LARBI BEN M'HIDI UNIVERSITY OUM EL BOUAGHI
FACULTY OF EXACT AND NATURAL SCIENCES AND LIFE SCIENCES
DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE

Level: 2nd year Computer Science **Degree Module: web application development**
Date: 10-05-2025 **Duration: 90 minutes**

Exam N° 01

Exercise N 01: (08 pts)

1) Give the meaning and role of the following keywords: **CSS**, **HTTPS**, **PHP**, **XML**.

2) Describe the main steps of service-oriented architecture and its importance in web development.

3) Explain the difference between :

a) Static and dynamic web page.

b) Website and web application.

c) DOM and **DTD** .

4) Explain how to integrate **CSS** into an **HTML document** .

Exercise N 02: (12 pts)

1- Create an HTML5 page with:

- A header containing a navigation menu (**Home, Services** , **Contact**).
- A main section with a contact form.
- A footer.
- Use CSS for styling.

The form must contain:

- Name (required),
- E-mail,
- Password,
- Message (text area),
- Submit button.

2- Validate the previous form using JavaScript with the following rule:

- Name: 4 characters minimum.

Good luck

Typical correction:

Exercise 01: (08 pts)

1) Meaning and role of keywords (……..03 pts)

- **CSS (Cascading Style Sheets):** A style language used to describe the presentation of an HTML document. It allows you to control layout, colors, fonts, etc. **(…… 0.75 pt)**
- **HTTPS** (HyperText Transfer Protocol Secure): A secure communication protocol for transferring data over the web, using SSL/TLS encryption. **(…… 0.75 pt)**
- **PHP** (Hypertext Preprocessor): Server-side programming language used to develop dynamic web applications. **(…… 0.75 pt)**
- **XML** (eXtensible Markup Language): An extensible markup language used to structure and transport data. **(…… 0.75 pt)**

2) Service-oriented architecture (……..01, 50 pts)

The architecture of web services consists of three main roles:

1. **Service Provider:** The entity that creates and hosts the web service.
2. **Service Requestor (Client):** The application that consumes the web service.
3. **Service Registry (Optional):** A directory where services are published and discovered.

The interaction follows these steps:

1. **Publish:** The provider registers the service in a registry.
2. **Find:** The client searches for the required service.
3. **Bind:** The client connects to the provider and invokes the service.

Importance: Enables a modular, cross-platform approach, promotes reuse, facilitates integration between systems and improves maintainability.

3) Differences (……..03 pts)

a) Static vs. dynamic web page: (……..01pt)

- **Static:** Fixed content, identical for all users (pure HTML)
- **Dynamic:** Content generated on demand, customizable (PHP , etc.)

b) Website vs web application: (……..01pt)

- **Website:** A set of linked web pages, primarily informative content.
- **Web application:** Interactive program with complex functionalities (data processing).

c) DOM vs DTD: (……..01pt)

- **DOM (Document Object Model):** Object representation of an HTML document allowing its manipulation .
- **DTD (Document Type Definition):** Definition of the structure and rules of an XML document.

4) CSS integration into HTML (……..01, 50 pts)

Three main methods:

1. <style> tag in the <head> .
2. Style attribute directly on HTML elements .
3. External file with <link rel="stylesheet" href="style.css"> .

Exercise No. 02 (12 pts)

1) HTML5 + CSS page (……..08 pts)

```
<!DOCTYPE html>
< html lang = "en" >
< head >
  <!-- Definition of metadata and page title -->
  < meta charset = "UTF-8" >
  < title > Contact </ title >
(……..0.5 pt)
  < style >
/* CSS styles for layout */ (……..01, 50 pts)
body {
font-family: Arial, sans-serif;
line-height: 1.6;
margin: 0;
padding: 0;
}
/* Header style */
header {
background: #333;
color: #fff;
padding: 1st;
}
/* Navigation style */
```

```
nav ul {
display: flex;
list-style: none;
padding: 0;
}
nav ul li {
margin-right: 1st;
}
nav ul li a {
color: #fff;
text-decoration: none;
}

/* Form style */
form {
max-width: 600px;
margin: 0 auto;
}
.form-group {
margin-bottom: 1st;
}
label {
display: block;
margin-bottom: 0.5rem;
}
input, textarea {
width: 100%;
padding: 0.5rem;
border: 1px solid #ddd;
border-radius: 4px;
}
button {
background: #333;
color: #fff;
padding: 0.5rem 1rem;
border: none;
border-radius: 4px;
cursor: pointer;
}
/* Footer style */
footer {
background: #333;
color: #fff;
```

```
text-align: center;
padding: 1st;
position: fixed;
bottom: 0;
width: 100%;
}
  </ style >
</ head >
< body >
  <!-- Page header with navigation --> (.....01pts)
  < header >
    < nav >
      < ul >
        < li >< a href = "#" > Home </ a ></ li >
        < li >< a href = "#" > Services </ a ></ li >
        < li >< a href = "#" > Contact </ a ></ li >
      </ ul >
    </ nav >
  </ header >

  <!-- Main content with contact form --> (.....01pts)
      < form id = "contactForm" >
      <!-- Name Field (required) -->
      < div class = "form-group" >
        < label for = "name" > Name </ label >
        < input type = "text" id = "name" name = "name" required >
      </div>

      <!-- Email field -->     (.....01pts)
      < div class = "form-group" >
        < label for = "email" > Email </ label >
        < input type = "email" id = "email" name = "email" >
      </div>

      <!-- Password Field --> (.....01pts)
      < div class = "form-group" >
        < label for = "password" > Password </ label >
        < input type = "password" id = "password" name = "password" >
      </div>

      <!-- Message field --> (.....01pts)
      < div class = "form-group" >
        < label for = "message" > Message </ label >
```

```
        < textarea id = "message" name = "message" rows = "5" ></ textarea >
      </div>

      <!-- Submit button --> (……0.5 pt)
      < button type = "submit" > Submit </ button >
    </ form >

  <!-- Footer --> (……0.5 pts)
  < footer >
    < p >&copy; 2025 My Website </ p >
  </ footer >
</ body >
</ html >
```

2) JavaScript Validation (……0 4 pts)

```
  <!-- JavaScript validation script -->
  <script> ( …… 0.5 pt)
// Event listener for form submission (……01 pt)
document.getElementById('contactForm').addEventListener('submit', function(e) {
e.preventDefault(); // Prevents sending by default

// Validation of the Name field (minimum 4 characters) (……01, 50 pts)
const name = document.getElementById('name').value;

if(name.length < 4) {
alert('Name must contain at least 4 characters');
return false;
}

// Success message if validation OK (……01 pt)
alert('Form validated successfully');
this.submit();
return true;
});
  </ script >
```