Front End Full Stack Evolution

Native Applications
Snappy, interactive UI

Web Pages
Ease of distribution

slides adapted from this talk by LUCA MARCHESINI at FullStackFest 2016:
https://www.youtube.com/watch?v=wtURpqTgtUs&list=PLe9psSNJBf76DOOKMkDpyo_A5PfZk7JWc

1990

GUI

1990

UI Interactive Dynamic

View Logic

Business Logic

Database

Native GUI
Desktop
Applications

1990

1992

1992

UI Static

View Logic

Business Logic

Database

First Simple
Web Apps

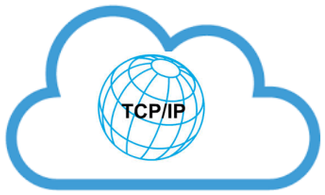
1985

1992

1995

1995

UI Interactive



View Logic

Business Logic

Database

html
css
js

Web Apps with
JS Enabled
Interactive
Features

1985

1992

HTML

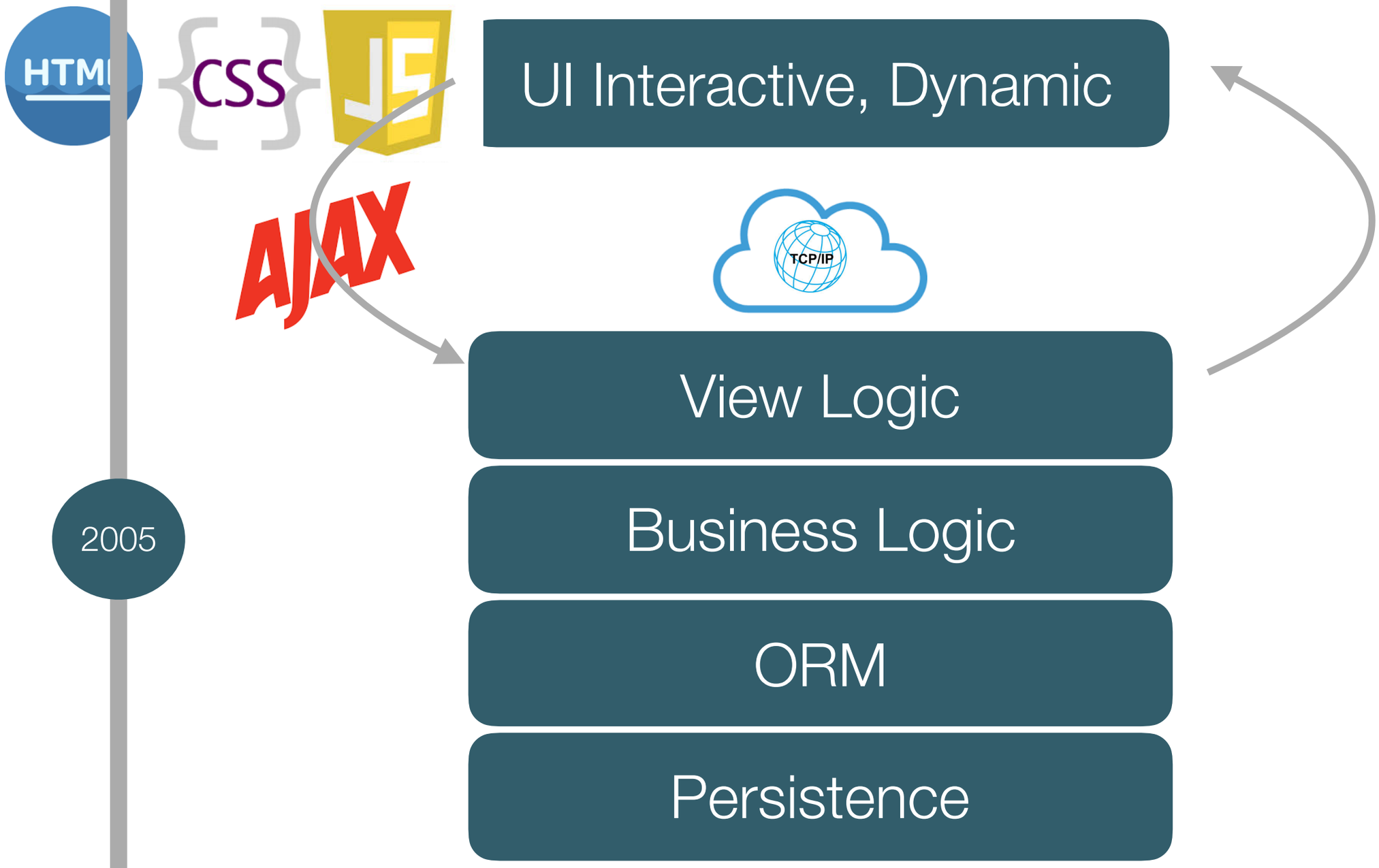
CSS

TCP/IP

1995

2005

AJAX



Ajax Enabled Web Apps - Matching Simple Desktop Apps

1985

1992

HTML

CSS

TCP/IP

1995

2005

AJAX

2010

Angular 1

ember

Backbone

{JSON}

2010

HTML CSS JS

UI Interactive, Dynamic

View Logic

Angular 1 ember Backbone

AJAX

{JSON}

REST API

Business Logic

ORM

Persistence

Web Apps
fully match
sophisticated
Desktop Apps

1985

1992

HTML

CSS

TCP/IP

1995

2005

AJAX

2010

Angular 1

ember

Backbone

{JSON}

2013

React

Angular 2

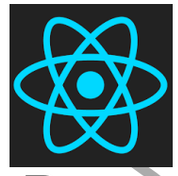
Vue

Aurelia

{fetch API}

UI Interactive, Dynamic

Components



React Angular 2 Vue Aurelia

{fetch API}

{JSON}

REST API

Business Logic

ORM

Persistence

2013

Web Apps based on modular architecture - become extensible and reusable

1985

1992

HTML

CSS

TCP/IP

1995

2005

AJAX

2010

Angular 1

Backbone

{JSON}

2013

React

Angular 2

Vue

Aurelia

{fetch API}

2015

GraphQL

Redux

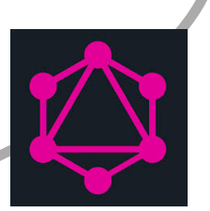
UI Interactive, Dynamic

Components

The Store

{fetch API}

{JSON}



GraphQL

Single Point API

Business Logic

ORM

Persistence

2015

Apps include sophisticated data model - complex analytics and data driven UI

Donation Versions

https://bitbucket.org/edeleastar/donation-web-final

https://bitbucket.org/edeleastar/donation-client

donation-web

- Labs 5-11
- Tags:
  - lab.XX.start
  - lab.XX.end
  - lab.XX.exercises

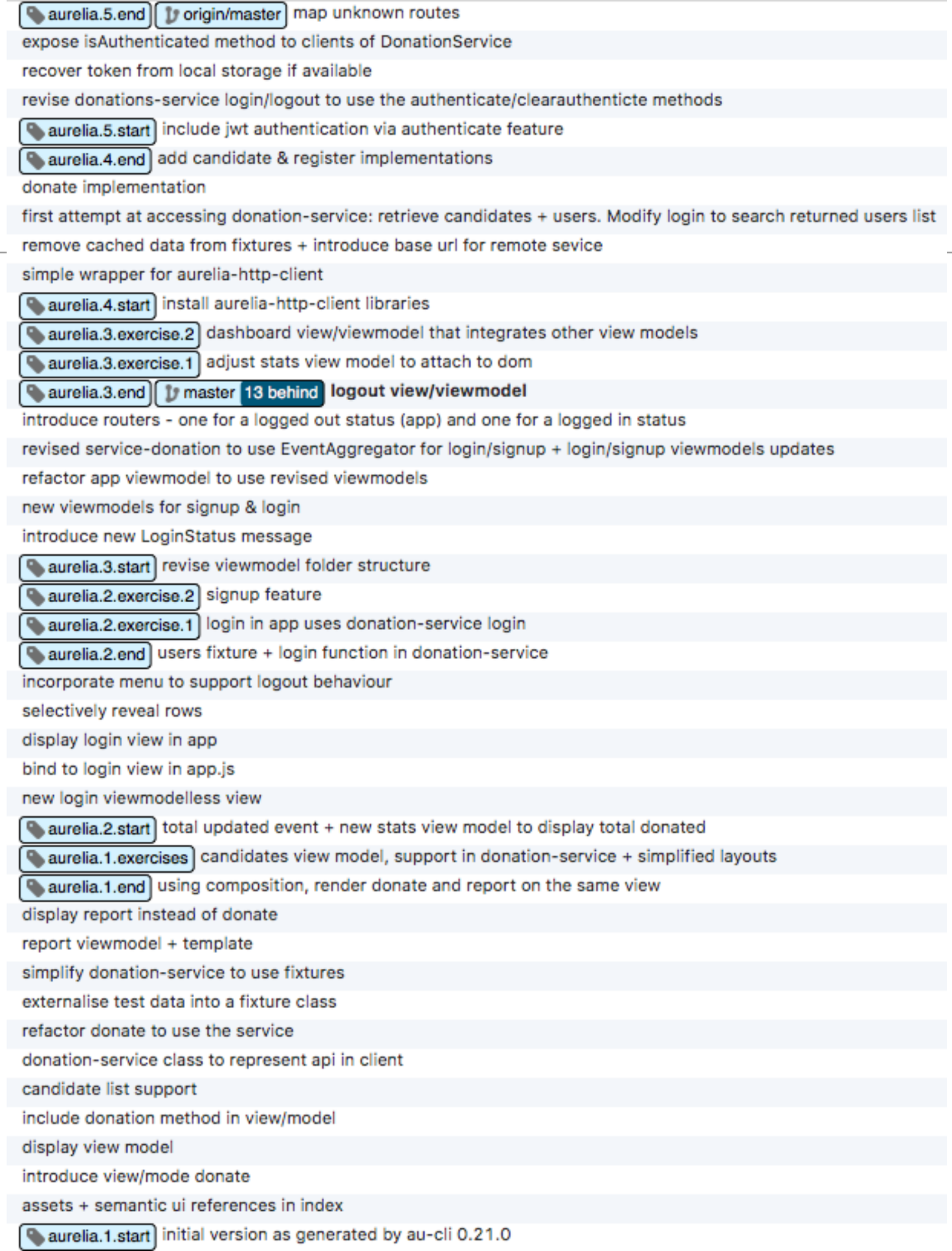
lab.11.exercises totals in report, variable donation amounts + validation in donate view
lab.11.end render candidates in report view
create candidate reference in donation object
introduce and seed candidate model
more extensive set of initial objects
lab.11.start seed single user + donation
lab.10.end production mode db uri
lab.10.start start script for app deployment
lab.9.exercises include validation in login and settings views
lab.9.end turn off abort early to enable multiple errors
engage validation in register from
form error partial
install joi plugin
populate donor object in donation query
lab.9.start incorporate donor object reference into donation
lab.8.exercises fix settings to use database stored user model
lab.8.end display donor email in donations
introduce donation model
introduce user model
lab.8.start connect to mongo via mongoose
lab.7.exercises implementation of settings view
lab.7.end dry - active menu item support
redirect protected routes to login page
cookie retrieval
cookie setting and clearing
default strategy + open accounts routes
lab.7.start cookie plugin initialisation
lab.6.exercises track users and associate with donations
lab.6.end very simple donation persistence
accounts and donations controllers + routes
report and home views + partials
vision layout + handlebar partials
first handlebar expressions
lab.6.start introduce vision hapi plugin + handlebars engine
lab.5.exercises additional routes + static views for login and signup
lab.5.end introduce and serve static assets
revised project structure + first simple biew (using inert)
java code standards. airbnb
first route + controller
first simple hapi server
lab.5.start initial empty project

donation-web

- Labs 12-15
- Tags:
  - lab.XX.start
  - lab.XX.end
  - lab.XX.exercises

donation-client

- Labs - aurelia 1-5
- Tags:
  - aurelia.X.start
  - aurelia.X.end
  - aurelia.X.exercises



UI Static

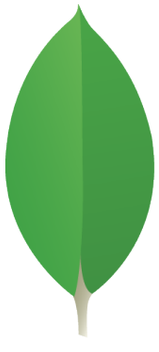
handlebars

View Logic

Business Logic

mongoose

ORM



MongoDB

Persistence

donation-web - to lab 11

UI Dynamic

Components




View Logic

REST API

Business Logic

ORM

Database

Final donation-web + donation-client

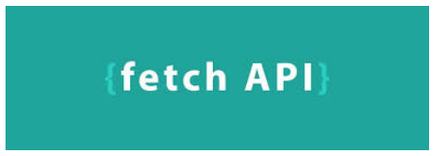
UI Interactive, Dynamic

Components

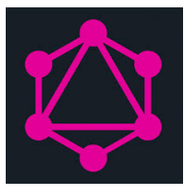
The Store

Aurelia

Redux



Single Point API

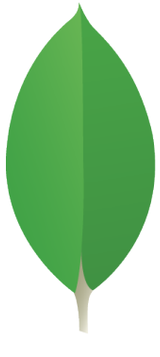


GraphQL

Business Logic

mongoose

ORM



MongoDB

Persistence

Future donation-web + donation-client

donation-web package.json

```
{
  "name": "donation-web",
  "version": "1.0.0",
  "description": "an application to host donations for candidates",
  "main": "index.js",
  "scripts": {
    "start": "node index",
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "boom": "^3.2.2",
    "handlebars": "^4.0.5",
    "hapi": "^14.1.0",
    "hapi-auth-cookie": "^6.1.1",
    "hapi-auth-jwt2": "^7.0.1",
    "hapi-cors-headers": "^1.0.0",
    "inert": "^4.0.1",
    "joi": "^9.0.4",
    "jsonwebtoken": "^7.1.9",
    "lodash": "^4.15.0",
    "mongoose": "^4.5.8",
    "mongoose-seeder": "^1.2.1",
    "vision": "^4.1.0"
  },
  "devDependencies": {
    "chai": "^3.5.0",
    "mocha": "^3.0.2",
    "sync-request": "^3.0.1"
  }
}
```

packages

“hapi"

“handlebars"
"inert"
“vision"
“joi"
'hapi-auth-cookie"

View Logic

REST API

“boom"
"hapi-auth-jwt2"
“hapi-cors-headers"
“jsonwebtoken"

Business Logic

ORM

"mongoose"
"mongoose-seeder"

Database

"lodash"
"chai"
"mocha"
"sync-request"