# Lab 9: Express & EJS

***The objectives of the lab:***

1. *Be able to serve pages (index.html, about.html and 404.html) using EJS and inject data into EJS templates to generate HTML.*
2. *Decompose the pages to reusable components (partials in EJS).*
3. Download the provided starter code from Moodle. Check the structure of your project. Try opening the index.html page using a browser. The data that is displayed is statically written inside the document. You are given a data.js file that contains the same data organized as an array of javascript objects.
4. Your project is currently not a node application. Initialize it, to create a package.json file so that you can use express and ejs (npm init -y).
5. Install express, dotenv and ejs (npm install express ejs dotenv).
6. In the .env file, set a value for the port that will be used by the application: PORT=….. (use a value between 1024 to 49151, these are the user ports, the others are reserved 0->1023 and until 65535, more about this here: <https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml>).
7. Declare the public folder as a static folder to be able to serve any files statically (like css files and images):

app.use(express.static('public'));

1. Register ejs as the view engine to be used by express:

app.set('view engine', 'ejs');

1. Move your pages to the views folder (create it first). Rename your html pages to the .ejs file extension. Serve your pages using app.get(routeHere, callback) using res.render():

app.get('/page', (request, response) => {

response.render('index', {data1: value1, data2: value2});

})

1. Replace the cards in your html pages by EJS generated html. You will need to pass the loaded data from data.js to the index template.

<% …… %> to include javascript

<%= …… %> to include the value of a variable.

1. Decompose your three pages by identifying common parts. Include these parts in partials files.

<%- include(‘file’, { data: value} %> To include a partial. **END**