Building Modern Web Applications & Services using Node.js





Eamonn de Leastar edeleastar@wit.ie https://edeleastar.github.io/oth-regensburg

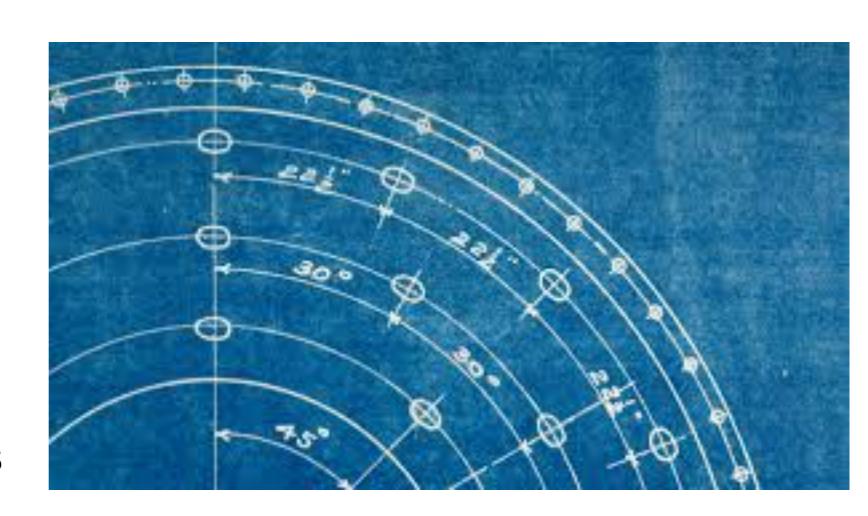
Course Mission

Transfer a set of foundation skills to enable you to design, build, secure, test and deploy a modern web application + API.



Agenda

- Prerequisites
- Preparing for the course
- Brief Overview
- Lab Requirements
- Assessment Guidelines



Perquisites

- Foundation level skills in:
 - HTML: Ability to effectively structure the html content of a small to medium static site, including the use of templates
 - CSS: Understand the fundamentals of CSS, and be able to realise simple layouts and designs
 - Javascript: Be familiar with the building blocks of the language and be able to compose realise algorithms to accomplish simple tasks.



5 Topics Top Level **Topics**



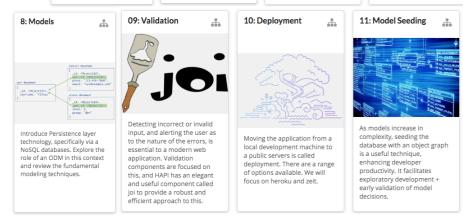




(1) Front End Foundation



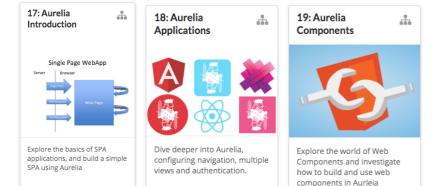
(2) Apis, Node & Hapi Applications



(3) Models, Persistence & Deployment



(4) Test Driven API Development

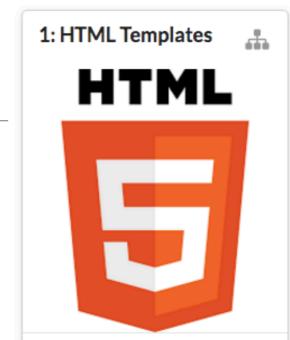


(5) Single Page Applications (optional)

(1) Front End Foundation

https://edeleastar.github.io/oth-regensburg

- Be able to structure and style a simple web site using html5, templating + a CSS framework.
- Understand the fundamentals of Javascript & JQuery



Review the fundamental structure of html5 documents, including templating techniques. Explore the basics of CSS + how to incorporate a CSS framework to simplify and enhance web site styling.



detail, leveraging it capabilities

to delivers rich UI experience.

Re-introdution to lavascript

3: JavaScript & DOM

Re-introdution to Javascript.
Review language structure and usage in the context of JQuery and client side script fragment.
Explore google maps API in this context.

- Material Already Published
- Covers the prerequisites
- Students should review/ complete this material prior to formal commencement of programme

Concepts

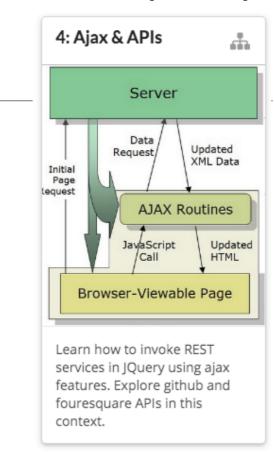
- Review Html + Css, focusing on templates + css frameworks
- Crash course in Javascript Fundamentals
- Learn basics of JQuery

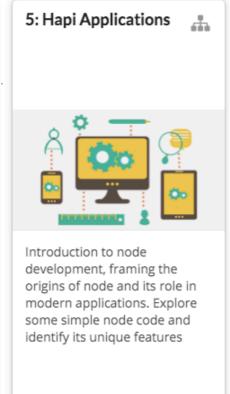
Tools:

- · Html5
- Semantic-ui
- jQuery
- · Chrome dev tools
- · DOM
- WebStorm IDE

(2) Apis, Node & Hapi Applications

- Be able to build a simple Node application incorporating templates views
- Understand and use the fundamentals of session management in the Concepts application







Explore the role of the HAPI application framework. Examine routing, connections, handlers and views within HAPI. Explore HAPI views in depthReview the DRY principle and the role of layouts. Employ the handlebars templating engine, incorporating partials and other constructs.



Review the role of Sessions in web applications. Incorporate the hapi-auth-cookie plugin into HAPI application to manage sessions. Review session strategies, protected and unprotected routes, cookie passwords, timeouts and other configuration options.

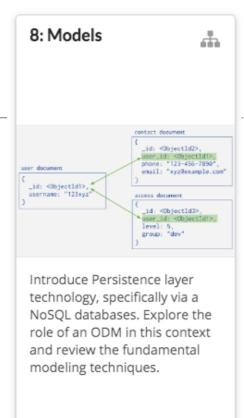
- Accessing APIs in Javascript
- Node.js Fundamentals
- Structure of a Hapi application
- Rendering views & templates
- Session Management

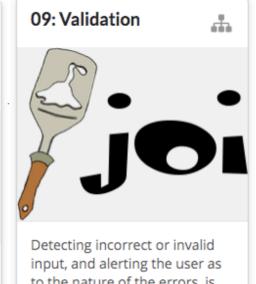
Tools:

- node & npm
- hapi.js
- inert, vision, handlebars, hapiauth-cookie

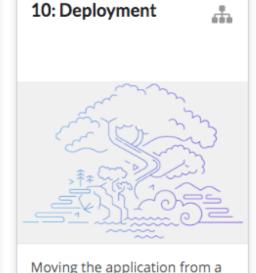
(3) Models, Persistence & Deployment

- Introduce
 persistence
 mechanisms into
 an hapi application
 and be able to
 employ data
 validation, seeding.
- Be able to deploy a hapi application





Detecting incorrect or invalid input, and alerting the user as to the nature of the errors, is essential to a modern web application. Validation components are focused on this, and HAPI has an elegant and useful component called joi to provide a robust and efficient approach to this.



local development machine to

deployment. There are a range

of options available. We will

focus on heroku and zeit.

a public servers is called

As models increase in complexity, seeding the database with an object graph is a useful technique, enhancing developer productivity. It facilitates exploratory development + early validation of model decisions.

11: Model Seeding

Concepts

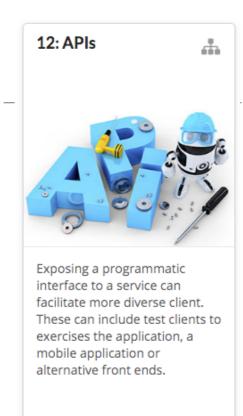
- Models & Schema
- Database access
- Validation of data
- Deployment

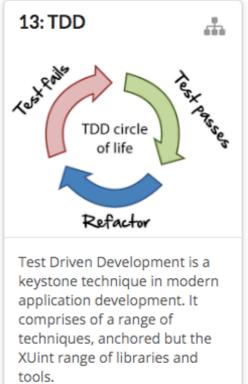
Tools

- MongoDB
- Heroku toolbelt
- Mongoose, mongooseseeder, joi

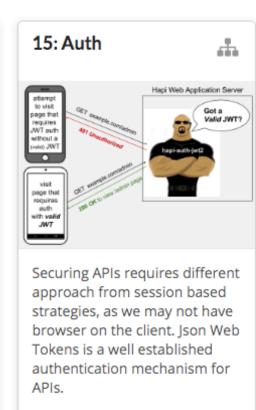
(4) Test Driven API Development

Be able to design, implement, test and secure a Restful API









Concepts

- Constructing a basic API
- TDD, Theory & Practice
- Fundamentals of REST
- Securing REST APIs

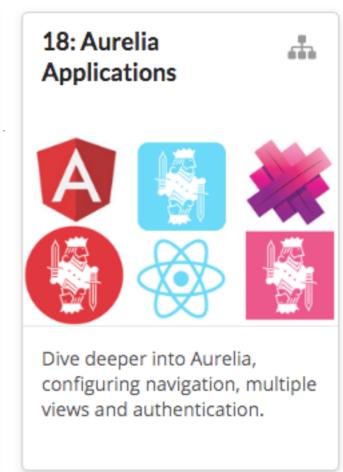
Tools:

 sync-request, mocha, chai, jsonwebtoken, hapi-auth-jwt2

(5) Single Page Applications (optional)

 Understand the SPA paradigm and be able to build a simple API driven SPA application







Concepts

- Single Page Applications
- Fundamentals of Aurelia.io
- Web Components
- Future of the web

Tools:

- aurelia.io
- gulp,
- aurelia-cli

before



structure of html5 documents techniques. Explore the basics of CSS + how to incorporate a enhance web site styling.



3: JavaScript & DOM 🙏

Review language structure and usage in the context of JQuery and client side script fragmen Explore google maps API in this

(1) Front End Foundation



4: Ajax & APIs

services in JQuery using ajax features. Explore github and fouresquare APIs in this



origins of node and its role in modern applications. Explore some simple node code and identify its unique features



detail, leveraging it capabilities

Explore the role of the HAPI Examine routing, connection handlers and views within HAPI. Explore HAPI views in depthReview the DRY principle and the role of layouts. Emplo the handlebars templating engine, incorporating partials and other constructs.



the hapi-auth-cookie plugin manage sessions. Review session strategies, protected and unprotected routes, cookie passwords, timeouts and other configuration options.

(2) Apis, Node & Hapi Applications



technology, specifically via a NoSQL databases. Explore the role of an ODM in this context and review the fundamental



nput, and alerting the user as to the nature of the errors, is application. Validation nents are focused on this, and HAPI has an elegant joi to provide a robust and



10: Deployment

Moving the application from a a nublic servers is called deployment. There are a range of ontions available. We will focus on heroku and zeit



complexity, seeding the database with an object graph enhancing developer productivity. It facilitates exploratory development early validation of model

(3) Models, Persistence & Deployment



nterface to a service car These can include test clients t alternative front ends



keystone technique in moderr application development. It comprises of a range of XUint range of libraries and



APIs are said to be Restful if the adhere to a set principles encompassing url patterns, payload and general structure



Securing APIs requires different strategies, as we may not have Tokens is a well established

(4) Test Driven API Development

after

17: Aurelia * Introduction Explore the basics of SPA SPA using Aurelia

18: Aurelia **Applications**

configuring navigation, multiple views and authentication

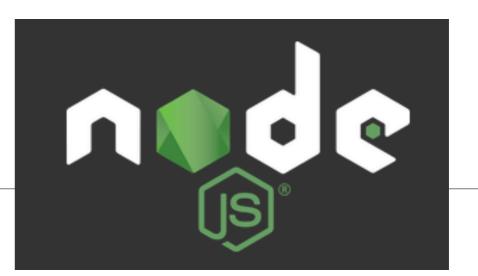


Components and investigate how to build and use web components in Aurleia

(5) Single Page Applications (optional)

Lab Requirements

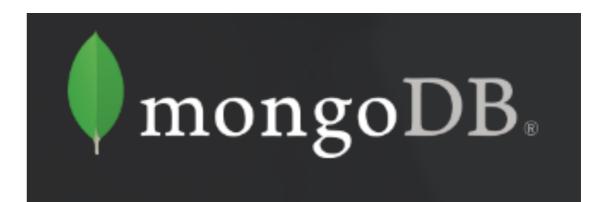
- WebStorm 2016
- Node.js
- Mongo DB
- Heroku-cli
- + additional libraries & tools as needed in labs



ws WebStorm

The smartest JavaScript IDE

Lightweight yet powerful IDE, perfectly equipped for complex client-side development and server-side development with Node.js





Assessment Guidelines

- 2 Components:
 - Assignment: A Node Application detailed specifications to be delivered in class

Research Note: A short paper (1500 words) on a specific research

topic (short list to be provided)

- Deadline: End of Semester
- Marks
 - Assignment: 80%
 - Research Node: 20%