

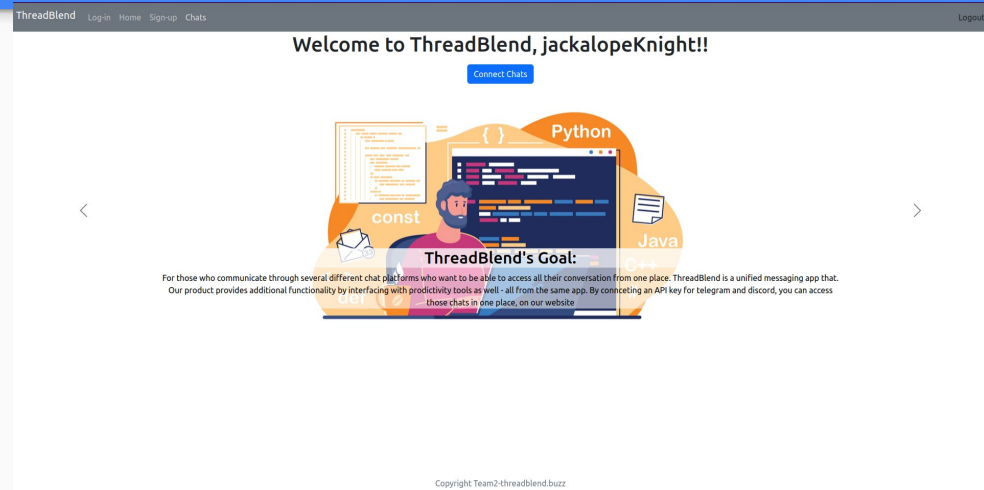
ThreadBlend

Jake McKnight, Aaron Semones, Adam Wuth, Henry Radovich

Project Vision

- Chat App
- Combine multiple applications into one
- Ease of use

Our overall goal was to create an app for those who communicate through several different chat platforms who want to be able to access all their conversation from one place. ThreadBlend is a unified messaging app that. Our MVP was just getting one chat app working.



Tools Used

- HTML/CSS/Bootstrap/JS
 - Used to create and handle processes for our webpages/api calls, $\frac{4}{5}$ stars
- Postgres
 - Used to create a database and tables for users and their data, $\frac{3}{5}$ stars
- Node express
 - Used to help with routes, $\frac{4}{5}$ stars
- Docker/Azure
 - Used to host and deploy our project, $\frac{4}{5}$ stars
- Cucumber
 - Used to create login test, $\frac{3}{5}$ stars

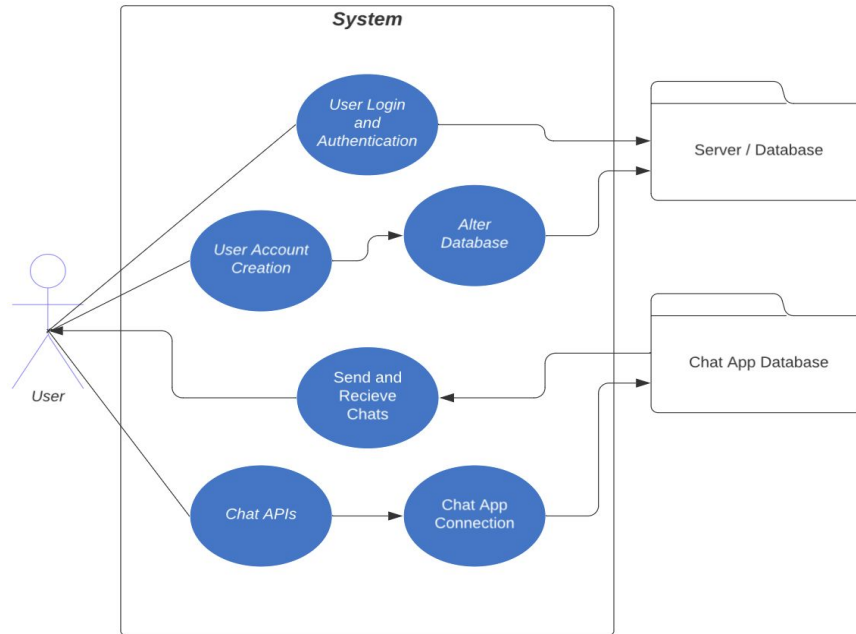
Tools Used Cont.

- GitHub
 - Used to store repository and manage project board, $\frac{3}{5}$ stars merge conflicts are more complicated than necessary
- Discord
 - Used to communicate, $\frac{4}{5}$ stars
- VS code
 - Used as IDE, $\frac{4}{5}$ stars
- Slack/Telegram Api
 - Used to send and receive chats, $\frac{4}{5}$ stars

Methodologies

- Agile
- Project Board
- Pair programming sessions in person

Architecture



Challenges

- Deployment
 - Overcame with help from professor
- Live updating with EJS - tools like React, Angular, and Vue.js
 - Chasing leads that never panned out... Matrix SDK, Socket.io
 - Overcame by being persistent and trying new ways to do things
- Integration - Merging commits, changing the same endpoint at the same time
 - Overcame with time spent resolving conflicts

Demo