<template>

  <div class="page-content">

    <h6 class="mb-0 text-uppercase">Daftar Sumur</h6>

    <div class="card">

      <div class="card-body">

        <div class="table-responsive">

          <table id="example" class="table table-striped table-bordered">

            <thead>

              <tr class="text-center">

                <th>Aktifitas</th>

                <th>Identitas Sumur</th>

              </tr>

            </thead>

            <tbody>

              <tr v-for="(waterhole, index) in data\_waterhole" :key="index">

                <td>

                  <div class="d-block d-md-flex justify-content-between">

                    <div class="d-flex order-actions">

                      <a

                        @click="openModalAdd(waterhole.whid)"

                        class="bg-dark cursor-pointer"

                        ><i class="bx bx-message-alt-edit"></i

                      ></a>

                    </div>

                  </div>

                </td>

                <td>{{ waterhole.whname }}</td>

              </tr>

            </tbody>

          </table>

        </div>

      </div>

    </div>

  </div>

  <!-- Modal Assessment Add -->

  <div

    class="modal fade modal-lg"

    id="modal\_add"

    aria-labelledby="modal\_add\_label"

  >

    <div class="modal-dialog">

      <div class="modal-content">

        <div class="modal-body">

          <div class="card">

            <div class="card-body">

              <div class="row">

                <div class="col-sm-9"><h6>{{ item\_head.whid }}</h6></div>

                <div class="col-sm-9"><h6>{{ item\_head.whname }}</h6></div>

              </div>

            </div>

          </div>

          <div class="card">

            <div class="card-body">

              <div class="row">

                <div class="col-sm-12">

                  <h6 for="month\_period" class="form-label">

                    Periode Penilaian

                  </h6>

                  <div class="position-relative input-icon">

                    <input

                      type="month"

                      class="form-control"

                      id="month\_period"

                      placeholder="Periode Laporan"

                      :max="maxmonth"

                      v-model="item\_head.month"

                      @input="loadQuestionItem"

                    />

                  </div>

                </div>

              </div>

            </div>

          </div>

          <div class="overflow-auto">

            <div class="card radius-10 w-100 mb-1">

              <div class="card-header bg-gradient-moonlit">

                <div class="d-flex align-items-center">

                  <div>

                    <h6 class="mb-1 text-white">Administrasi</h6>

                  </div>

                </div>

              </div>

              <div class="product-list p-1 mb-2">

                <div class="d-flex flex-column gap-1">

                  <div

                    v-for="(question, index) in questionitem"

                    :key="index"

                    class="d-flex align-items-center"

                    @dblclick="openModalAnswer(question)"

                  >

                    <div class="flex-grow-1">

                      <h6 class="m-0 p-0">{{ question.itemname }}</h6>

                    </div>

                    <div class="">

                      <span class="m-0 p-0"

                        ><i class="bx bx-help-circle bx-sm text-warning m-0"></i

                      ></span>

                    </div>

                  </div>

                </div>

              </div>

            </div>

            <div class="card radius-10 w-100 mb-1">

              <div class="card-header bg-gradient-moonlit">

                <div class="d-flex align-items-center">

                  <div>

                    <h6 class="mb-1 text-white">Catatan</h6>

                  </div>

                </div>

              </div>

              <div class="product-list p-1 mb-2">

                <div class="d-flex flex-column gap-1">

                  <div class="d-flex align-items-center">

                    <textarea class="form-control"></textarea>

                  </div>

                </div>

              </div>

            </div>

          </div>

        </div>

        <div class="modal-footer">

          <button

            type="button"

            class="btn btn-primary"

            @click="submitAnswer"

          >

            <i class="bx bx-save"></i> Simpan

          </button>

        </div>

      </div>

    </div>

  </div>

  <!-- Modal Assessment Add -->

  <!-- Modal Assessment Answer -->

  <div

    class="modal fade bg-gray-trans"

    id="modal\_answer"

    aria-labelledby="modal\_answer\_label"

  >

    <div class="modal-dialog">

      <div class="modal-content">

        <div class="modal-header">

          <h6 class="modal-title" id="modal\_answer\_label">Keterangan</h6>

        </div>

        <div class="modal-body">

          <div class="card mb-0">

            <div class="card-body">

              <div class="row">

                <div class="col-sm-12">

                  <strong

                    >Item penilaian "{{ loadquestion.value.itemname }}" memiliki bobot

                    nilai sebesar {{ loadquestion.value.itemvalue }}</strong

                  >

                </div>

              </div>

              <hr />

              <div class="row">

                <div class="d-grid gap-2 col-6 mx-auto">

                  <button

                    type="button"

                    class="btn btn-block btn-success"

                    @click="answerValue(loadquestion.value)"

                  >

                    Ada

                  </button>

                </div>

              </div>

            </div>

          </div>

        </div>

        <div class="modal-footer">

          <button

            type="button"

            class="btn btn-secondary"

            @click="closeModalAnswer"

          >

            Batal

          </button>

        </div>

      </div>

    </div>

  </div>

  <!-- Modal Assessment Answer -->

</template>

<script setup>

  import { ref, onMounted } from "vue";

  import { format } from "date-fns";

  // Reactive variables

*const* item\_head = ref({

    whid: "",

    whname: "",

    month: null,

  });

*const* maxmonth = ref(format(new *Date*(), "yyyy-MM"));

*const* questionitem = ref([]);

  // Define itemlist as a reactive reference

*const* itemlist = ref([]);

  // Modal references

*const* modal\_add = ref(null);

*const* modal\_answer = ref(null);

  // Function to handle modal opening for adding assessment

*const* openModalAdd = (*whid*) *=>* {

    item\_head.value.whid = whid;

    item\_head.value.whname = `Waterhole ${whid.substring(2)}`;

    modal\_add.value.show();

  };

  // Function to close modal for adding assessment

*const* closeModalAdd = () *=>* {

    item\_head.value.month = null;

    modal\_add.value.hide();

  };

  // Function to open modal for answering assessment question

*const* openModalAnswer = (*question*) *=>* {

    loadquestion.value = { ...question };

    modal\_answer.value.show();

  };

  // Function to close modal for answering assessment question

*const* closeModalAnswer = () *=>* {

    modal\_answer.value.hide();

  };

  // Function to handle answering assessment question

*const* answerValue = (*question*) *=>* {

*const* index = itemlist.value.findIndex(*item* *=>* item.id === question.itemid);

    if (index !== -1) {

      // If item with same itemid exists, replace it

      itemlist.value.splice(index, 1, {

        id: question.itemid,

        name: question.itemname,

        value: question.itemvalue,

      });

    } else {

      // Otherwise, add new item

      itemlist.value.push({

        id: question.itemid,

        name: question.itemname,

        value: question.itemvalue,

      });

    }

    closeModalAnswer();

  };

  // Function to submit the answered assessment

*const* submitAnswer = () *=>* {

    console.log('itemlist =', itemlist.value);

    // Reset itemlist if needed

    itemlist.value = [];

  };

  // Function to load questions based on selected month

*const* loadQuestionItem = () *=>* {

    // Simulated data for question items

    questionitem.value = [

      { itemid: "i001", itemname: "Question Item 1", itemvalue: 10 },

      { itemid: "i002", itemname: "Question Item 2", itemvalue: 5 },

      { itemid: "i003", itemname: "Question Item 3", itemvalue: 15 },

    ];

  };

  // Simulated data for waterholes

*const* data\_waterhole = ref([

    { whid: "wh001", whname: "Waterhole 1" },

    { whid: "wh002", whname: "Waterhole 2" },

    { whid: "wh003", whname: "Waterhole 3" },

  ]);

  // Initialize modals on mount

  onMounted(() *=>* {

    modal\_add.value = new bootstrap.Modal(document.getElementById('modal\_add'));

    modal\_answer.value = new bootstrap.Modal(document.getElementById('modal\_answer'));

  });

</script>

<template>

  <div class="page-content">

    <h6 class="mb-0 text-uppercase">Daftar Sumur</h6>

    <div class="card">

      <div class="card-body">

        <div class="table-responsive">

          <table id="example" class="table table-striped table-bordered">

            <thead>

              <tr class="text-center">

                <th>Aktifitas</th>

                <th>Identitas Sumur</th>

              </tr>

            </thead>

            <tbody>

              <tr v-for="(waterhole, index) in data\_waterhole" :key="index">

                <td>

                  <div class="d-block d-md-flex justify-content-between">

                    <div class="d-flex order-actions">

                      <a

                        @click="openModalAdd(waterhole.whid)"

                        class="bg-dark cursor-pointer"

                        ><i class="bx bx-message-alt-edit"></i

                      ></a>

                    </div>

                  </div>

                </td>

                <td>{{ waterhole.whname }}</td>

              </tr>

            </tbody>

          </table>

        </div>

      </div>

    </div>

  </div>

  <!-- Modal Assessment Add -->

  <div

    class="modal fade modal-lg"

    id="modal\_add"

    aria-labelledby="modal\_add\_label"

  >

    <div class="modal-dialog">

      <div class="modal-content">

        <div class="modal-body">

          <div class="card">

            <div class="card-body">

              <div class="row">

                <div class="col-sm-9"><h6>{{ item\_head.whid }}</h6></div>

                <div class="col-sm-9"><h6>{{ item\_head.whname }}</h6></div>

              </div>

            </div>

          </div>

          <div class="card">

            <div class="card-body">

              <div class="row">

                <div class="col-sm-12">

                  <h6 for="month\_period" class="form-label">

                    Periode Penilaian

                  </h6>

                  <div class="position-relative input-icon">

                    <input

                      type="month"

                      class="form-control"

                      id="month\_period"

                      placeholder="Periode Laporan"

                      :max="maxmonth"

                      v-model="item\_head.month"

                      @input="loadQuestionItem"

                    />

                  </div>

                </div>

              </div>

            </div>

          </div>

          <div class="overflow-auto">

            <div class="card radius-10 w-100 mb-1">

              <div class="card-header bg-gradient-moonlit">

                <div class="d-flex align-items-center">

                  <div>

                    <h6 class="mb-1 text-white">Administrasi</h6>

                  </div>

                </div>

              </div>

              <div class="product-list p-1 mb-2">

                <div class="d-flex flex-column gap-1">

                  <div

                    v-for="(question, index) in questionitem"

                    :key="index"

                    class="d-flex align-items-center"

                    @click="openModalAnswer(question)"

                  >

                    <div class="flex-grow-1">

                      <h6 class="m-0 p-0">{{ question.itemname }}</h6>

                    </div>

                    <div class="">

                      <span class="m-0 p-0"

                        ><i class="bx bx-help-circle bx-sm text-warning m-0"></i

                      ></span>

                    </div>

                  </div>

                </div>

              </div>

            </div>

            <div class="card radius-10 w-100 mb-1">

              <div class="card-header bg-gradient-moonlit">

                <div class="d-flex align-items-center">

                  <div>

                    <h6 class="mb-1 text-white">Catatan</h6>

                  </div>

                </div>

              </div>

              <div class="product-list p-1 mb-2">

                <div class="d-flex flex-column gap-1">

                  <div class="d-flex align-items-center">

                    <textarea class="form-control"></textarea>

                  </div>

                </div>

              </div>

            </div>

          </div>

        </div>

        <div class="modal-footer">

          <button type="button" class="btn btn-primary" @click="submitAnswer">

            <i class="bx bx-save"></i> Simpan

          </button>

        </div>

      </div>

    </div>

  </div>

  <!-- Modal Assessment Add -->

  <!-- Modal Assessment Answer -->

  <div

    class="modal fade bg-gray-trans"

    id="modal\_answer"

    aria-labelledby="modal\_answer\_label"

  >

    <div class="modal-dialog">

      <div class="modal-content">

        <div class="modal-header">

          <h6 class="modal-title" id="modal\_answer\_label">Keterangan</h6>

        </div>

        <div class="modal-body">

          <div class="card mb-0">

            <div class="card-body">

              <div class="row">

                <div class="col-sm-12">

                  <strong

                    >Item penilaian "{{ loadquestion.itemname }}" memiliki bobot

                    nilai sebesar {{ loadquestion.itemvalue }}</strong

                  >

                </div>

              </div>

              <hr />

              <div class="row">

                <div class="d-grid gap-2 col-6 mx-auto">

                  <button

                    type="button"

                    class="btn btn-block btn-success"

                    @click="answerValue(loadquestion)"

                  >

                    Ada

                  </button>

                </div>

              </div>

            </div>

          </div>

        </div>

        <div class="modal-footer">

          <button

            type="button"

            class="btn btn-secondary"

            @click="closeModalAnswer"

          >

            Batal

          </button>

        </div>

      </div>

    </div>

  </div>

  <!-- Modal Assessment Answer -->

</template>

<script setup>

  import { ref, onMounted } from "vue";

  import axios from "axios";

  import { format } from "date-fns";

  // Reactive variables

*const* item\_head = ref({

    whid: "",

    whname: "",

    month: null,

  });

*const* maxmonth = ref(format(new *Date*(), "yyyy-MM"));

*const* questionitem = ref([]);

*const* loadquestion = ref({

    itemid: "",

    itemname: "",

    itemvalue: 0,

  });

  // Functions to handle modals and data

*const* openModalAdd = (*whid*) *=>* {

    // Example: Fetch data for waterhole detail based on whid

    item\_head.value.whid = whid;

    item\_head.value.whname = `Waterhole ${whid.substring(2)}`; // Example logic, adjust as needed

    modal\_add.value.show();

  };

*const* closeModalAdd = () *=>* {

    item\_head.value.month = null;

    modal\_add.value.hide();

  };

*const* openModalAnswer = (*question*) *=>* {

    loadquestion.value = { ...question }; // Populate loadquestion with the selected question

    modal\_answer.value.show();

  };

*const* closeModalAnswer = () *=>* {

    modal\_answer.value.hide();

  };

*const* answerValue = (*question*) *=>* {

    itemlist.value.push({

      id: question.itemid,

      name: question.itemname,

      value: question.itemvalue.toString(),

    });

    closeModalAnswer();

  };

*const* submitAnswer = () *=>* {

*const* answerData = {

      item\_head\_id: item\_head.value.whid,

      item\_head\_name: item\_head.value.whname,

      item\_sub: itemlist.value.map((*item*) *=>* ({

        id: item.id,

        name: item.name,

        value: item.value,

      })),

    };

    console.log("Answer Data:", answerData);

    // Here you can proceed with sending this data to your API or further processing

    // Example: axios.post('/api/submitAnswer', answerData);

  };

  // Initialize modal references

*const* modal\_add = ref(null);

*const* modal\_answer = ref(null);

  // Simulated data for waterholes and questions

*const* data\_waterhole = ref([

    { whid: "wh001", whname: "Waterhole 1" },

    { whid: "wh002", whname: "Waterhole 2" },

    { whid: "wh003", whname: "Waterhole 3" },

  ]);

*const* loadQuestionItem = () *=>* {

    // Simulated data for question items

    questionitem.value = [

      { itemid: "i001", itemname: "Question Item 1", itemvalue: 10 },

      { itemid: "i002", itemname: "Question Item 2", itemvalue: 5 },

      { itemid: "i003", itemname: "Question Item 3", itemvalue: 15 },

    ];

  };

  // Lifecycle hook to initialize modals and data

  onMounted(() *=>* {

    modal\_add.value = new bootstrap.Modal("#modal\_add", {

      backdrop: "static",

      keyboard: false,

    });

    modal\_answer.value = new bootstrap.Modal("#modal\_answer", {

      backdrop: "static",

      keyboard: false,

    });

  });

</script>

<style scoped>

  /\* Optional: Add scoped styles here \*/

</style>

<template>

  <div>

    <div>

      <input type="text" v-model="answer\_head.name" />

      <input type="text" v-model="answer\_head.address" />

      <input type="text" v-model="answer\_head.status" />

    </div>

    <div v-for="(item\_sub, index) in sub\_detail" :key="index">

      <div @click="bindValue(item\_sub.item\_id, item\_sub.item\_name, item\_sub.item\_value)">

        {{ item\_sub.item\_name }}

      </div>

      <input type="text" v-model="answer\_value[index].item\_id" />

      <input type="text" v-model="answer\_value[index].item\_value" />

    </div>

    <button type="button" @click="answerValue">Submit</button>

  </div>

</template>

<script setup>

  import { ref } from "vue";

*const* answer\_head = ref({

    name: "",

    address: "",

    status: "",

  });

*const* sub\_detail = ref([

    {

      item\_id: "001",

      item\_name: "Sub Item 1",

      item\_value: 10,

    },

    {

      item\_id: "002",

      item\_name: "Sub Item 2",

      item\_value: 5,

    },

    {

      item\_id: "003",

      item\_name: "Sub Item 3",

      item\_value: 15,

    }

  ]);

*const* answer\_value = ref(sub\_detail.value.map(*item* *=>* ({

    item\_id: '',

    item\_value: ''

  })));

*const* bindValue = (*item\_id*, *item\_name*, *item\_value*) *=>* {

*const* index = sub\_detail.value.findIndex(*item* *=>* item.item\_id === item\_id);

    answer\_value.value[index].item\_id = item\_id;

    answer\_value.value[index].item\_value = item\_value;

  };

*const* answerValue = () *=>* {

*const* sub\_items = sub\_detail.value.map(*item* *=>* ({

      id: item.item\_id,

      name: item.item\_name,

      value: answer\_value.value.find(*av* *=>* av.item\_id === item.item\_id)?.item\_value || ""

    }));

*const* answerData = {

      head\_name: answer\_head.value.name,

      head\_address: answer\_head.value.address,

      head\_status: answer\_head.value.status,

      sub\_item: sub\_items

    };

    console.log("Answer Data:", answerData);

  };

</script>

<template>

  <div>

    <div>

      <input *type*="text" v-model="answer\_head.name" />

      <input *type*="text" v-model="answer\_head.address" />

      <input *type*="text" v-model="answer\_head.status" />

    </div>

    <div v-for="(item\_sub, index) in sub\_detail" :key="index">

      <div @click="bindValue(item\_sub.item\_id, item\_sub.item\_value)">

        {{ item\_sub.item\_name }}

      </div>

      <input *type*="text" v-model="answer\_value.item\_id" />

      <input *type*="text" v-model="answer\_value.item\_value" />

    </div>

    <button *type*="button" @click="answerValue"></button>

  </div>

</template>

<script setup>

  import { ref, onMounted } from "vue";

  import axios from "axios";

    const head\_detail = ref({

      name: "",

      address: "",

      status: "",

    });

    head\_detail.value.name = 'Data Name';

    head\_detail.address = 'Data Address';

    head\_detail.status = 'Active';

    const sub\_detail = ref({

      item\_id: "001",

      item\_name: "Sub Item 1",

      item\_value: 10,

    },

    {

      item\_id: "002",

      item\_name: "Sub Item 2",

      item\_value: 5,

    },

    {

      item\_id: "003",

      item\_name: "Sub Item 3",

      item\_value: 15,

    });

    const subdetail\_Value=ref({

      head\_name : '',

      head\_address : '',

      head\_status : '',

      sub\_item\_id:[],

      sub\_item\_name:[],

      sub\_item\_value:[],

    });

    const bindValue = (item\_id, item\_value) {

      answer\_head.value.name = head\_detail.value.name;

      answer\_head.value.address = head\_detail.value.address;

      answer\_head.value.status = head\_detail.value.status;

      answer\_value.value.item\_id = item\_id;

      answer\_value.value.item\_value = item\_value;

    }

    const answerValue = () => {

      // acomodate this answer value with answer\_head and array from answer\_value here

    };

</script>

no i want to output like this :

Answer Data: {

  head\_name: "John Doe",

  head\_address: "123 Main St",

  head\_status: "Active",

  sub\_item:[

    {

        id: "001",

        name: "Sub Item 1",

        value: "10",

    },

    {

        id: "002",

        name: "Sub Item 2",

        value: "5",

    },

    {

        id: "003",

        name: "Sub Item 3",

        value: "15",

    }

    ]

}