# Final sprint 5 release notes:

## Git repo:

<https://github.com/arjen0203/TwitterButForCodeBackEnd>

## Idea document:

<https://github.com/arjen0203/TwitterButForCodeBackEnd/blob/master/documentation/pdf/Idea.pdf>

## Architecture document:

<https://github.com/arjen0203/TwitterButForCodeBackEnd/blob/master/documentation/pdf/Architectuur.pdf>

OWASP report: <https://github.com/arjen0203/TwitterButForCodeBackEnd/blob/master/documentation/pdf/OWASP%20report.pdf>

PDR: <https://github.com/arjen0203/TwitterButForCodeBackEnd/blob/master/documentation/pdf/Personal%20Development%20Report.pdf>

## CI/CD workflow through git actions:

<https://github.com/arjen0203/TwitterButForCodeBackEnd/blob/master/.github/workflows/cicd.yml>

## Trello board where I keep track of my user stories

<https://trello.com/b/MKWlhjNf/twitter-but-for-code>

## Achievements in individual project

Been able to set up the following working services within Kubernetes in docker, as well as been able to deploy this to the cloud using scripts.

**Authorization service:** This service creates and reads JWT cookies. Every request that needs to be authorized goes through this service. If a JWT token is valid it will add the respective user Id and role to the headers request before sending it through to the requested service.

**User service:** This service manages the users information, like his id, name, username and hashed password. If a user wishes to delete their account a message is send with RabbitMQ that tells other services to delete that users information.

**Post service:** This service allows a user to create their own posts. Users can then like, comment or revision these posts. A revision is a new post that links to a different post as a means to show your own implementation of a code snippet. This service also allows the getting of a page of posts of a certain user or the trending posts based on the traffic handled in the trending service.

**Trending service:** This service keeps track of all the traffic that posts have. It does this by receiving messages with RabbitMQ that contain a post id and a timestamp. This service then keeps this traffic in a queue and writes this queue to the database in batches based on time or the amount in the queue. Based on this traffic the service can create sorted pages by a certain time frame containing post ids and their traffic amount. These pages will be cached by certain timeframes increasing its response speed.

**Frontend service:** A react frontend that uses all the current features of the backend.

**CI/CD:** My project gets completely build, tested and scanned using GitHub actions. This is done when I want to merge to the main branch. GitHub will send me a email if the job fails and won’t allow users other than the admin to merge the branch.

**FAAS:** I have also written a function as a service that will filter out profanity in a given string. Request can then be send to the URL given to me by azure.

## Sprint retrospective

I was able to get a lot done in the last sprint. For example I didn’t think I’d have time to deploy my project within the cloud but with a little bit of help from my classmates I was able to successfully deploy my application within azure cloud. I worked on a lot of small things this final sprint to make sure I could get my learning objectives to proficient. I am satisfied with the current results of my individual project and am confident in my understanding and knowledge of the learning outcomes.