

Building Modern Web Applications & Services using Node.js



Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

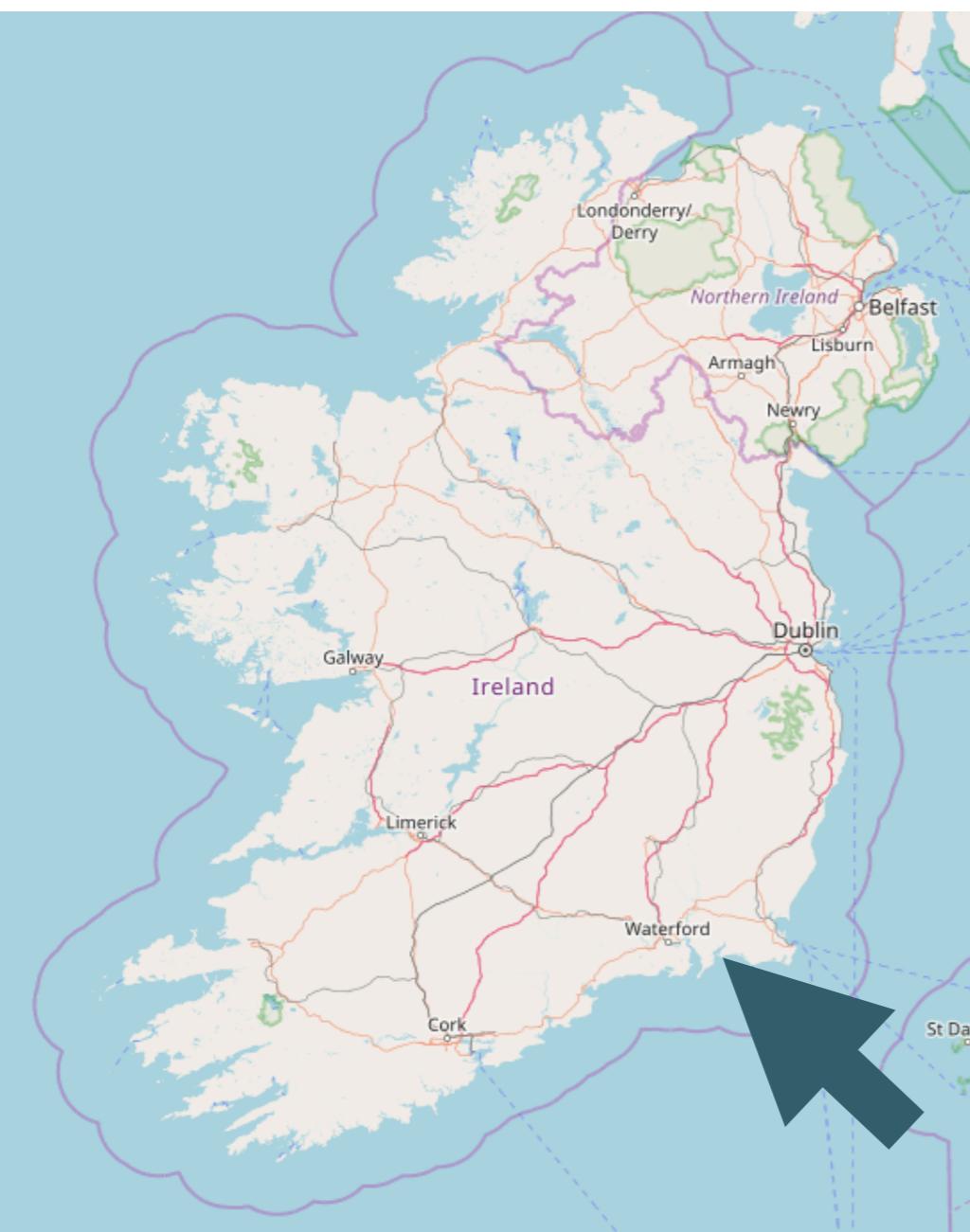


Eamonn de Leastar
edeleastar@wit.ie

<https://wit-oth-regensburg-2017-dmas.github.io>

<https://elearning.uni-regensburg.de/course/view.php?id=29928>

Waterford



Waterford (from Old Norse *Veðrafjørðr*, meaning "ram (wether) fjord", Irish: *Port Láirge*) is a city in Ireland. It is in the South-East Region, Ireland and is part of the province of Munster. The city is situated at the head of Waterford Harbour. It is the oldest^{[2][3]} and the fifth most populous city in the Republic of Ireland. It is the eighth most populous city on the island of Ireland.

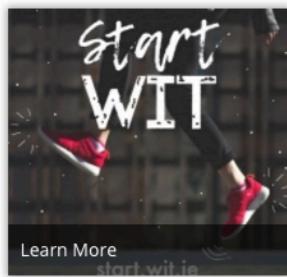


WIT A World of Imagination

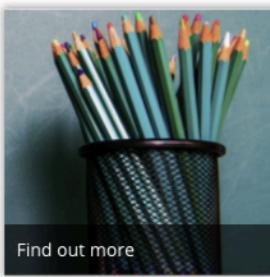
IMAGINE WIT - VISIT OUR CAO HUB

Advice and information to help you research your CAO options

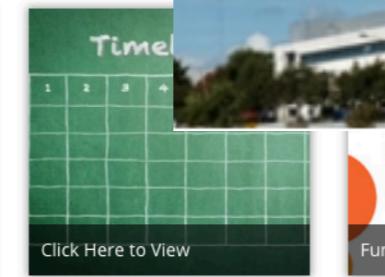
START WIT - REGISTRATION & ORIENTATION



PART TIME COURSES

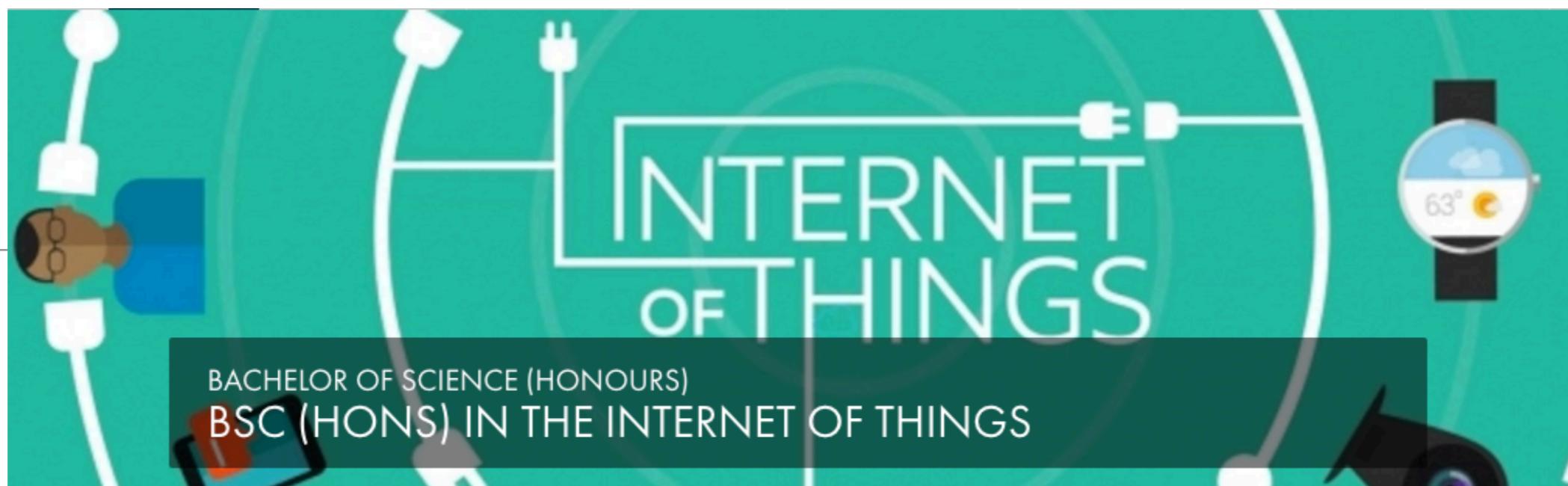


ACADEMIC TIME SEMESTER 1



Waterford Institute of Technology (WIT) is a university-level institution in the South-East of Ireland with over 10,000 students and 1,000 staff.

WIT offers tuition and research programmes in various areas from Higher Certificate to Degree to PhD.



Home » Courses » School » Science » Department of Computing, Maths & Physics

BY SCHOOL

Business

Lifelong Learning & Education

Engineering

Health Sciences

Humanities

Science and Computing

Code: WD197

Level: 8

Credits: 240 ECTS

Delivery: Full Time

Campus: Cork Road

2016 CAO range: 275 - 430

Duration: 4 years

APPLY FOR THIS COURSE



DESCRIPTION

OUTLINE

ENTRY

OPPORTUNITIES

PROJECTS

STORIES

CONTACT

Last Word Interview



Today FM Last Word Internet of Things Discussion

Matt Cooper, Eamonn de Leastar(WIT IoT Course Leader) & Barry Morris

0:00 / 11:44



- ▶ 1. Today FM Last Word Internet of Things Discussion by Matt Cooper, Eamonn de Leastar(WIT IoT Course Lead... 11:44
- 2. Limericks Live 95fm: The Internet of Things by Shane McAllister, Eamonn de Leastar & Anthony Quigney 20:07

Be part of the Internet of Things (IOT)

On this exciting new course developed in conjunction with the Institute's world class research groups TSSG, ACG, and CTRG, you will learn how to programme the next wave of connected devices, you will explore the software and hardware that is transforming the world, connecting things to create radical new services and products.



ABOUT » PROJECTS RESEAR

Home » About » People » Eamonn de Leastar

Eamonn de Leastar , Board Member



Eamonn de Leastar
Board Member
Ph: +353 (0)51 302 965
Email: edeleastar@tssg.org



Personal Summary

Eamonn de Leastar is a co-founder and Chief Technical Officer of the Software Systems Group (TSSG).

After graduation in 1982, he taught computer science at Dundalk Institute of Technology. He was responsible for a range of new curricula within this sector. He then moved into industry, gaining significant experience in senior programmer, analyst, quality assurance and project leadership roles. This included periods at Wordstar International, developing some of the first word processors on the then newly established PC standard and at Contel Business Systems



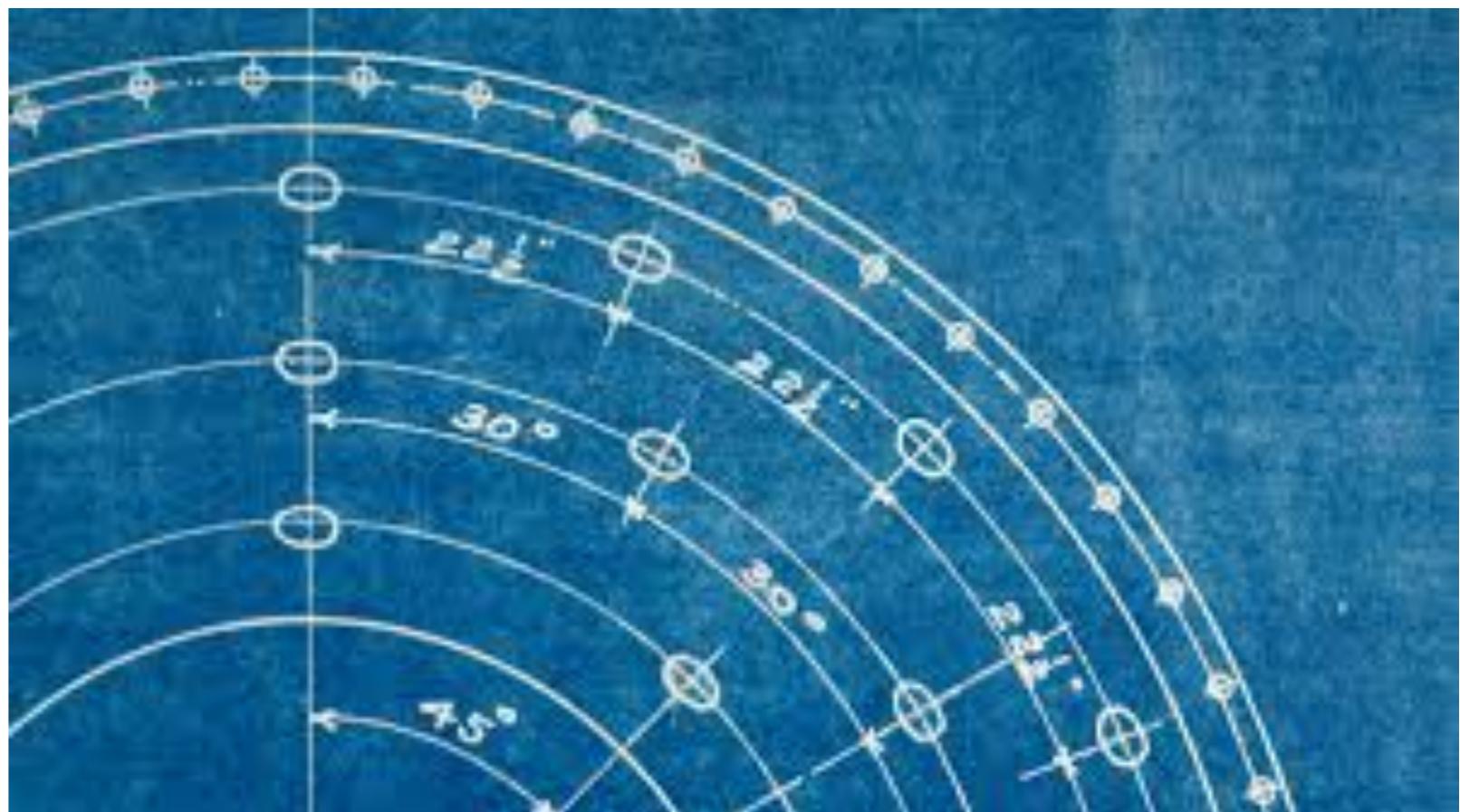
Course Mission

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
- Schedule

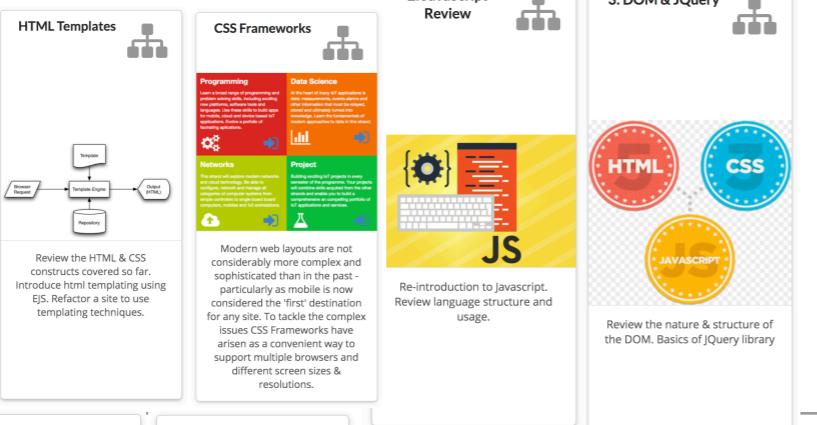


Perquisites

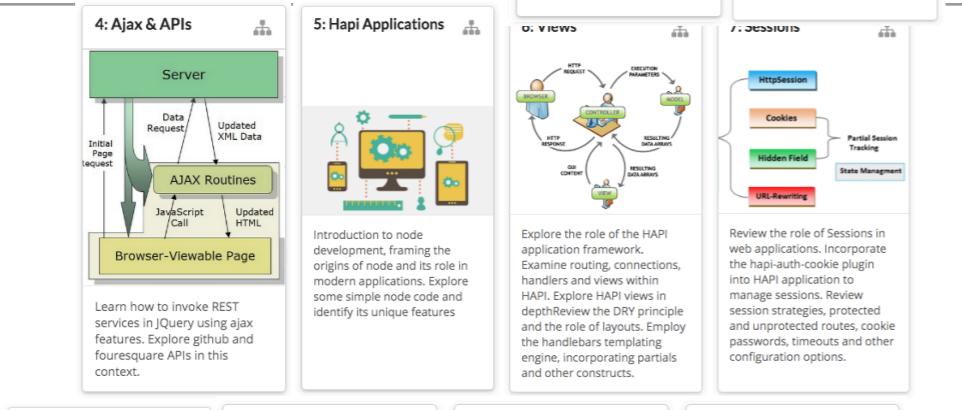
- Foundation level skills in:
 - HTML: Ability to effectively structure the html content of a small to medium static site, including the use of templates
 - CSS: Understand the fundamentals of CSS, and be able to realise simple layouts and designs
 - Javascript: Be familiar with the building blocks of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



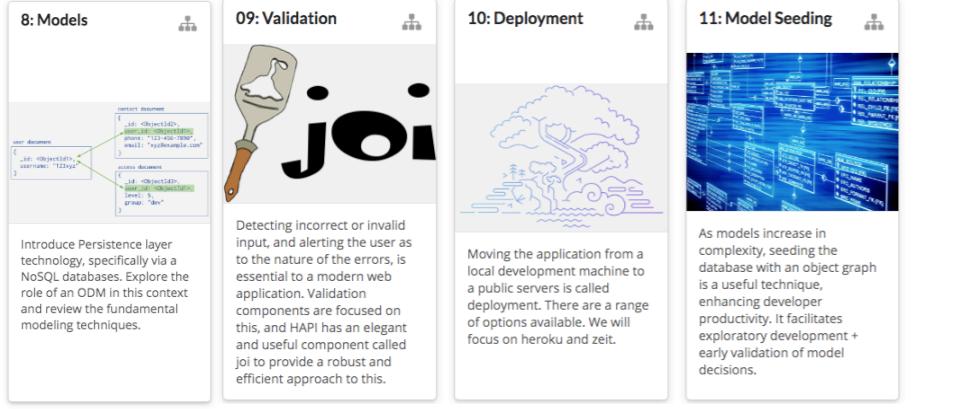
5 Topics Top Level Topics



(1) Front End Foundation (+JS)



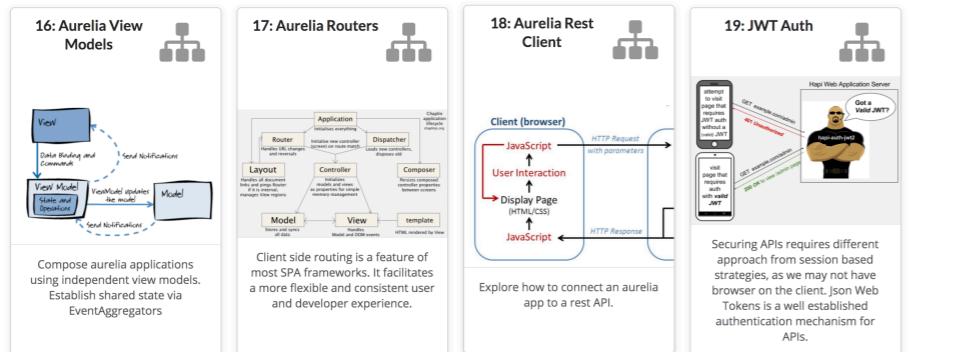
(2) Apis, Node & Hapi Applications



(3) Models, Persistence & Deployment



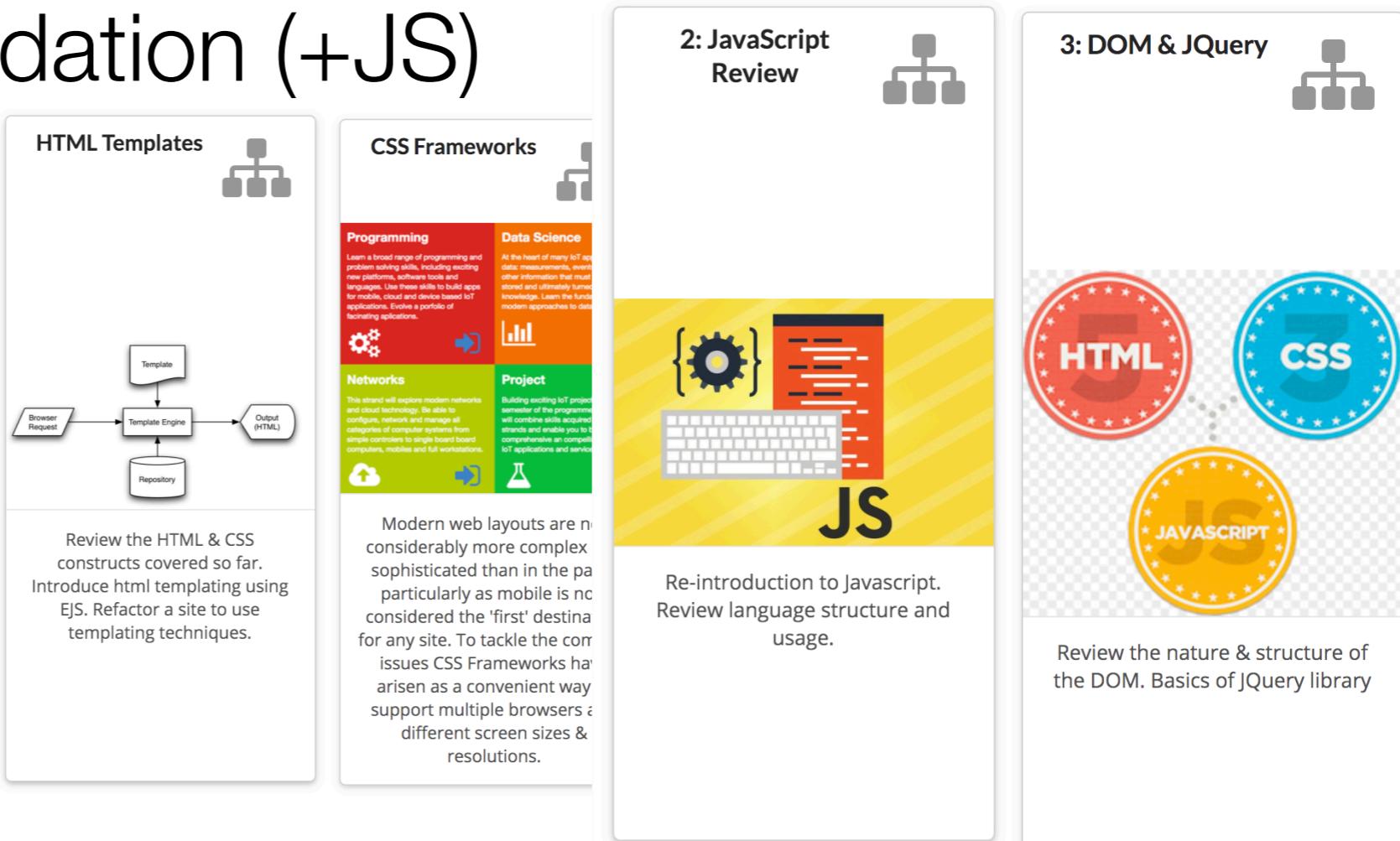
(4) Test Driven API Development



(5) Single Page Applications (TypeScript)

(1) Front End Foundation (+JS)

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



Concepts

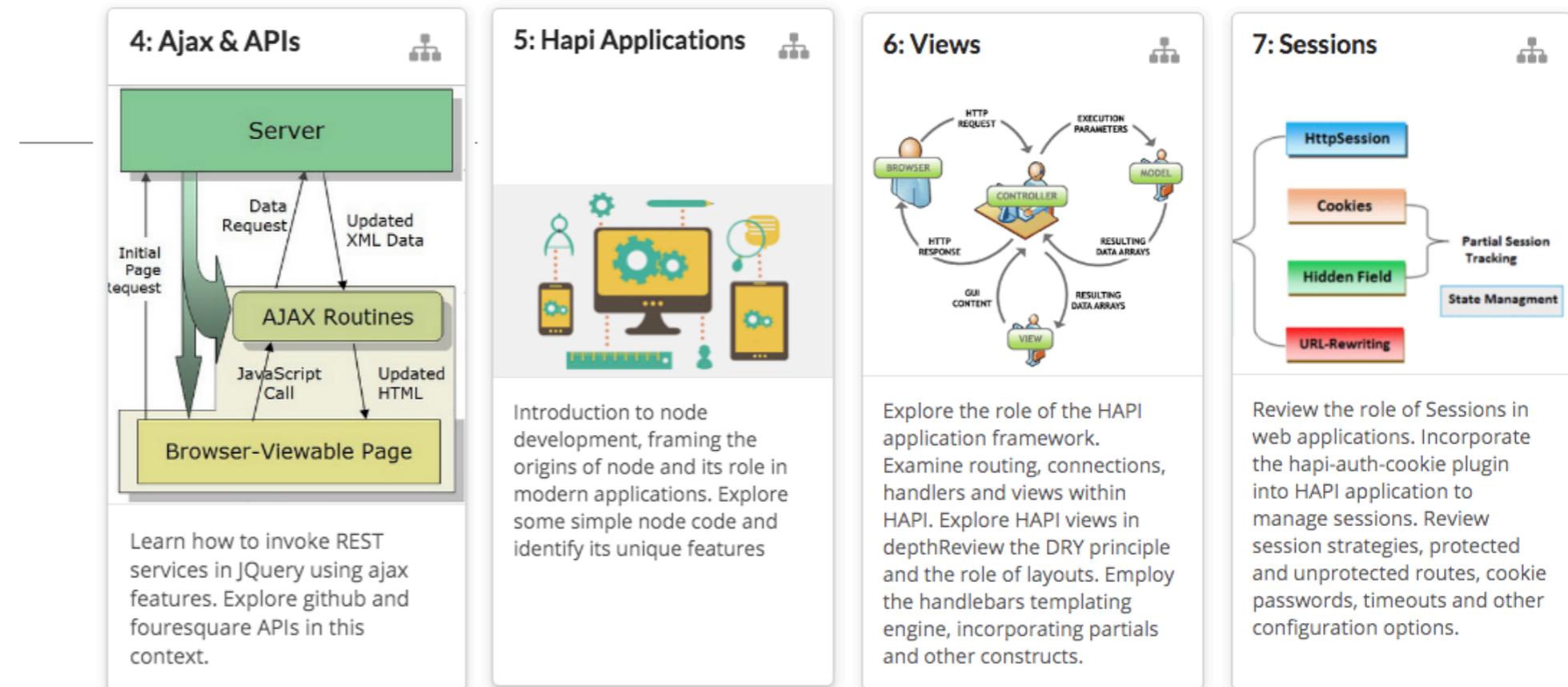
- Review Html + Css, focusing on templates + css frameworks
- Crash course in Javascript Fundamentals
- Learn basics of JQuery

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 application

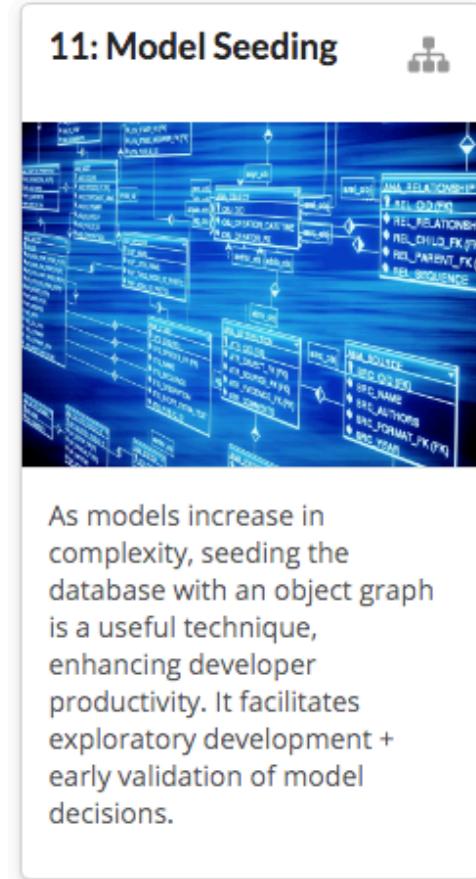
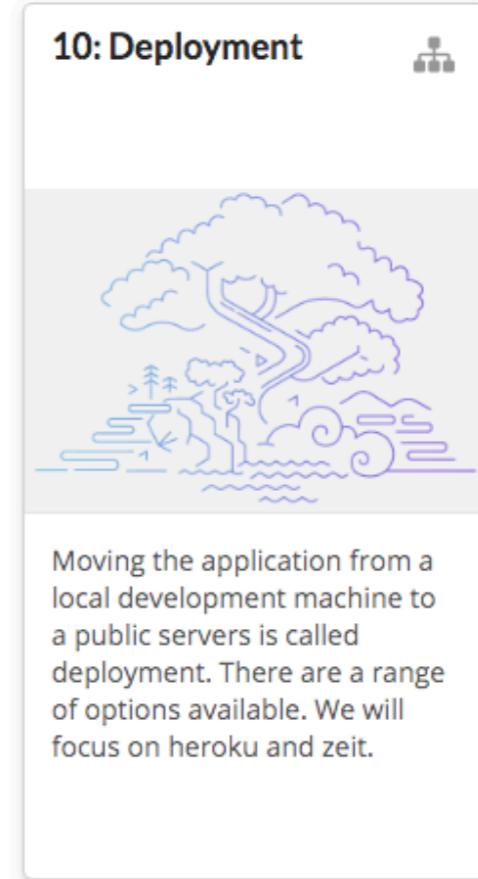
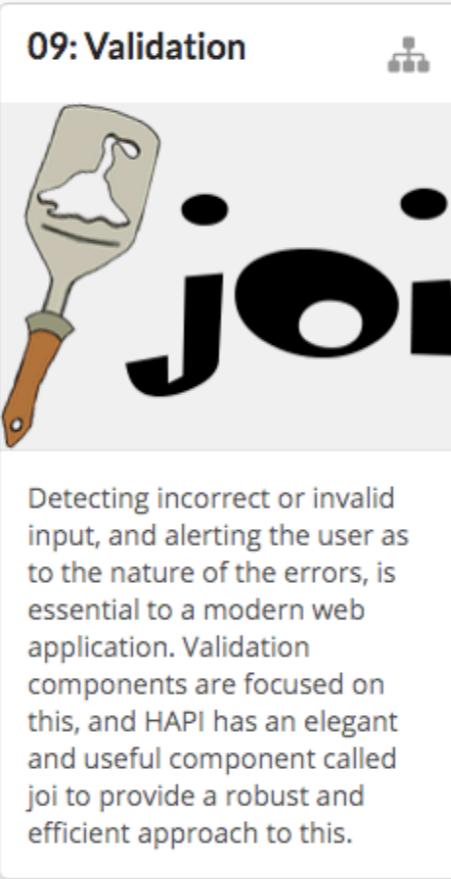
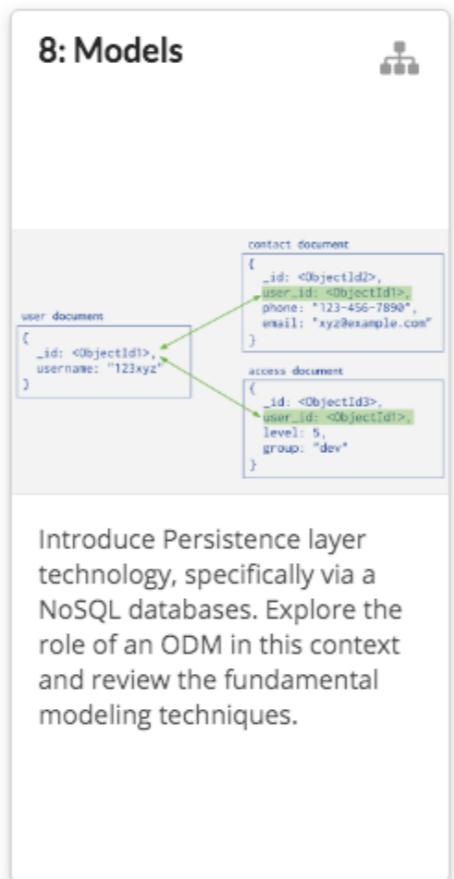


management in the application

Concepts	<ul style="list-style-type: none">• Accessing APIs in Javascript• Node.js Fundamentals• Structure of a Hapi application• Rendering views & templates• Session Management	Tools: <ul style="list-style-type: none">• node & npm• hapi.js• inert, vision, handlebars, hapi-auth-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



Concepts

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

Tools

- MongoDB
- Heroku toolbelt
- Mongoose, mongoose-seeder, joi

(4) Test Driven API Development

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

12: APIs  <p>Exposing a programmatic interface to a service can facilitate more diverse client. These can include test clients to exercises the application, a mobile application or alternative front ends.</p>	13: TDD <p>Test Driven Development is a keystone technique in modern application development. It comprises of a range of techniques, anchored but the XUnit range of libraries and tools.</p>	14: REST  <p>APIs are said to be Restful if they adhere to a set principles encompassing url patterns, payload and general structure.</p>	15: Auth  <p>Securing APIs requires different approach from session based strategies, as we may not have browser on the client. Json Web Tokens is a well established authentication mechanism for APIs.</p>
--	---	--	---

Concepts

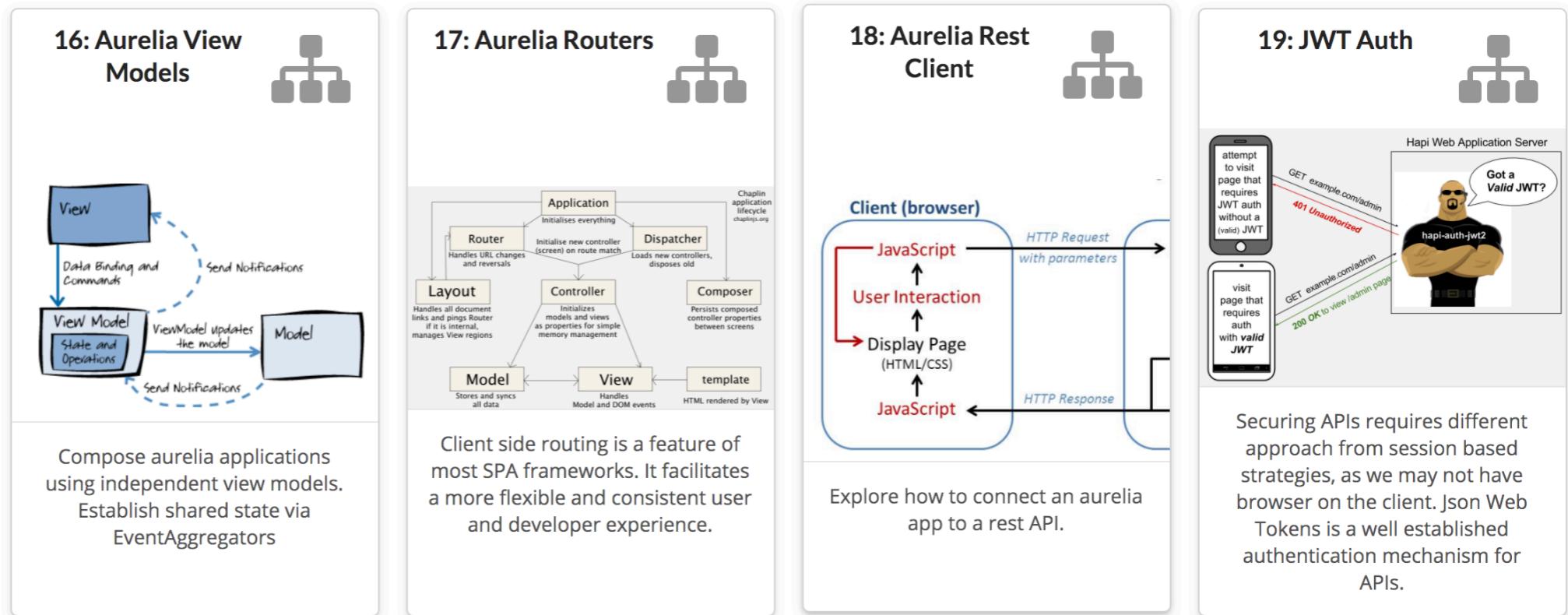
- Constructing a basic API
- TDD, Theory & Practice
- Fundamentals of REST
- Securing REST APIs

Tools:

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

(5) Single Page Applications

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

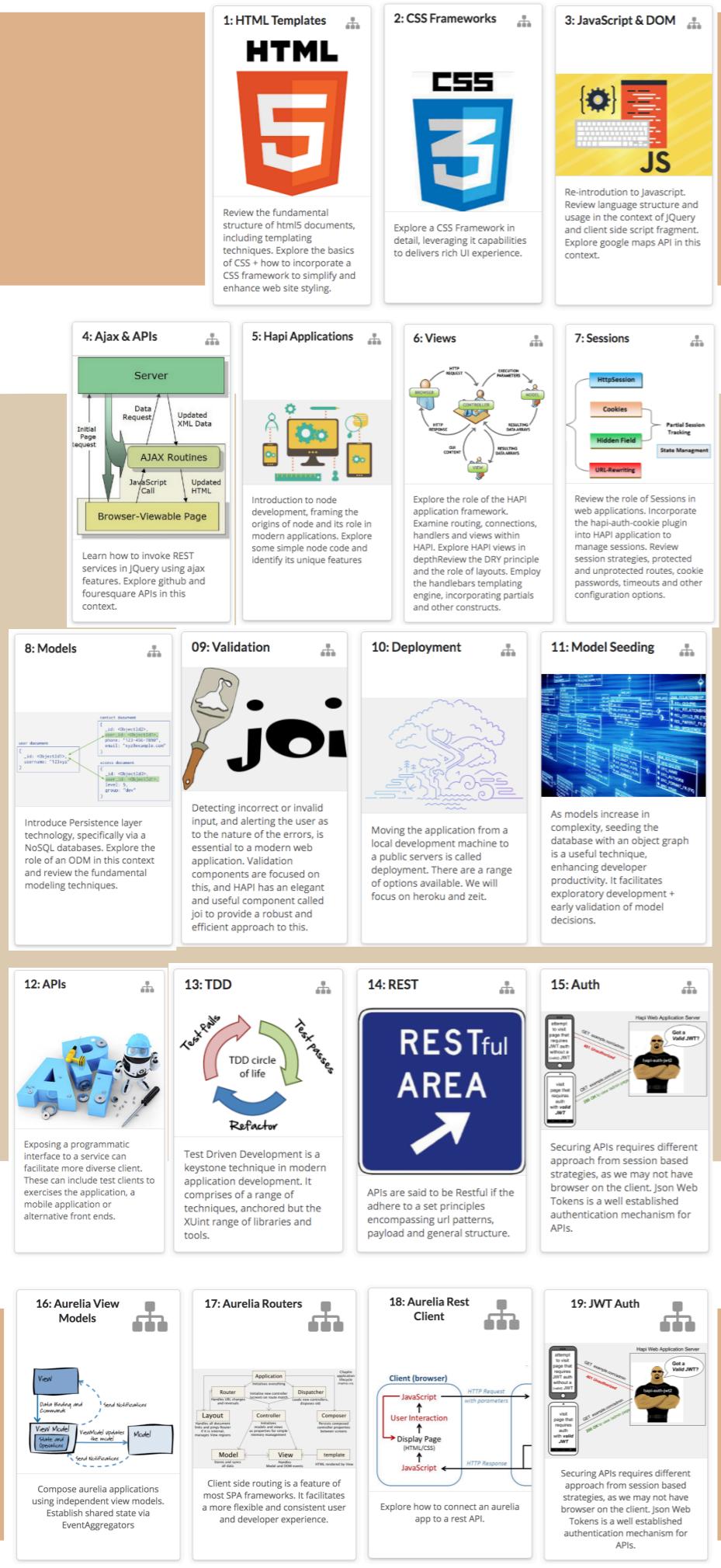


Concepts

- Typescript
- Single Page Applications
- Fundamentals of Aurelia.io
- Aurelia View Models, Routers
- JWT

Tools:

- aurelia.io
- [aurelia-cli](https://github.com/aurelia/cli)



(1) Front End Foundation

(2) APIs, Node & Hapi Applications

(3) Models, Persistence & Deployment

(4) Test Driven API Development

(5) Single Page Applications (optional)

Assessment

Single Project - submitted at
end of semester (late
December 2017)

Assessment Specification

<https://wit-oth-regensburg-2017-dmas.github.io/topic-00-intro>

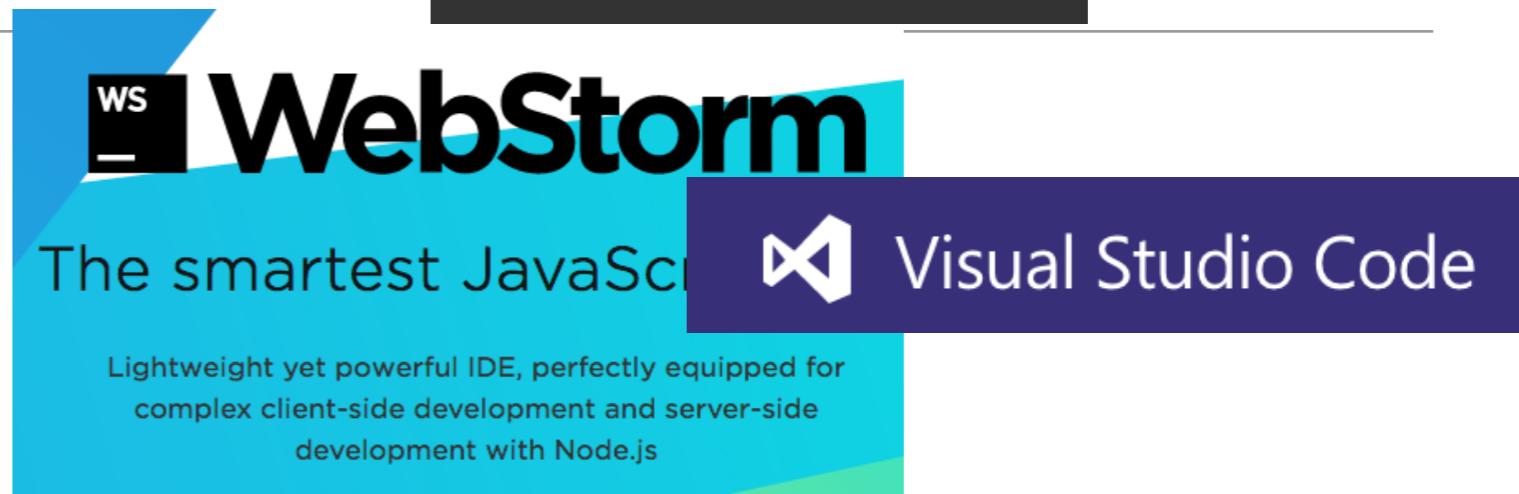
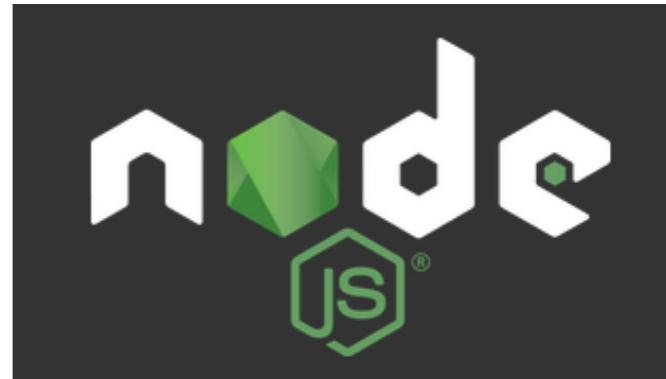
talks

Assessment

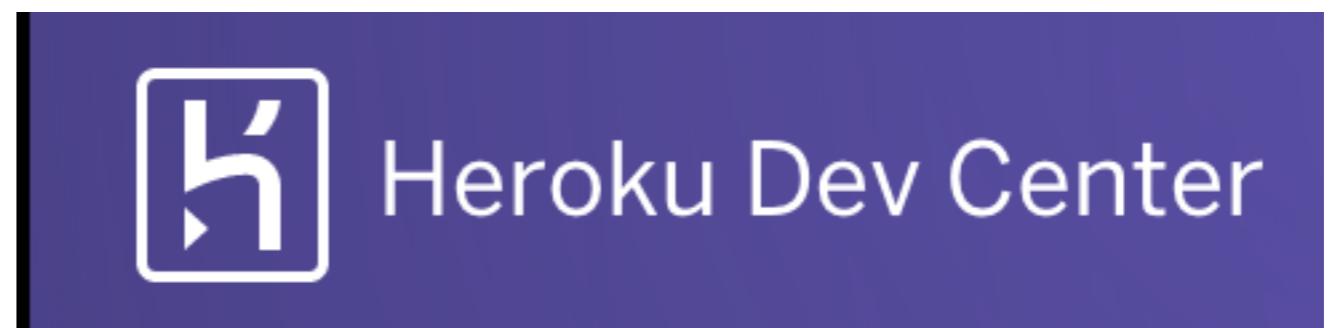
The specification, submission guidelines and grading spectrum for the Assignment for this course.

Lab Requirements

- WebStorm 2016
- VSCode
- Node.js
- Mongo DB
- Robo 3T
- Heroku-cli
- + additional libraries & tools as needed in labs



Robo 3T



Schedule (pre and post tuition)

Pre Tuition				
	02/10/2017	09/10/2017	16/10/2017	23/10/2017
18:45 - 20:15	00: Course Overview	01 HTML Templates	02 CSS Frameworks	03 Javascript Review I

	Post Tuition					
	06/11/2017	13/11/2017	20/11/2017	27/11/2017	04/12/2017	11/12/2017
18:45 - 20:15	18 Aurelia Intro	19 Aurelia View/ Models	20 Aurelia Routers	21 Aurelia Rest Client	22 JWT Auth	23 Aurelia & JWT

	Sat:28h
10:00-11:00	01-03 Introduction + Javascript Review
11:00-11:15	Coffee Break
11:15-12:30	04 Ajax + APIs I
12:30-1:15	Lunch
1:15-3:00	04 Ajax + APIs II
3:00-3:15	Coffee Break
3:15-4:45	05 Node + Hapi

	Thursday 2nd	Friday: 3rd	Saturday 4h
10:00-11:00		08 Models	12 APIs
11:00-11:15		Coffee Break	Coffee Break
11:15-1:15		09 Validation	13 TDD
1:15-2:00			Lunch
2:00-3:15			14 REST
3:15-3:30			Coffee Break
3:30-4:45		10 Deployment	Assignment Briefing
4:45-5:00		Coffee Break	
5:00-6:15	05 Views	11 Modeel Seeding	
6:15-6:30	Coffee Break		
6:30-8:15	07 Sessions		

Discussion Forum also here:

<https://elearning.uni-regensburg.de/course/view.php?id=29928§ion=1>

Home ► Courses ► OTH Regensburg ► Fakultät Informatik und Mathematik ► Internationale Gastdozierende ► YMWA-de Leastar-IN,IW-WiSe17/18

QUICKLINKS

- Meine Kurse (Dashboard)
- Mitteilungen
- Logout

NAVIGATION

- Home
 - Dashboard
 - Site pages
 - Current course
 - YMWA-de Leastar-IN,IW-WiSe17/18
 - Participants
 - Badges
 - General
 - 0: Overview & Assessment
 - Assignment
 - 1: HTML & CSS
 - 2: CSS Frameworks
 - 3: Javascript Introduction
 - JQuery & DOM
 - My courses

Course: Building Modern Web Applications & Services using Node.js – de Leastar – IN,IW – WiSe17/18, Topic: 0: Overview & Assessment

Nachrichtenforum

0: Overview & Assessment

1: HTML & CSS

Your progress 

Course Overview

A brief overview of the entire course

Assessment

The specification, submission guidelines and grading spectrum for the Assignment for this course.

Assignment

<https://wit-oth-regensburg-2017-dmas.github.io>

https://wit-oth-2017-dmas.git x Eamonn

Secure | https://wit-oth-2017-dmas.github.io

Building Modern Web Applications & Services

Module Home

Prepared for OTH Regensburg by Eamonn de Leistar, WIT.
Creative Commons License

1: Introduction

2: HTML & CSS Revision

3: JavaScript Review

4: DOM & JQuery

Introducing the module, assessment and schedule.

Ask Questions here:

OPEN CHAT

<https://wit-oth-regensburg-2017-dmas.github.io>

The screenshot shows a web browser window with the URL <https://wit-oth-2017-dmas.github.io>. The page displays a course landing page for "Building Modern Web Applications & Services". The landing page features a sidebar with various course modules and a main content area with a large title box containing the text: "Public Chat Room: Requires account/ sign in on <https://gitter.im> first". Below this, another box contains the text: "I will drop in occasionally to answers any questions or chat in general about course". A small portrait of a man with glasses and a blue beanie is visible on the right side of the page. At the bottom, there is a chat interface with a message placeholder: "Click here to type a chat message. Supports GitHub flavoured markdown."

Building Modern Web Applications & Services

good day all!

1: Introduction

1: HTT

Public Chat Room:
Requires account/
sign in on <https://gitter.im> first

I will drop in
occasionally to
answers any
questions or chat in
general about
course

Click here to type a chat message. Supports GitHub flavoured markdown.

2016 course is still online.

However,
assessment and
content will be
refreshed

The screenshot shows a web browser displaying the course website at <https://edeleastar.github.io/oth-regensburg>. The page title is "Building Modern Web Applications & Services". The top navigation bar includes links for "All Slides" and "All Labs". Below the title, a section titled "Top Level Topics" lists 11 topics, each with a thumbnail, title, and brief description:

- 0: Assignments**: Calendar and specifications for the assessments on the course.
- 1: HTML Templates**: 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.
- 2: CSS Frameworks**: Explore a CSS Framework in detail, leveraging its capabilities to deliver rich UI experience.
- 3: JavaScript & DOM**: Re-introduction to Javascript. Review language structure and usage in the context of jQuery and client-side script fragments. Explore Google Maps API in this context.
- 4: Ajax & APIs**: Learn how to invoke REST services in jQuery using Ajax features. Explore GitHub and Foursquare APIs in this context.
- 5: Hapi Applications**: Introduction to node development, framing the origins of node and its role in modern applications. Explore some simple node code and identify its unique features.
- 6: Views**: Explore the role of the HAPI application framework. Examine routing, connections, handlers and views within HAPI. Explore HAPI views in depth. Review the DRY principle and the role of layouts. Employ the Handlebars templating engine, incorporating partials and other constructs.
- 7: Sessions**: Review the role of sessions in web applications. Incorporate the hapi-auth-cookie plugin into HAPI applications to manage sessions. Review session strategies, protected and unprotected routes, cookie passwords, timeouts and other configuration options.
- 8: Models**: Introduce Persistence layer technology, specifically via a NoSQL database. Explore the role of an ODM in this context and review the fundamental modeling.
- 9: Validation**: 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 and HAPI has an elegant
- 10: Deployment**: Moving the application from a local development machine to a public servers is called deployment. There are a range of options available. We will focus on heroku
- 11: Model Seeding**: As models increase in complexity, seeding the database with an object graph is a useful technique, enhancing developer productivity. It facilitates exploratory

The URL at the bottom of the page is <https://edeleastar.github.io/oth-regensburg/topic-01-html/index.html>.