



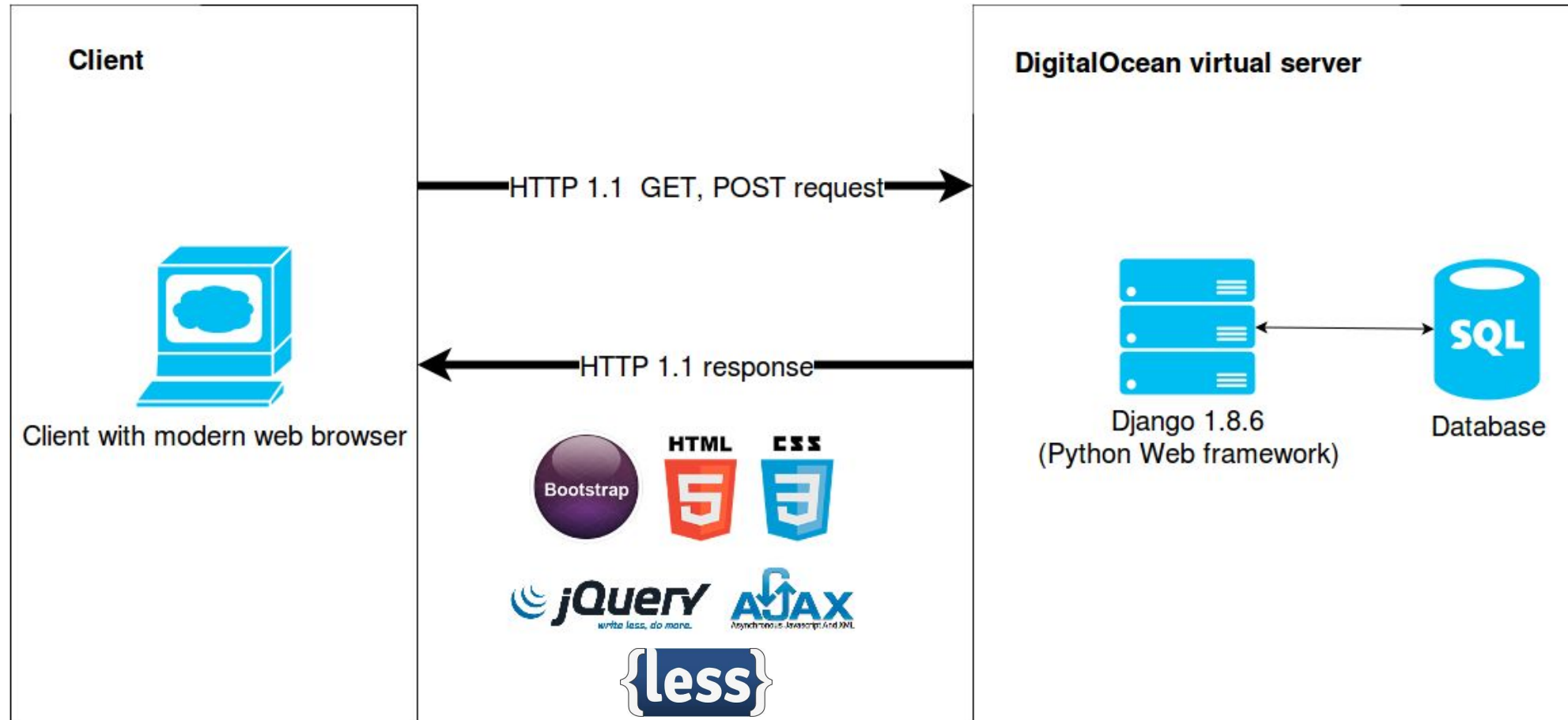
Christoph Malin, Thibault Rapin

# Idea

- Platform for managing events e.g. catering company
- Company has a lot of freelancers
- Company can
  - create events
  - manage users which work for an event
  - manage users (groups, user data)
- Employees can
  - register themselves
  - apply for events
  - change their data
  - see if they are accepted to work for an event



# Architecture & Technologies



# Employee functionalities

- Register with personal data + picture upload
- Change personal data, password and profile picture
- See all available new events
- Apply for an event
- See the status of applied events



# Administrator functionalities

- Manage events (add, change)
- Manage users working for an event
- Manage users
  - activate new users
  - deactivate users
  - change user accounts (user data, password, profile picture)
- Manage user groups (add, change)



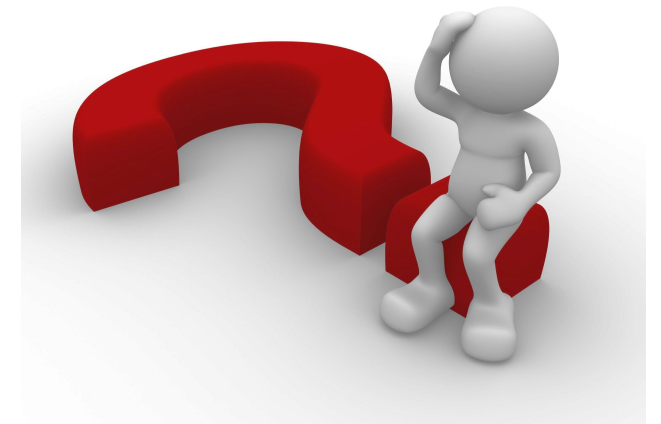
# Security

- SQL injection
  - Unlikely cause of the usage of the django database-abstraction API
- Password Storage
  - Password algorithm recommended by NIST with a SHA256 hash
- Session Management (part of the framework)
  - Change password logs out all sessions of specific user
- Cross site request forgery (CSRF)
  - CSRF token needed for every POST-REQUEST



# Problems encountered

- Achieve agreement about functionality and design
- Understanding the basic workflow of Django
- Uploading pictures
  - How to handle picture upload in user registration part?
  - Displaying pictures with the DropzoneJS library
- Deployment to web



# Further Plans

- Increase security (OWASP Top 10, TLS)
- Increase usability (more AJAX)
- Add test cases with high code coverage
- Add more functionality
  - Search for users
  - Mail notification service
  - SMS notification service
- Create an android application for the employees
- Check compatability with different web browsers



# Demo

[www.bluehive.me](http://www.bluehive.me)