

National Technical University of Athens School of Electrical & Computer Engineering

Eir Web Application

Appathon@NTUA 2020: Applications for people

Internet Applications

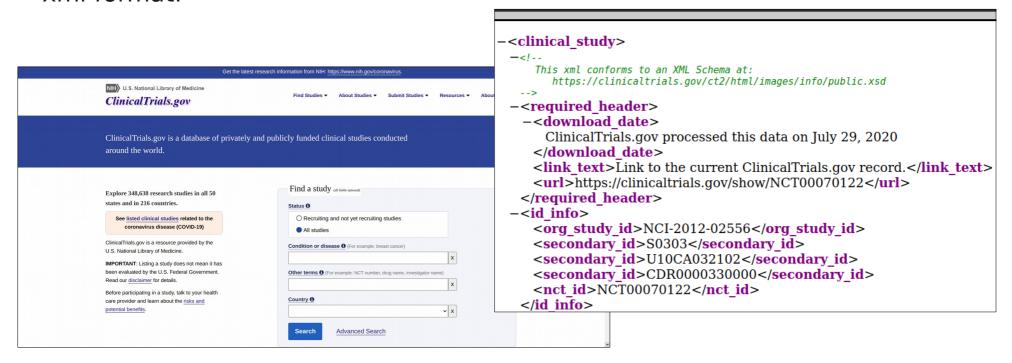
Eir Web Application

Contents of these slides:

- Dataset & Initiatives
- Application Structure & Technologies used
- Brief description of code & its components
- Working Examples

Dataset

Clinical Studies of this application are available from https://clinicaltrials.gov/ in xml format.



• 347,165 Studies were imported and used for this Web app, that was published on July 29, 2020

Dataset

- <u>Goal</u>: Use of these data for extracting further information, not obvious at first sight
- Interesting Attributes for further investigation and simple processing:
- 1. enrollment type: 'Actual'. [Relevant to the actual number of volunteers participating in each clinical study]
- 2. enrollment type: 'Anticipated'. [Anticipated number of participants/volunteers in each study]
- 3. study_first_submitted & last_update_submitted [Dates indicating when each study first submitted and when last updated]
- Last two attributes could be used for computing the average time for requitment by a simple subtrackion.

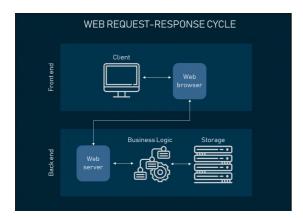
Application Structure

<u>Front-end Server handling:</u>

- User Interface and client requests by a simple browser
- Presentation of the extracted data and information in user friendly environment

Back-end Rest Service:

- Processing data and basics queries about them
- No concern about their presentation to the end-user

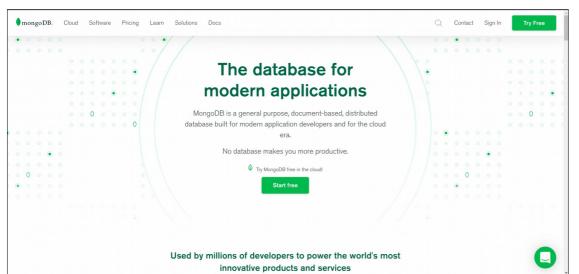


Source of image (Last seen August 12, 2020): https://www.altexsoft.com/blog/front-end-development-technologies-concepts/

No Relational database (and its relevant server) storing downloaded documents in a more consistent way

Application Structure: Database

No-SQL database was created using mongodb as a DBMS



```
> use eir
switched to db eir
> db.clinical_studies.count()
347165
>
```

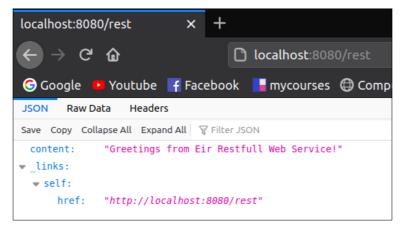
• XML formated data converted to JSON files using xmljson python Library and Gdata convention (attributes are saved under the field name type and actual field values are stored under the field \$t - we later changed it, as mongo doesn't support fields with '\$' in them)

Application Structure: Back-end Rest Service

- Written in Java 11 with Spring Boot Framework and Maven as a built automation tool
- Connection with database established with Spring Data MongoDB module

Service is described as RESTfull (or better as Roy Fielding described R.P.C. Remote Procedure Call), due to the fact that Spring HATEOAS library is
included. It is ensure for each rest reply that includes links to it self and other

components.







Application Structure: Front-end Server

Web site providing access to the aforamentioned service is constructed using:

- Angular 9 Framework & Typescript
- HTML 5 & CSS 3
- Npm was used as the package manager for this application
- Angular Material was used for theme purposes







Brief description of code: Back-end

It consists of an entity package, a repository package and a rest package:

- <u>Entity package</u> contains classes concerning data and documents representation of the NO-SQL database (ClinicalStudies, ActualNumberOfVolunteers, AnticipatedNumberOfVolunteers e.t.c.)
- <u>Repository package</u> contains the MyMongoCollectionRepository class, which implements mongo queries to db and the <u>exception package</u> for basic error handling.
- <u>Rest package</u> consists only of the EirRestController which is responsible for handling GET Requests, listening at http://localhost:8080/
- Parameters of each request are url encoded by each client and decoded by service itself. If no proper parameter was parsed, an exception is raised returning an HTTP Response with status code 400.

Brief description of code: Front-end

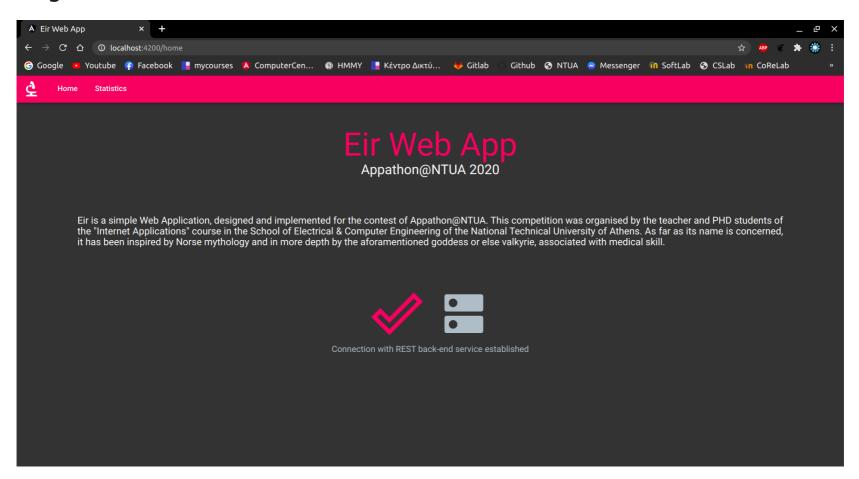
A component oriented solution was chosen:

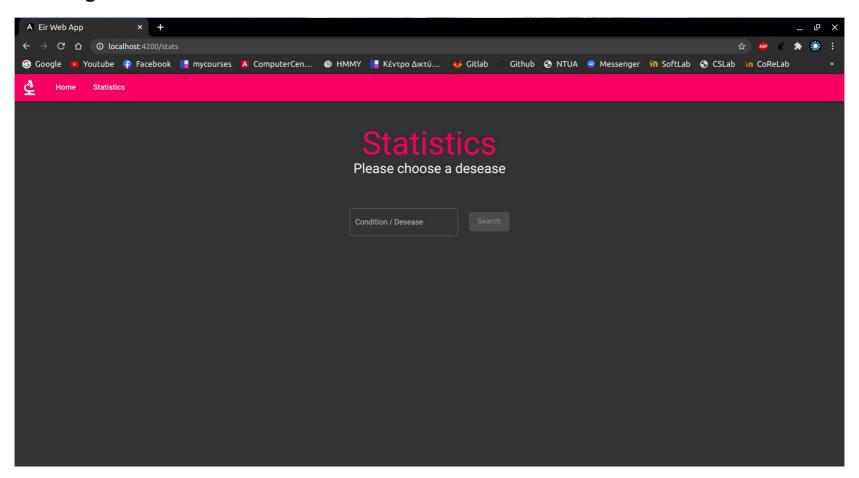
- <u>Dataservice</u> service-component is responsible for consuming back-end REST service and offering a simple interface for all the other components for each other component.
- Home component rendering home page
- Stats component offering a basic page for all the known utilities of Web App
- Each component includes a .css and an html file for its appearance and a .ts file for its functionality
- All calls to back-end service are done asynchronously, offering a more responsive, non-blockin ui

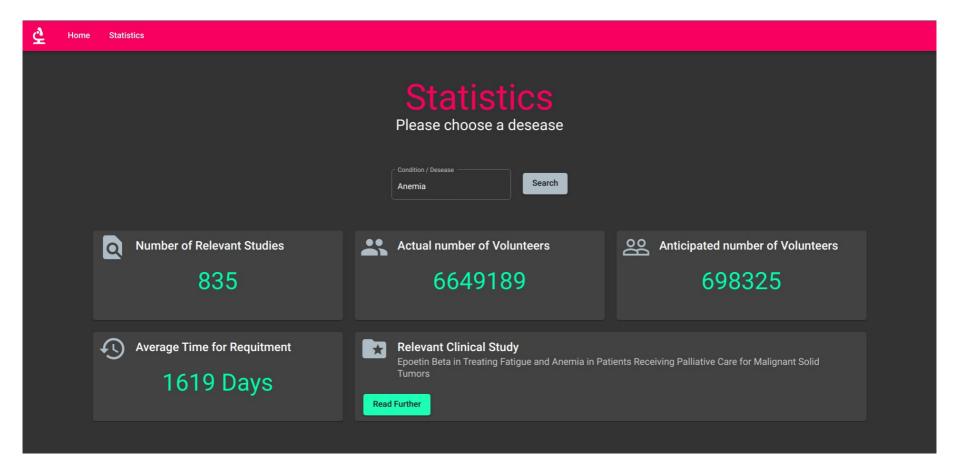
Running all servers in the same machine under a linux distro:

```
ng serve
                                                                                                     :front-end$ ng serve
                                                                                           chunk {main} main.js, main.js.map (main) 58 kB [initial] [rendered]
  ./mvnw spring-boot:run
                                                                                           chunk {polyfills} polyfills.js, polyfills.js.map (polyfills) 141 kB [initial] [rendered]
                                                                                          chunk {runtime} runtime.js, runtime.js.map (runtime) 6.15 kB [entry] [rendered]
chunk {styles} styles.js, styles.js.map (styles) 330 kB [initial] [rendered]
chunk {vendor} vendor.js, vendor.js.map (vendor) 4.42 MB [initial] [rendered]
                                                                                                                                                              - Time: 21084ms
                                                                                           ** Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ **
                                                                                          : Compiled successfully
                                          main] g.n.e.internetappli.eir.EirApplication : Starting EirApplication on wi
                                                                 /Eir-Appathon-NTUA/back-end)
 john in
                                                              eir.EirApplication : No active profile set, falling back
                                         main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data MongoDB
                                         main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository sca
ning in 58ms. Found 1 MongoDB repository interfaces.
                                                                                Tomcat initialized with port(s): 80
80 (http)
                                                                                Starting service [Tomcat]
                                                                                Starting Servlet engine: [Apache To
cat/9.0.371
                                                                                Initializing Spring embedded WebAp
licationContext
                                         main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initia
zation completed in 1116 ms
:020-08-12 22:50:49.625 INFO 1<mark>7036 --- [ main] org.mongodb.driver.cluster</mark>
==[localhost:27017], mode=SINGLE, requiredClusterType=UNKNOWN, serverSelectionTimeout='30000 ms']
                                                                                Cluster created with settings {hos
                                                                                Opened connection [connectionId{lo
alValue:1, serverValue:1}] to localhost:27017
                                                                                Monitor thread successfully co: ~$ sudo systemctl status mongod
Loaded: loaded (/lib/systemd/system/mongod.service; disabled; vendor preset:
                                                                                               Active: active (running) since
                                                                                                  Docs: https://docs.mongodb.org/manual
                                                                                            Main PID: 16577 (mongod)
                                                                                               CGroup: /system.slice/mongod.service
                                                                                                            └16577 /usr/bin/mongod --config /etc/mongod.conf
```

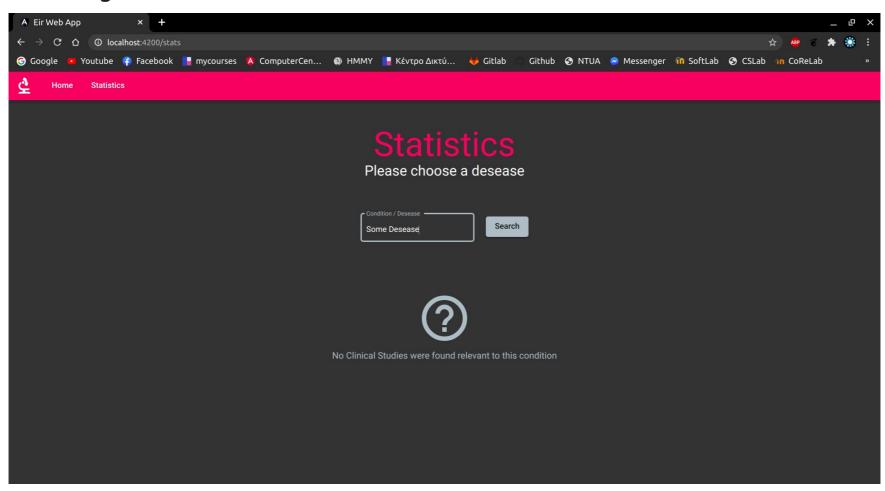
Home Page



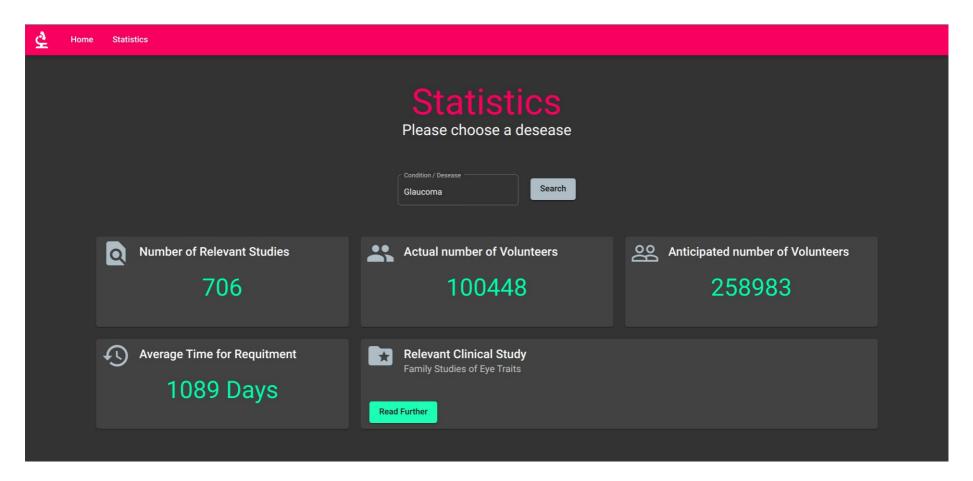




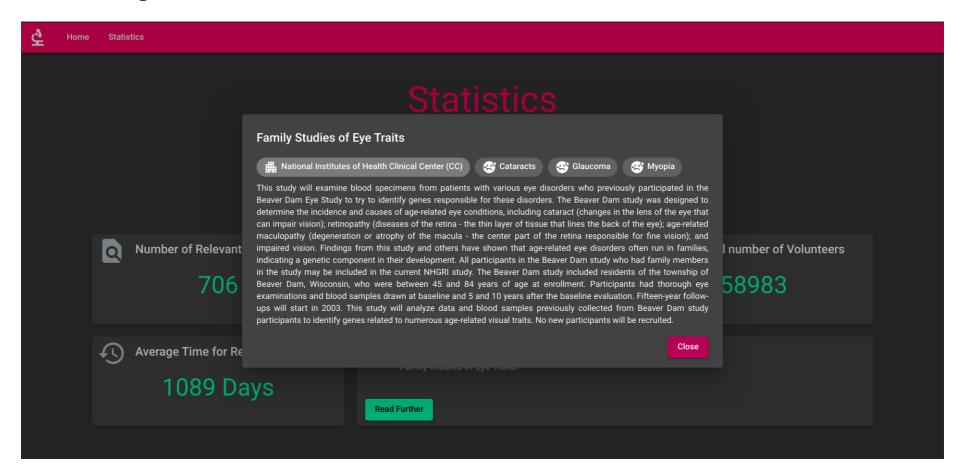
More Examples



More Examples



More Examples



Further Information

- GitHub Repository: https://github.com/john98nf/Eir-Appathon-NTUA
- Appathon 2020 NTUA: http://147.102.19.19/wordpress (probably unavailable)
- Internet Applications: https://www.ece.ntua.gr/en/undergraduate/courses/3346
- School of Electrical & Computer Engineering: https://www.ece.ntua.gr/en
- National Technical University of Athens: https://www.ntua.gr/en/

Closure

Thank you for your time!

Giannis Fakinos