The Social Cookbook

David Rutqvist

Erik Halén

Technical Choices

- Node.JS – Back-end
- Express – Web Application Framework
- Vash – View Engine (.NET MVC Razor like)
- LESS – For “intelligent” CSS
- Semantic UI – Frontend framework
- JQuery – Frontend JavaScript Framework
- JSON Web Tokens – Used for authentication

Direction of Project

- More focused on big scale architecture, scalability etc. than actual programming
- Goal is to create a recipe sharing service where users get inspired from each others cooking

Server Architecture

DigitalOcean

Internet

DMZ

CentOS

DNS BIND

CentOS

API dev
Frontend dev
API prod
Frontend prod
Database

NGINX

PRIVATE

Automation

1. Developer pushes changes

2. GitHub Webhook

3. Jenkins pulls repo

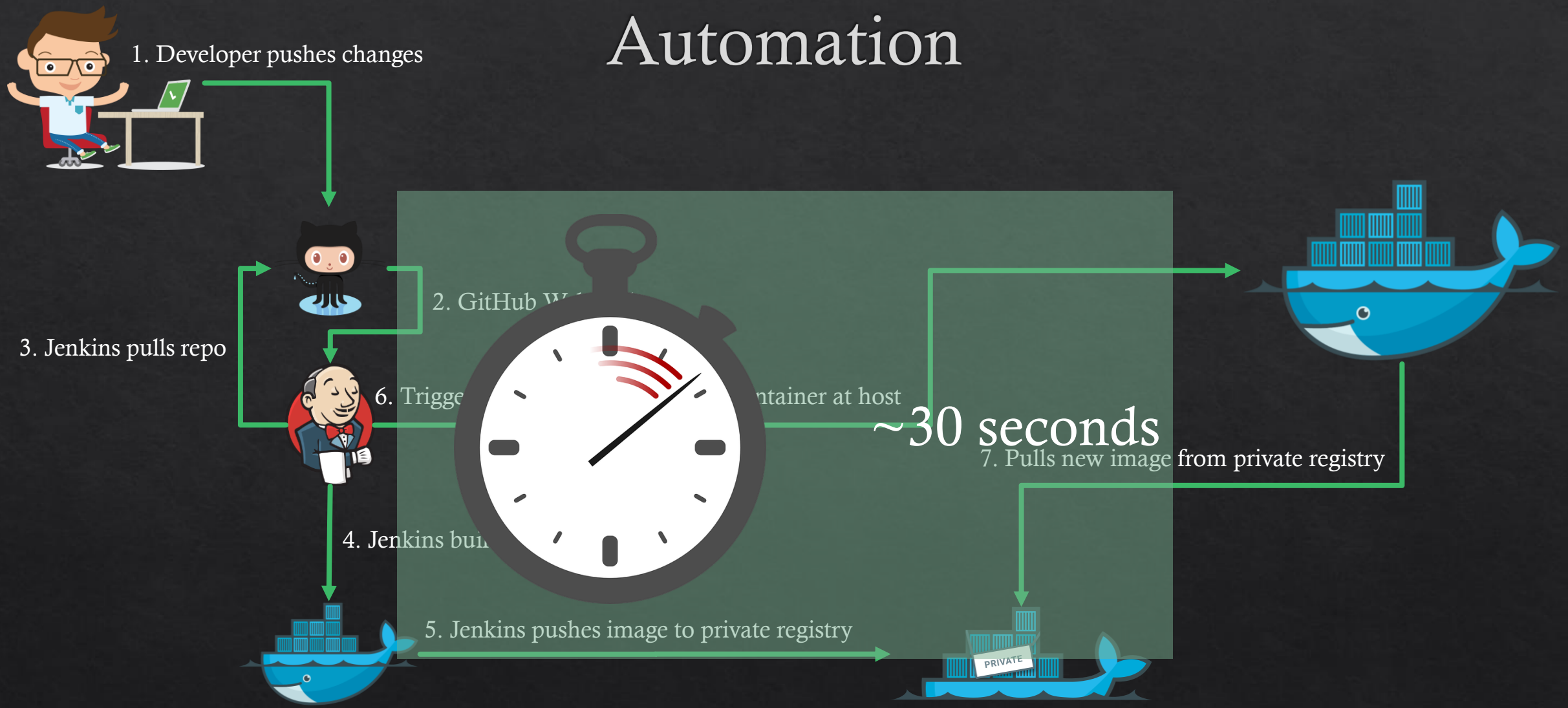
6. Triggers update of development container at host

7. Pulls new image from private registry

4. Jenkins builds Docker image

5. Jenkins pushes image to private registry

Automation


1. Developer pushes changes
2. GitHub W
3. Jenkins pulls repo
6. Trigge
ntainer at host
~30 seconds
7. Pulls new image from private registry
4. Jenkins bui
5. Jenkins pushes image to private registry
PRIVATE

System Architecture

Frontend (what the user sees)	Backend	API	Database
User makes request	Incoming request is mapped to a REST request	Actual request is handled	CRUD of Data
User sees result	REST response is mapped to a view and sent to the browser	REST response	

Current Status

- A lot of time has been used to set up a proper development, testing and production environment so the actual development is in the early stages
- The frontend is currently just a default Semantic UI template, will be done last
- API supports authentication using JWT
- Database design to be done in the following days

Questions?