Play! Framework

Bylting í Java vefþróun

Ólafur Gauti Guðmundsson

Hvað er **Play**?

- “**Full-stack**” vef “framework”
- Stofnað af Guillaume Bort árið 2008
- Núverandi útgáfa: **1.2.4**
- Ca. 380 þús niðurhöl
- Ca. 4300 meðlimir á póstlista
- Yfir 100 módúlar

Play er öðruvísi!

- **EKKI** byggt á Servlet API
- Stateless, **engin session**
- Full-stack, all included

Af hverju var **Play** smíðað?

- Java Enterprise stack er of **flókinn** og **tímafrekur**

Spring Framework
Guice
JBoss Seam
Velocity
<FreeMarker>
RESTlet
webwork Struts
Struts 2
LOG4J
JUnit.org
maven
Apache Tomcat
RichFaces
JavaServer Faces Technology
jetty://
SLF4J Simple Logging Facade for Java
<APACHE ANT>
HIBERNATE
Lucene

JavaEE: Ekkert “out-of-the-box”

- Þurfum að finna library fyrir **grunnvirkni**
  - MVC, DB access, caching, logging, email, json, file upload, etc.
  - Unit testing
- Samþætta og viðhalda aragrúa af JAR skrám
- Áður en við byrjum að leysa “business” vandann!

JavaEE: Tímafrek þróun

- Ant / Maven
- Compile, deploy, restart, etc.
- “Dependency” stillt í XML súpu
- Villuskilaboð oft margar skjáfyllir
- Of tímafrekt, of mikið vesen!

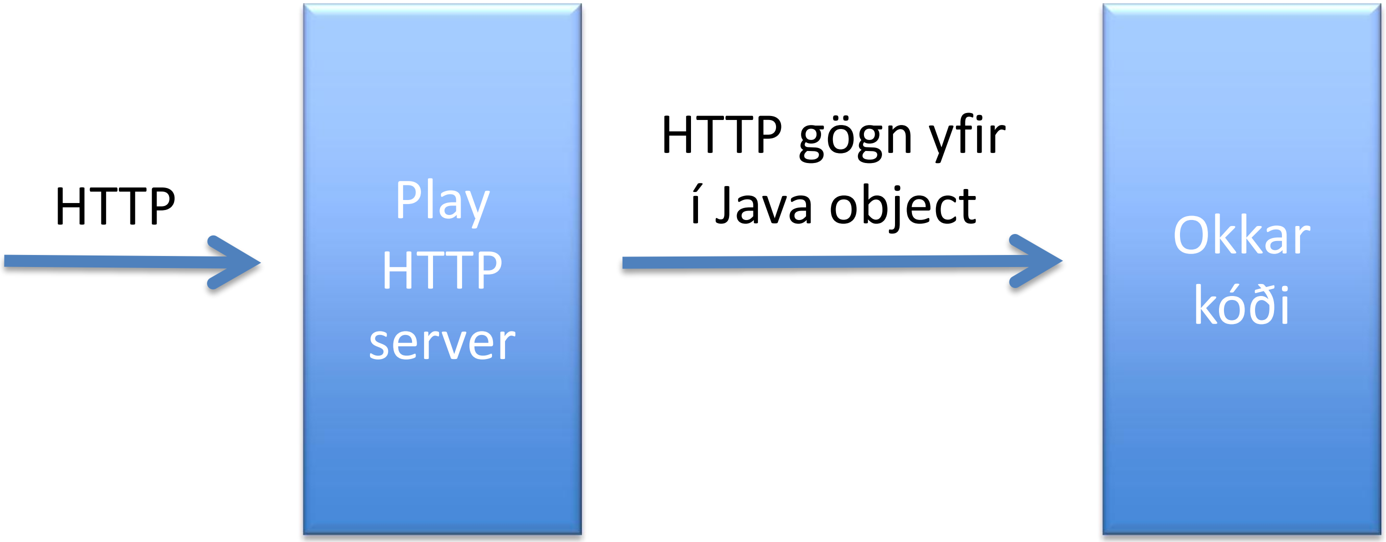
JavaEE: Sér ekki skóginn fyrir trjánum

- Við “of-hönnum” einföld vefkerfi
- Of mikill tími fer í pípulagnir
- Tíminn ætti að fara beint í að leysa “**business**” þarfirnar

Play it simple

- “Full-stack”
  - Öll grunnvirkni innifalin
- Þægilegt þróunarferli
  - Incremental compiler
  - Breyta í editor, hressa í browser
- Einfalt dependency management
- Góð villuskilaboð
- Frábær skjölun

Play byggir **ekki** á Servlet API



- Hraðvirkur HTTP server byggður á Netty

URL fyrir fullkomnunarsinna

http://myPlayApp.com/items/**123**

- URL routing er stillt í conf/routes:

```
GET  /items/{id}    Items.display
```

Controller: static methods

```
package controllers;
import models.Item;
import play.mvc.Controller;

public class Items extends Controller {

   public static void display(Long id) {
      Item item = Item.findById(id);
      notFoundIfNull(item);
      render(item);
   }
}
```

Sniðmát: Groovy

Í views/Items/display.html:

```
#{extends 'main.html' /}
#{set title: item.title /}
<h1>${item.title}</h1>
<p>${item.body}</p>
```

Fullt af innbyggðum tögum og viðbótum.

Einfalt að skrifa út mism. **content type**

```
…
public static void itemAsJson(Long id) {
   Item item = Item.findById(id);
   renderJSON(item);
}
…
public static void itemImage(Long id) {
   Item item = Item.findById(id);
   renderBinary(item.image.getFile());
}
…
```

ORM: **JPA** innbyggt

```
package models;
import javax.persistence.*;
import play.db.jpa.*;

@Entity
public class Item extends Model {

   public String title;
   public String body;

   public Item(String title, String body) {
      …
   }
}
```

Save og find erfast frá Model

```
Item item = new Item(“hello”, “world”);
item.save();

Item fromDb = Item.findById(item.id);
fromDb.delete();

List<Item> items = Item.find(“byTitle”,”hello”)
   .from(10).fetch(10);
```

Gagnagrunnur stilltur í application.conf

- Í þróun á localhost er þægilegt að nota:

  db=**mem** (transient in-memory db)

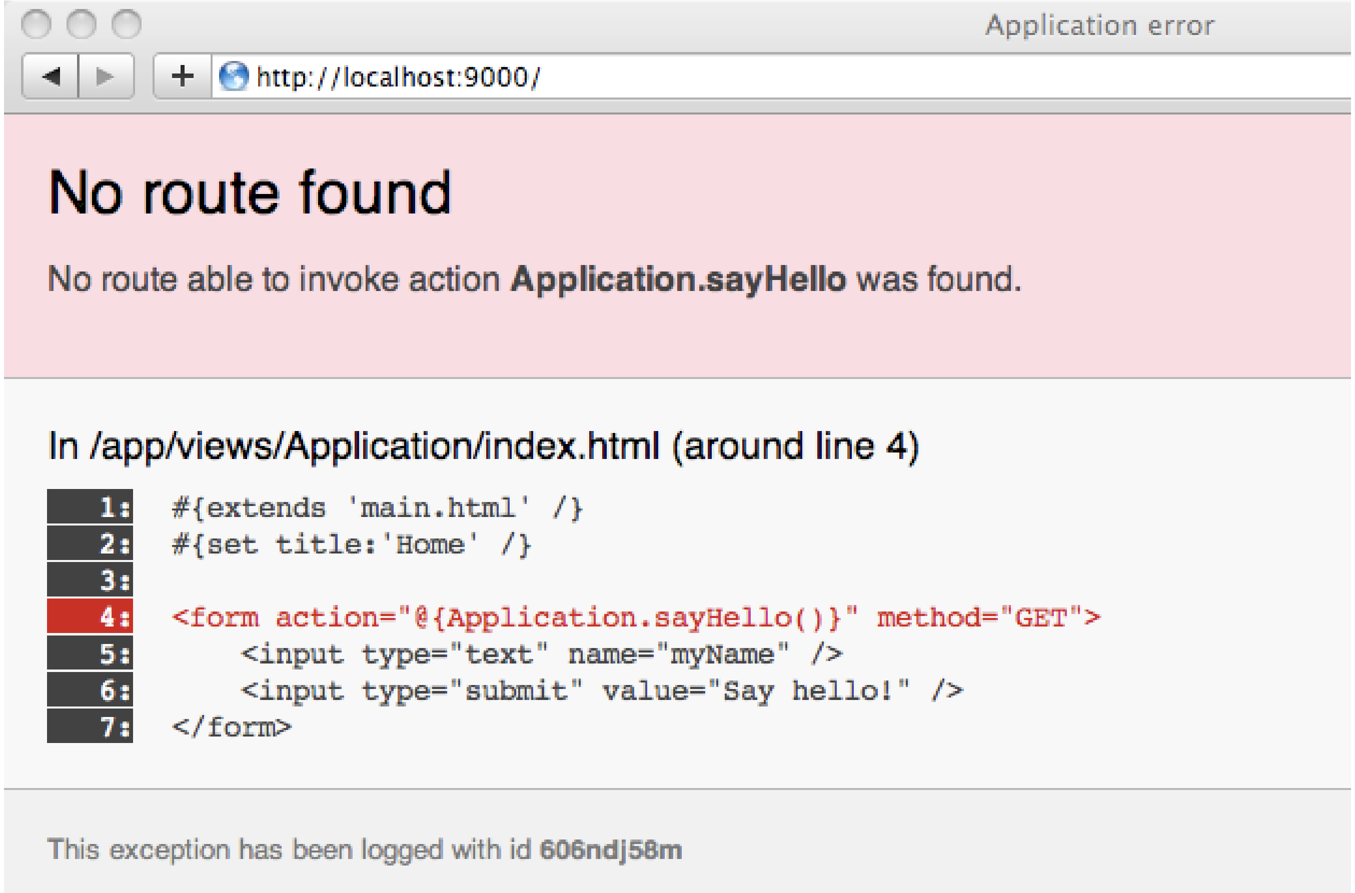
  db=**fs** (simple file db)

- Mism. stillingar fyrir dev / test / prod

  %test.db=mem

  %prod.db=mysql:user:pwd@database_name

Góð **villuskilaboð**



No route found

No route able to invoke action **Application.sayHello** was found.

In /app/views/Application/index.html (around line 4)

```
1: #{extends 'main.html' /}
2: #{set title:'Home' /}
3:
4: <form action="@{Application.sayHello()}" method="GET">
5:     <input type="text" name="myName" />
6:     <input type="submit" value="Say hello!" />
7: </form>
```

This exception has been logged with id **606ndj58m**

Play!

Þýðingar

- application.langs = is,en
- messages.is

  welcome = Velkomin(n), %s
- messages.en

  welcome = Welcome, %s
- Í sniðmáti:

  <h2>&{‘welcome’, user.fullName}</h2>

Asynchronous **Jobs**

```
@Every("1h")
public class CleanupJob extends Job {

   public void doJob() {
      …
   }
}
```

Einnig hægt að skilgreina:

```
   @OnApplicationStart
   @On("0 0 12 * * ?")
```

Einfaldari **dependency**

- Stillt í dependencies.yml:

```
require:
  - play 1.2.4
  - play -> secure
  - play -> fbgraph 0.3
  - org.elasticsearch -> elasticsearch 0.18.2
```

Innifalið: **Test** framework

- Vefviðmót til að keyra þrenns konar test:
  - Unit test
  - Functional test (beinar prófanir á controllerum)
  - Selenium test (acceptance test á vefviðmóti)

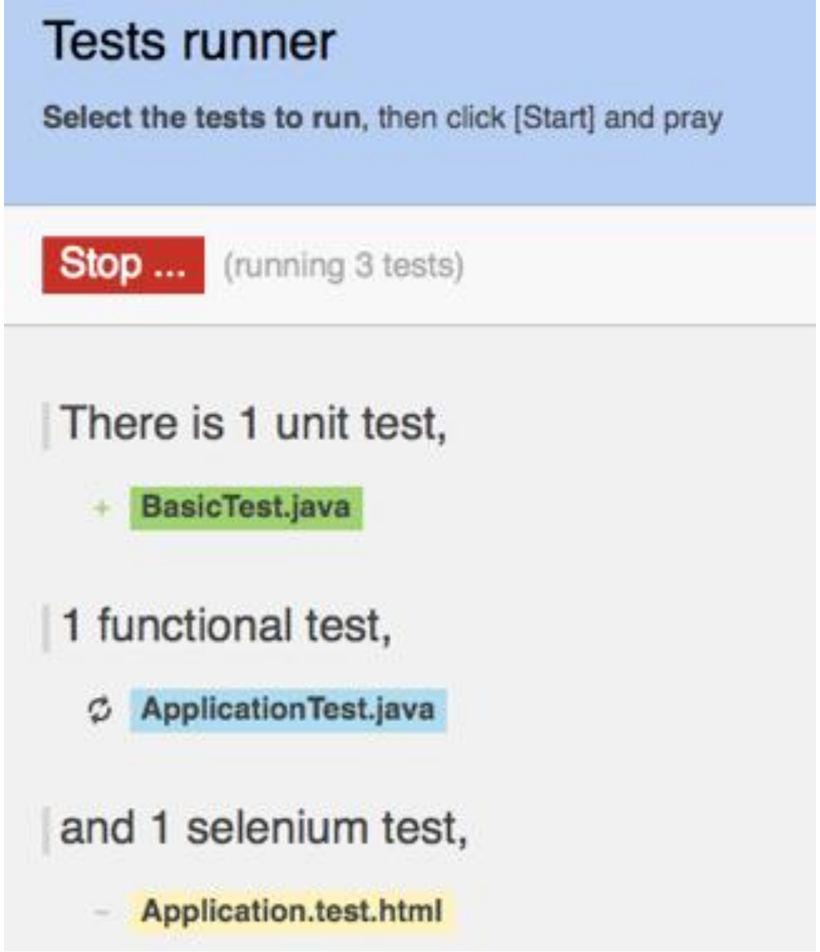
- Auðvelt að importa gögnum fyrir test:

```
@Before
public void setUp() {
   Fixtures.deleteAll();
   Fixtures.loadModels("test-data.yml");
}
```

Test viðmót

```
# play test myApp
```

http://localhost:9000/@tests



Tests runner

Select the tests to run, then click [Start] and pray

(running 3 tests)

There is 1 unit test,

- BasicTest.java

1 functional test,

- ApplicationTest.java

and 1 selenium test,

- Application.test.html

Margt fleira **innbyggt**

- Validation
- Security
- Caching
- Database evolutions
- Logging
- Sending email
- Web service client
- Websocket support
- Asynchronous request handling
- O.fl.

IDE Friendly

- IntelliJ IDEA:

```
# play idealize myApp
```

- Eclipse:

```
# play eclipsify myApp
```

- Netbeans:

```
# play netbeansify myApp
```

Deployment

- Hægt að keyra **standalone** með Play:

```
# play start myApp
```

- Líka hægt að búa til **WAR** skrá fyrir app server:

```
# play war myapp -o myapp.war
```

- …eða deploy-a í “skýið”

Play í **Skýinu**

- Margir cloud provider-ar styðja Play
  - Ýmist beint, eða í gegnum module
- Playapps.net

```
# play playapps:deploy myApp
```

- Google App Engine

```
# play gae:deploy --gae=/path/to/appengine-sdk-1.6.0
```

- Heroku

Modules

- Scala module
- PDF module
- Excel module
- MongoDB
- ElasticSearch
- CoffeeScript
- LESS
- Facebook
- O.fl.

Play 2.0

- Play komið inn í Typesafe
- Native Scala support
  - Hægt að hafa bæði Java og Scala skrár í sama appi
- AKKA actor kerfi
  - Frábært fyrir concurrent fídusa
- LESS, CoffeeScript, Google Closure compiler

Leikið ykkur með **Play!**

www.**playframework**.org

Takk fyrir mig!