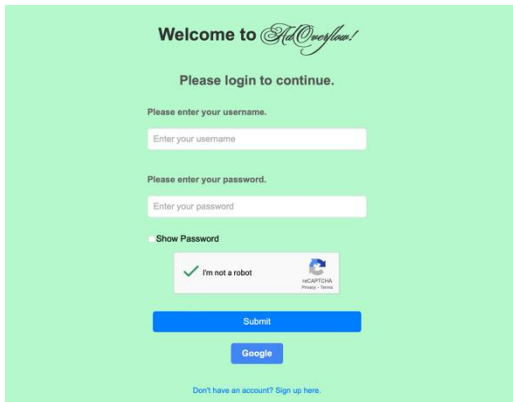
The old FakeOverflow needed an update. There wasn't enough security. There wasn't enough core user functionality. And, most importantly, there wasn't enough revenue!

We addressed these issues with three new user stories: enhanced login (SSO, 2FA, Captcha, Session), remote code execution (execute code, edit files with other users), and the titular ads (popups, targeting, analytics).

Feel free to explore the site and mess around!

Demo and Source

Visit our project at <https://www.adoverflow.ninja>
Check out the code in our repo [here](#).

A screenshot of the login page for 'AdOverflow!'. The page has a light green background. At the top, it says 'Welcome to AdOverflow!' with the site's logo. Below that, it says 'Please login to continue.' There are two input fields: 'Please enter your username.' and 'Please enter your password.' Below the password field is a 'Show Password' link. There is a checkbox for 'I'm not a robot' with a green checkmark and a reCAPTCHA logo. At the bottom, there is a blue 'Submit' button and a smaller blue 'Google' button. At the very bottom, it says 'Don't have an account? Sign up here.'

Technology Stack

Our project uses a Node.js backend running an express web server. The server uses MongoDB for data persistence. The front end is made using React and communicates with the server using Axios and socket.io.

Login features use Passport.js, express-session, NodeMailer, and Google's reCAPTCHA. Code execution is handled using the Judge0 API. Ads are stored Google Cloud

Future Work

With more time to work on this project, we would have liked to flesh out the ad functionality. There is plenty of room to refine our humorous but practical implementation.

We are interested in making the code editor work in line with posts and potentially adding post analytics in a similar vein as our ad analytics.