

# Rapid development with Google AppEngine

Aleš Justin, OpenBlend

# What is GAE?

- PaaS by Google
  - Python, Java, Go, PHP
  - Custom API (with quotas / limits)
  - New managed envs, remote API integration, ...
  - Excellent console
  - Simple deployment
  - ...

# GAE alternatives?

- When it's all about the API
  - Java → CapEDwarf
  - Appscale

# Simple app?

- Frontend / UI
  - JavaScript
  - Servlets → Endpoints
- Business Logic
  - Error handling
  - Authentication / Authorisation
  - Validation
  - Transactions
- "Database"
  - Datastore
- Clients
  - Browser
  - Mobile
  - CLI

# Frontend / UI

- JavaScript
  - jQuery, bootstrap, backbone, ...
- Servlets
  - Endpoints
    - REST-like
    - Auto-magical js client generation
    - Issues with Users API ...

# Business Logic

- Common module – server and clients shared
- Separation of concern aka AOP
  - Aspects
    - Error handling
    - Authentication / Authorisation
      - GAE Users API vs. OAuth API?
    - Validation
    - Transactions
    - ...

# Business Logic

- Custom AOP
  - Could / should be Guice
- Simple chain of aspects / interceptors

```
private final Administration administration = ProxyFactory.proxy(
    Administration.class,
    new AdministrationImpl(),
    Arrays.asList(
        Aspects.ERROR,
        Aspects.ERASE,
        Aspects.ADMIN,
        Aspects.TX,
        Aspects.validators(Validators.RESERVATION, Validators.PLAYER,
            Validators.COMPETITION, Validators.CP, Validators.MATCH, Validators.OUTAGE)
    )
);
```

# Business Logic

- Database Lookup
  - Cached → as an aspect!
    - GAE Memcache API
- Audit log / notifications
  - Async
  - Transactional
  - GAE TaskQueue API
    - DeferredTasks
  - GAE Mail API (sender setup, ...)

# Database

- ◉ As an interface
  - ◉ Can be changed to other impl
- ◉ Currently DatastoreService based
  - ◉ Could / should be using Objectify
  - ◉ Be aware of tx and "entity groups"!
- ◉ Cache simple data
  - ◉ It's cheap

# Clients

- Browser + JavaScript
  - Endpoints generated client
  - jQuery "REST" client
- Mobile, CLI, ...
  - Simple ServerProxy interface

# Testing?

- Simply test all ServerProxy methods
  - Dev server or CapeDwarf
  - Appspot + "test" app id
- Mock Login with WebDriver
  - Graphene, Selenium, PhantomJS, ...

# Console

- Evolved A LOT in last few years
- Most used things (in my case)
  - Datastore view
  - Logging view
  - AppEngine view
    - Memcache (flush)
  - Permissions (IAM)
- Debugging
  - Upload your code to their Git repo → debug!

# Examples

- IRC bot
  - IRC ↔ GAE ↔ Android
- Tennis reservations
  - Started small & quick
  - Now a semi-serious project :-)

Q&A

- ales.justin@gmail.com
- @alesj