



Smart Tax Payment Tracking System

A Final Term Project Report

Course Name: Database Systems

Course Code: CSIT555

Section: 04FA23

Instructor: Professor Xiaofeng Li

Team Rolex

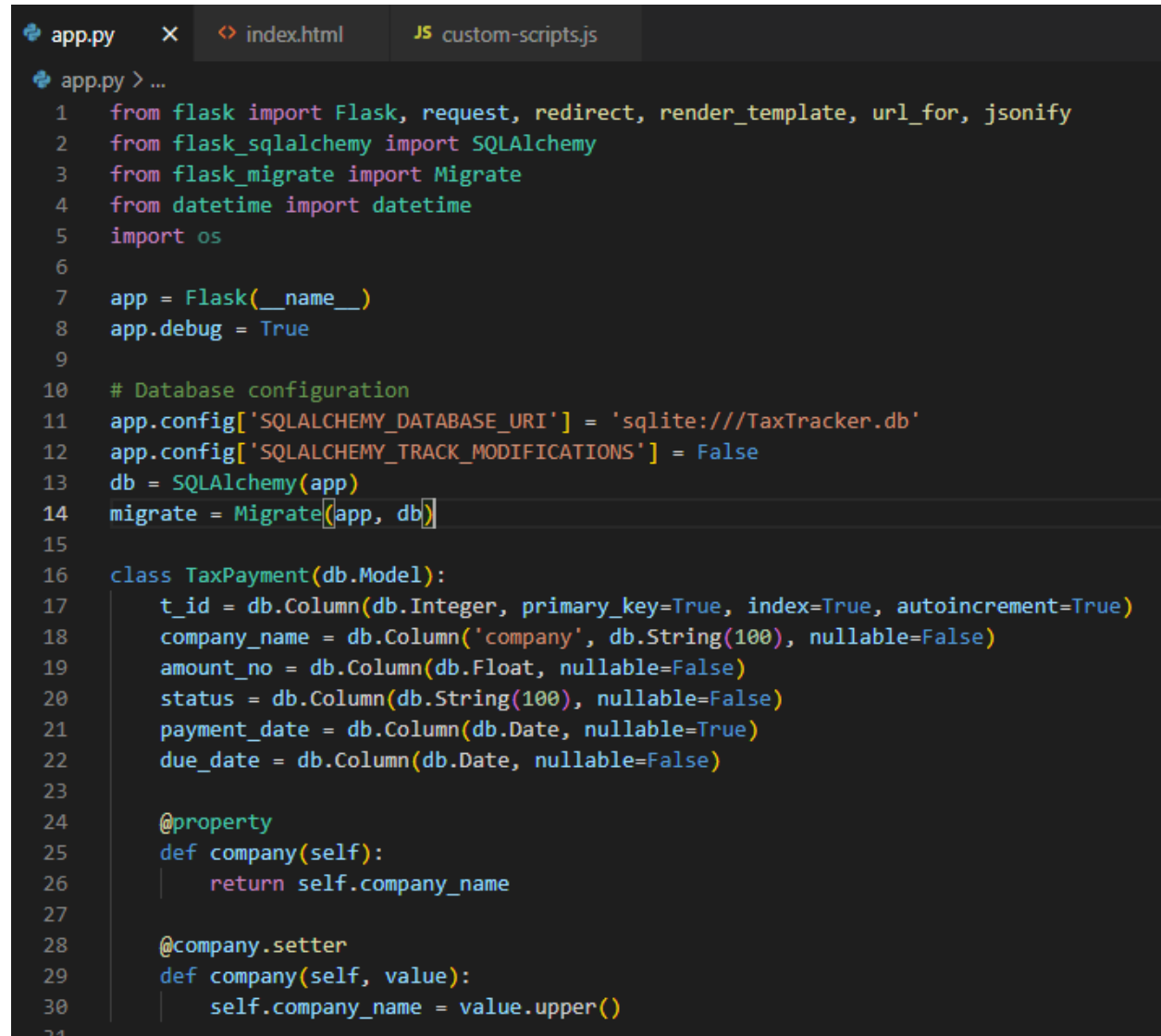
Members

Shibbir Ahmed Arif, Shivaprasad Dubyala,
Udayteja Reddy Pingili and Achyuth Rapolu

December 08, 2023

#Connect Database

In our project, we utilized the Python Flask framework to build an instance called app.py, and then connected to our database using the Sqlite. We have set SQLALCHEMY_DATABASE_URI as 'sqlite:///TaxTracker.db'. A connection is established between the db and the app. We made the db and migration object from the app, after the database is created. And then, we built the TaxPayment model with six columns where t_id is the primary key.



```
app.py  X  index.html  JS custom-scripts.js
app.py > ...
1  from flask import Flask, request, redirect, render_template, url_for, jsonify
2  from flask_sqlalchemy import SQLAlchemy
3  from flask_migrate import Migrate
4  from datetime import datetime
5  import os
6
7  app = Flask(__name__)
8  app.debug = True
9
10 # Database configuration
11 app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///TaxTracker.db'
12 app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False
13 db = SQLAlchemy(app)
14 migrate = Migrate(app, db)
15
16 class TaxPayment(db.Model):
17     t_id = db.Column(db.Integer, primary_key=True, index=True, autoincrement=True)
18     company_name = db.Column('company', db.String(100), nullable=False)
19     amount_no = db.Column(db.Float, nullable=False)
20     status = db.Column(db.String(100), nullable=False)
21     payment_date = db.Column(db.Date, nullable=True)
22     due_date = db.Column(db.Date, nullable=False)
23
24     @property
25     def company(self):
26         return self.company_name
27
28     @company.setter
29     def company(self, value):
30         self.company_name = value.upper()
31
```

After that, we have created a route for showing the home page of the app. And then we created a route to get the tax payment records based on a specific due date.

```
@app.route('/')
def index():
    records = TaxPayment.query.all()
    return render_template('index.html', records=records)

@app.route('/fetchTaxRecords/<dueDate>')
def fetchTaxRecords(dueDate):
    if dueDate == 'all':
        records = TaxPayment.query.all()
    else:
        due_date_obj = datetime.strptime(dueDate, '%Y-%m-%d').date()
        records = TaxPayment.query.filter(TaxPayment.due_date == due_date_obj)

    json_records = []
    for record in records:
        json_record = {
            't_id': record.t_id,
            'company_name': record.company_name,
            'amount_no': record.amount_no,
            'payment_date': record.payment_date.strftime('%Y-%m-%d') if record.payment_date else None,
            'status': record.status,
            'due_date': record.due_date.strftime('%Y-%m-%d')
        }
        json_records.append(json_record)

    return jsonify(json_records)
```

Then we created another route to insert a new record

```
@app.route('/insert-record', methods=['POST'])
def insert_record():
    company = request.form.get('company')
    amount_no_str = request.form.get('amount_no')

    if amount_no_str is not None and amount_no_str.strip():
        try:
            amount_no = float(amount_no_str)
        except ValueError:
            return "Invalid amount provided", 400
    else:
        return "Amount is required", 400

    payment_date_str = request.form.get('payment_date')
    payment_date = datetime.strptime(payment_date_str, '%Y-%m-%d') if payment_date_str else None
    status = request.form.get('status')
    due_date = datetime.strptime(request.form.get('due_date'), '%Y-%m-%d')

    new_record = TaxPayment(company=company, amount_no=amount_no, payment_date=payment_date, status=status, due_date=due_date)
    db.session.add(new_record)
    db.session.commit()
    return redirect(url_for('index'))
```

After that, we created another route to update any existing record. Then we create a route to delete any existing records.

```
@app.route('/update/<int:t_id>', methods=['POST'])
def update_record(t_id):
    record = TaxPayment.query.get_or_404(t_id)
    record.company = request.form.get('company')
    record.amount_no = float(request.form.get('amount_no'))
    record.payment_date = datetime.strptime(request.form.get('payment_date'), '%Y-%m-%d')
    record.status = request.form.get('status')
    record.due_date = datetime.strptime(request.form.get('due_date'), '%Y-%m-%d')
    db.session.commit()
    return redirect(url_for('index'))

@app.route('/delete/<int:t_id>', methods=['POST'])
def delete_record(t_id):
    record = TaxPayment.query.get_or_404(t_id)
    db.session.delete(record)
    db.session.commit()
    return redirect(url_for('index'))

if __name__ == '__main__':
    app.run()
```

API and Controller:

ENDPOINT '/' (only controller)

METHOD: GET

ENDPOINT '/fetch/TaxRecords/<dueDate>'

ENDPOINT '/insert-record'

METHOD: POST

ENDPOINT: '/update/<int:t_id>'

METHOD: POST

ENDPOINT: 'delete/<int:t_id>'

METHOD: POST

Index.html Code:

```
app.py index.html X JS custom-scripts.js
templates > index.html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Smart Tax Payment Tracking System</title>
7   <link rel="stylesheet" href="{{ url_for('static', filename='layout-styles.css') }}">
8   <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">
9
10 </head>
11 <body>
12   <h1>Smart Tax Payment Tracking System</h1>
13
14   <!-- New Record Entry Form -->
15   <form id="taxForm" action="/insert-record" method="post">
16     <h4>Enter A New Record</h4>
17     <label for="companyInput">Company</label>
18     <input type="text" id="companyInput" name="company" required>
19     <label for="amountInput">Amount</label>
20     <input type="number" id="amountInput" name="amount_no" required>
21     <label for="statusInput">Status</label>
22     <select id="statusInput" name="status" required>
23       <option value="paid">Paid</option>
24       <option value="unpaid">Unpaid</option>
25     </select>
26     <label for="paymentDateInput">Payment Date</label>
27     <input type="date" id="paymentDateInput" name="payment_date" placeholder="Payment Date">
28     <label for="due_date">Due Date</label>
29     <select id="dueDate" name="due_date" required>
30     </select>
31     <button type="submit">Save</button>
32   </form>
```

```
app.py index.html X JS custom-scripts.js
templates > index.html
34 <!-- Records Display Table -->
35 <div id="recordsContainer">
36   <h4> Tax Summary</h4>
37   <label>Select Due Date</label>
38
39   <select class="form-control mb-2" id="selectDueDate" name="due_date" onchange="updateDataTable();">
40     <option value="all">All</option>
41   </select>
42
43   <table id="recordsTable">
44     <thead>
45       <tr>
46         <th>Company</th>
47         <th>Amount</th>
48         <th>Status</th>
49         <th>Payment Date</th>
50         <th>Due Date</th>
51         <th>Actions</th>
52       </tr>
53     </thead>
54     <tbody id="recordsTableBody">
55       {% for record in records %}
56       <tr>
57         <td>{{ record.company_name }}</td>
58         <td name="record-amount">{{ record.amount_no }}</td>
59         <td>{{ record.status }}</td>
60         <td>{{ record.payment_date.strftime('%Y-%m-%d') if record.payment_date else 'NA' }}</td>
61         <td>{{ record.due_date.strftime('%Y-%m-%d') }}</td>
62         <td>
63           <button type="button" class="btn btn-primary" onclick="showUpdateModal('{{ record.t_id }}', '{{ record.t_id }}')>Update</button>
64           <button type="button" class="btn btn-danger" onclick="confirmDelete('{{ record.t_id }}')>Delete</button>
65         </td>
```

```

app.py  index.html X  JS custom-scripts.js
templates > index.html
67      <!-- Record Update -->
68      <div class="modal fade" id="updateModal" tabindex="-1" role="dialog" aria-labelledby="updateModalLabel">
69          <div class="modal-dialog" role="document">
70              <div class="modal-content">
71                  <div class="modal-header">
72                      <h5 class="modal-title" id="updateModalLabel">Let's update the record</h5>
73                      <button type="button" class="close" data-dismiss="modal" aria-label="Close">
74                          <span aria-hidden="true">&times;</span>
75                      </button>
76                  </div>
77
78                  <div class="modal-body">
79
80                      <!-- Form for update record -->
81                      <form id="updateForm" action="/update/{{ record.t_id }}" method="post">
82                          <input type="hidden" id="updateIdInput" name="t_id">
83                          <label for="updateCompanyInput">Company</label>
84                          <input type="text" class="form-control mb-2" id="updateCompanyInput" name="company" req
85                          <label for="updateAmountInput">Amount</label>
86                          <input type="number" class="form-control mb-2" id="updateAmountInput" name="amount_no"
87                          <label for="updateStatusInput">Status</label>
88                          <select class="form-control mb-2" id="updateStatusInput" name="status" required>
89                              <option value="paid">Paid</option>
90                              <option value="unpaid">Unpaid</option>
91                          </select>
92                          <label for="updatePaymentDateInput">Payment Date</label>
93                          <input type="date" class="form-control mb-2" id="updatePaymentDateInput" name="payment_
94                          <label for="updateDueDateSelect">Due Date</label>
95                          <select class="form-control mb-2" id="updateDueDateSelect" name="due_date" required>
96                          </select>
97                      </form>
98                  </div>

```

```

app.py  index.html X  JS custom-scripts.js
templates > index.html
99
100      <div class="modal-footer">
101          <button type="button" class="btn btn-secondary" data-dismiss="modal">Close</button>
102          <button type="button" class="btn btn-primary" onclick="submitUpdate()">Save changes</bu
103      </div>
104  </div>
105  </div>
106  </div>
107  </tr>
108  {% endfor %}
109  </tbody>
110  </table>
111
112  <div id="taxContainer">
113      <table id="taxTable">
114          <tr>
115              <td width="80%">Total Amount</td>
116              <td id="totalAmountSpan"></td>
117          </tr>
118          <tr>
119              <td width="80%">Rate of Tax</td>
120              <td><input type="number" class="form-control mb-2" id="taxRateInput" placeholder="0.03" oninput="calcul
121          </tr>
122          <tr>
123              <td width="80%">Tax Due</td>
124              <td id="taxDueSpan"></td>
125          </tr>
126      </table>
127  </div>
128  </div>
129

```

```

127     </div>
128 </div>
129
130 <script src="https://code.jquery.com/jquery-3.3.1.min.js"></script>
131 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
132 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
133 <script src="{{ url_for('static', filename='custom-scripts.js') }}"></script>
134
135 </body>
136 </html>
137

```

layout-styles.css

```

app.py  index.html  JS custom-scripts.js  # layout-styles.css X
static > # layout-styles.css > #taxRateInput
1  body {
2      font-family: 'Open Sans', sans-serif;
3      margin: 0;
4      padding: 0;
5      background-color: #f4f4f4;
6  }
7
8  h1 {
9      text-align: center;
10     color: #2c3e50;
11     font-size: 40px;
12     padding-top: 10px;
13     text-shadow: 2px 2px 4px rgba(0, 0, 0, 0.2);
14 }
15
16 h4 {
17     text-align: center;
18     margin-top: 50px;
19     color: #333;
20 }
21
22 /* Styles for the Form */
23 form {
24     max-width: 500px;
25     margin: 20px auto;
26     padding: 30px;
27     background-color: #green;
28     border-radius: 10px;
29     color: #white;
30     box-shadow: 0 0 15px rgba(0, 0, 0, 0.1);
31 }
32

```

```
app.py index.html JS custom-scripts.js # layout-styles.css X
static > # layout-styles.css > #taxRateInput
30     box-shadow: 0 0 15px rgba(0, 0, 0, 0.1);
31 }
32
33 form input, form select, form button {
34     width: 100%;
35     padding: 15px;
36     margin: 15px 0;
37     border-radius: 8px;
38     border: 1px solid #ddd;
39 }
40
41 form button {
42     background-color: black;
43     color: white;
44     border: none;
45     cursor: pointer;
46     transition: background-color 0.3s ease;
47 }
48
49 form button:hover {
50     background-color: #2980b9;
51 }
52
53 /* Styles for the Table */
54 table {
55     width: 90%;
56     margin: 20px auto;
57     border-collapse: collapse;
58 }
59
```

```
app.py index.html JS custom-scripts.js # layout-styles.css X
static > # layout-styles.css > #taxRateInput
59
60 table, th, td {
61     border: 1px solid #ddd;
62     padding: 15px;
63     text-align: center;
64 }
65
66 th {
67     background-color: black;
68     color: white;
69 }
70
71 table tr:hover {
72     background-color: #e6f7ff;;
73 }
74
75 /* Styles for the Tax Summary*/
76 #recordsContainer {
77     background-color: #fff;
78     border-radius: 8px;
79     box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
80     padding: 20px;
81     width: 90%;
82     margin: 20px auto;
83 }
84
85 #recordsContainer h4, #recordsContainer label {
86     color: #333;
87     margin-bottom: 10px;
88     font-weight: bold;
89 }
90
```



```
app.py index.html JS custom-scripts.js # layout-styles.css X
static > # layout-styles.css > #taxRateInput
91 #SelectDueDate {
92     width: 100%;
93 }
94
95 /* Styles for the Tax Container */
96 #taxContainer {
97     border-radius: 8px;
98     box-shadow: none;
99     padding: 20px;
100     margin-top: 20px;
101     width: 93.5%;
102     margin: 20px auto;
103     font-weight: bold;
104 }
105
106 #taxTable {
107     width: 100%;
108     border-collapse: collapse;
109     margin-top: 20px;
110 }
111
112 #taxTable, #taxTable th, #taxTable td {
113     border: 1px solid #ddd;
114     padding: 15px;
115     text-align: left;
116 }
117
118 #taxTable tr:hover {
119     background-color: #e6f7ff;
120 }
121
122 #taxRateInput {
123     width: 60%;
124     padding: 10px;
125     border: 2px solid #ccc;
126     margin: 0px auto;
127 }
128
129 #totalAmountSpan, #taxDueSpan {
130     font-size: 20px;
131     font-weight: bold;
132 }
133
```

custom-scripts.js

```
app.py  index.html  JS custom-scripts.js X  # layout-styles.css
static > JS custom-scripts.js > ...
1  function showUpdateModal(recordId, company, amount, paymentDate, status, dueDate) {
2      // Fill the modal with existing data
3      $('#updateIdInput').val(recordId);
4      $('#updateCompanyInput').val(company);
5      $('#updateAmountInput').val(amount);
6      $('#updateStatusInput').val(status);
7      $('#updatePaymentDateInput').val(paymentDate);
8      $('#updateDueDateSelect').val(dueDate);
9
10     $('#updateForm').attr('action', '/update/' + recordId);
11     $('#updateModal').modal('show');
12 }
13
14 // Function to submit the updated data
15 function submitUpdate() {
16     var actionUrl = '/update/' + $('#updateIdInput').val();
17     $('#updateForm').attr('action', actionUrl);
18     $('#updateForm').submit();
19 }
20
21 // Function to delete a record
22 function confirmDelete(tId) {
23     if (confirm("Do you want to delete this existing record?")) {
24         var actionUrl = '/delete/' + tId;
25         $('#updateForm').attr('action', actionUrl);
26         $('#updateForm').submit();
27     } else {
28         return false;
29     }
30 }
```

```
app.py  index.html  JS custom-scripts.js X  # layout-styles.css
static > JS custom-scripts.js > ...
31
32 // Function to display the dropdown menu for due date
33 function dueDateDropdownMenu(dropdownmenuID, includeAllOption = false) {
34     var dropdown = document.getElementById(dropdownmenuID);
35     var years = [2023, 2024, 2025];
36
37     // Clear existing options
38     dropdown.innerHTML = '';
39
40     // Add 'All' option to select a due date from dropdown menu
41     if (includeAllOption) {
42         var allOption = document.createElement('option');
43         allOption.value = 'all';
44         allOption.textContent = 'All';
45         dropdown.appendChild(allOption);
46     }
47
48     // Store new options in the dropdown menu
49     years.forEach(function (year) {
50         [0, 3, 5, 8].forEach(function (monthOffset) {
51             var dueDate = new Date(year, monthOffset, 15);
52             var option = document.createElement('option');
53             option.value = dueDate.toISOString().split('T')[0]; // Format as 'YYYY-MM-DD'
54             option.textContent = getFormattedDate(dueDate);
55             dropdown.appendChild(option);
56         });
57     });
58 }
```

```

JS custom-scripts.js X
static > JS custom-scripts.js > ...
60 // Function to format the date as 'MMM DD, YYYY'
61 function getFormattedDate(date) {
62     var options = { month: 'long', day: 'numeric', year: 'numeric' };
63     return date.toLocaleDateString(undefined, options);
64 }
65
66 function updateDataTable() {
67     let dateText = $('#selectDueDate option:selected').val();
68
69     if (dateText === 'all') {
70         $.ajax({
71             type: "GET",
72             url: "/fetchTaxRecords/all",
73             success: function (result) {
74                 updateTable(result);
75             }
76         });
77     } else {
78         $.ajax({
79             type: "GET",
80             url: "/fetchTaxRecords/" + dateText,
81             success: function (result) {
82                 updateTable(result);
83             }
84         });
85     }
86 }

```

```

JS custom-scripts.js X
static > JS custom-scripts.js > ...
88 function updateTable(tableData) {
89     const tableBody = document.getElementById('recordsTableBody');
90     let tableContent = '';
91     tableData.forEach((row) => {
92         tableContent += `<tr>
93             <td>${row.company_name}</td>
94             <td name="record-amount">${row.amount_no}</td>
95             <td>${row.status}</td>
96             <td>${formatDate(row.payment_date)}</td>
97             <td>${formatDate(row.due_date)}</td>
98             <td>
99                 <button type="button" class="btn btn-primary" onclick="showUpdateModal('${row.t_id}', '${row.amount_no}')>Update</button>
100                 <button type="button" class="btn btn-danger" onclick="confirmDelete('${row.t_id}')">Delete</button>
101             </td>
102         </tr>`;
103     });
104     tableBody.innerHTML = tableContent;
105
106     updateTaxTable();
107 }
108
109 function formatDate(dateStr) {
110     if (dateStr !== null && dateStr !== undefined) {
111         return new Date(dateStr).toISOString().split('T')[0]; // Format as 'YYYY-MM-DD'
112     }
113     else {
114         return "N/A"
115     }
116 }
117

```

```

JS custom-scripts.js X
static > JS custom-scripts.js > ...
116 }
117
118 function updateTaxTable() {
119     // Update total amount
120     let elements = document.getElementsByName('record-amount');
121
122     if (elements.length === 0) {
123         document.getElementById("taxContainer").style.display = 'none';
124     } else {
125         let sum = 0;
126         for (const e of elements) {
127             sum += parseFloat(e.textContent);
128         }
129         document.getElementById("totalAmountSpan").innerHTML = "$" + sum;
130         document.getElementById("taxContainer").style.display = 'block';
131     }
132
133     calculateTaxDue();
134 }
135
136 function calculateTaxDue() {
137     let totalAmount = parseFloat(document.getElementById("totalAmountSpan").innerHTML.replace("$", ""));
138     let taxRate = parseFloat(document.getElementById("taxRateInput").value);
139
140     if (!isNaN(totalAmount) && !isNaN(taxRate)) {
141         let taxAmount = totalAmount * taxRate;
142         taxAmount = (Math.round(taxAmount * 100) / 100).toFixed(2); // round to two decimal
143         document.getElementById("taxDueSpan").innerHTML = "$" + taxAmount;
144     }
145 }
146
147 // Event listener for DOMContentLoaded
148 document.addEventListener('DOMContentLoaded', function () {
149     dueDateDropDownMenu('dueDate');
150     dueDateDropDownMenu('selectDueDate', true);
151     dueDateDropDownMenu('updateDueDateSelect');
152
153     updateTaxTable();
154 });

```

Screenshot of Project:

For all the due date:

Smart Tax Payment Tracking System

Enter A New Record

Company

Amount

Status

Paid

Payment Date

mm/dd/yyyy

Due Date

15 January 2023

Save

Tax Summary					
Select Due Date					
All					
Company	Amount	Status	Payment Date	Due Date	Actions
BANK OF AMERICA	2000.0	paid	2023-12-13	2023-06-15	<button>Update</button> <button>Delete</button>
JPMORGAN CHASE	2000.0	unpaid	NA	2023-04-15	<button>Update</button> <button>Delete</button>
LAKELAND BANK	3000.0	paid	2023-03-09	2023-09-15	<button>Update</button> <button>Delete</button>
Total Amount				\$7000	
Rate of Tax				<input type="text" value="0.2"/>	
Tax Due				\$1400.00	

For any specific due date:

Smart Tax Payment Tracking System

Enter A New Record

Company

Amount

Status

Paid

Payment Date

mm/dd/yyyy

Due Date

15 January 2023

Save

Tax Summary

Select Due Date

15 June 2023

Company	Amount	Status	Payment Date	Due Date	Actions
BANK OF AMERICA	2000	paid	2023-12-13	2023-06-15	<button>Update</button> <button>Delete</button>
VALLEY BANK	5000	paid	2023-08-12	2023-06-15	<button>Update</button> <button>Delete</button>

Total Amount	\$7000
Rate of Tax	<input type="text" value="0.3"/>
Tax Due	\$2100.00

GitHub Repository:

<https://github.com/ShibbirAhmedArif/Smart-Tax-Payment-Tracking-System>

YouTube Video Link:

<https://www.youtube.com/watch?v=5kvpKA6kE-U>