CCT | College Dublin
Computing • IT • Business

Theory & Practical Electricity Bill App

Students:

Charles Franklin Jahn...2020315
Xiaohui Weng...............2020387

Lecturer: Dr Mohammed Iqbal

19.05.2022
BSc (Hons) in Computing in IT - 2nd Year
Module: Cross Platform Development

Code

Xiaohui and I decided to create this app, Electricity Bill Calculator, using Loan Calculator App, from Tutorial week 11.

So we made a lot of changes, we divided the tasks as initial, middle and final.

Initial:

This part was made for me (Charles), the tasks were to create Github Repository, copy and paste the Loan Calculator App, in the directory, and start changing the code, like,change variables' names, classes' names, ids' names and add new necessaries inputs and the CSS.

Code screenshots Charles' Task:

Div main box:

In this part of the code I did all the necessary changes and added one more div field,
I decided to give the option of ability to change the VAT and electricity rate, although they are setted as requested in the CA3 descriptor.

```html
<h1 id="app-title">Electricity Bill</h1>
<div id="form-holder">
    <div id="main-box">
        <form action="" id="calculate-Bill-form">
            <div class="form-item">
                <label for="number_Units"><b>Number of Units</b></label>
                <input type="number" step="any" min="0" name="number_Units"
                id="number_Units"
                    placeholder="xxxxx" class="input-item text-right" required>
            </div>
            <div class="form-item">
                <label for="days_for_bill"><b>Billing Period (days)</b></label>
                <input type="number" step="any" min="1" name="days_for_bill"
                id="days_for_bill" placeholder="x"
                    class="input-item text-right" required>
            </div>
            <div class="form-item">
                <label for="electriRate"><b>Electricity Rate</b></label>
                <input type="number" step="any" name="electriRate" id="electriRate"
                placeholder="20.0%"
                    class="input-item text-right" value="0.20" required>
            </div>
            <div class="form-item">
                <label for="vat"><b>VAT</b> <small>(%)</small></label>
                <input type="number" step="any" name="vat" id="vat" placeholder="13.5%"
                    class="input-item text-right" value="13.5" required>
            </div>
            <div class="form-item">
                <button class="btn-block" id="calculate-btn" type="submit">Calculate</
                button>
                <button class="btn-block" id="reset-btn" type="reset">Reset</button>
            </div>
        </form>
        <table id="result">
```

Table result:

For this part of the code only the id's names were changed.

```html
<tbody>
    <tr>
        <td class=""><b>Total of Units</b></td>
        <td class="text-right" id="total_Of_Units"></td>
    </tr>
    <tr>
        <td class=""><b>VAT(%)</b></td>
        <td class="text-right" id="VAT"></td>
    </tr>
    <tr>
        <td class=""><b>Period Bill (Days)</b></td>
        <td class="text-right" id="bill_period_days"></td>
    </tr>
    <tr>
        <td class=""><b>Electricity Rate</b></td>
        <td class="text-right" id="electriRateFinal"></td>
    </tr>
    <tr>
        <td class=""><b>Bill without VAT</b></td>
        <td class="text-right" id="total-pay"></td>
    </tr>
    <tr>
        <td class=""><b>Total with Vat</b></td>
        <td class="text-right" id="TotalwithVat"></td>
    </tr>
</tbody>
```

Script.js:

In script .js my task was to change the variables' names.

```javascript
const totalUnits = document.getElementById('number_Units').value;
const electriRate = document.getElementById('electriRate').value;
const daysOfBill = document.getElementById('days_for_bill').value;
const vatRate = document.getElementById('vat').value;
```

```javascript
document.getElementById('total_Of_Units').textContent = ""
document.getElementById('VAT').textContent = ""
document.getElementById('bill_period_days').textContent = ""
document.getElementById('TotalwithVat').textContent = ""
document.getElementById('total-pay').textContent = ""
document.getElementById('electriRateFinal').textContent = ""
document.getElementById('result').style.display = 'none';
document.getElementById('reset-btn').style.display = 'none';
end_loader()
```

Middle:

This part was made for Xiaohui, the tasks were to create a new function to calculate the electricity, allocate each information in the right variable and right *getElementById*, comment what needed to be commented and apply the Unit Testing.
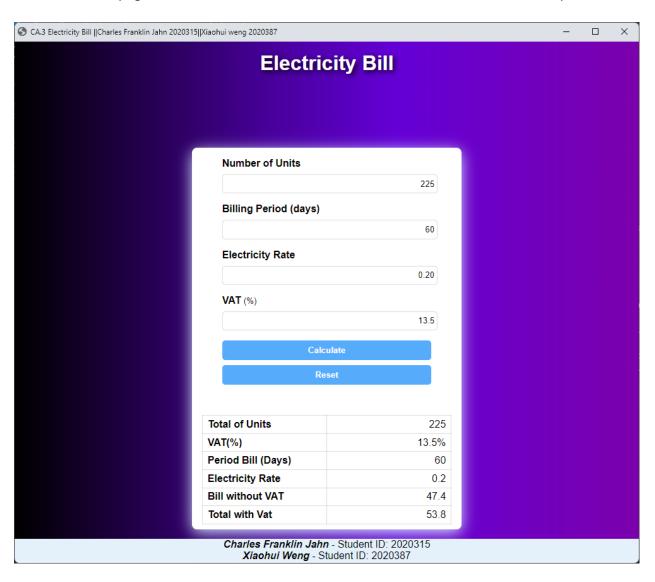
Code screenshots Xiaohui's Task:

Function and Allocation :

```javascript
const totalUnits = document.getElementById('number_Units').value;
const electriRate = document.getElementById('electriRate').value;
const daysOfBill = document.getElementById('days_for_bill').value;
const vatRate = document.getElementById('vat').value;

var total = 0,    //total without vat
    totalWvat= 0;//total with vat

    //base on the CA requirements
    //- Input/ Output for Electricity Bill
    // - Number of Units: 225
    // - Billing Period (Days): 60
    // - Amount of Bill without VAT: 225 * 0.20 + 60 * 0.04 = 47.4
    // - Total payable amount including VAT (13.5%): 47.4 + 47.4 * 13.5/100 = 47.4 + 6.40 = €53.80
    // (Rounded values up to 2 decimal places)
    total = ((parseFloat(totalUnits)) * ((parseFloat(electriRate)))) + ((parseFloat(daysOfBill)) * 0.04);
    totalWvat = (total * (parseFloat(vatRate) / 100)) + total;

setTimeout(() => {
    document.getElementById('total_Of_Units').textContent = parseFloat(totalUnits).toLocaleString("en-US", { style: "decimal", maximumFractionDigits: 2 })
    document.getElementById('VAT').textContent = parseFloat(vatRate).toLocaleString("en-US", { style: "decimal", maximumFractionDigits: 2 }) + "%";
    document.getElementById('bill_period_days').textContent = parseFloat(daysOfBill).toLocaleString("en-US", { style: "decimal", maximumFractionDigits: 2 });
    document.getElementById('TotalwithVat').textContent = parseFloat(totalWvat).toLocaleString("en-US", { style: "decimal", maximumFractionDigits: 2 });
    document.getElementById('total-pay').textContent = parseFloat(total).toLocaleString("en-US", { style: "decimal", maximumFractionDigits: 2 });
    document.getElementById('electriRateFinal').textContent = parseFloat(electriRate).toLocaleString("en-US", { style: "decimal", maximumFractionDigits: 2 });
    document.getElementById('result').style.display = 'table';
    document.getElementById('reset-btn').style.display = 'block';
```

Unit Testing:

```javascript
test > JS script.test.js > ...
  1    'use strict';
  2
  3  ∨ describe(
  4  ∨     'script', () =>{
  5  ∨         describe(
  6  ∨             '#find', () =>{
  7  ∨                 it(
  8                        'should return result when a file matches a term'
  9                     );
 10                 }
 11             );
 12         }
 13     );
```

4

Application

Here is a final screenshot of the application after all changes and the extra implementations.
In the end of the page is able to find the links for the Github and the video with more explanation.

5

Links:

Github:

https://github.com/Charlesjahn/CA3_CrossPlatform

Video:

https://drive.google.com/file/d/1Hg2dwaaoZvqyZaVEQ60tK3Wcl5kZPypJ/view