

Administrateurs

- [Paramétrage de la fiche détail](#)
- [Catégorie](#)
- [Sous-couche de carte](#)
- [Liste des API](#)

Paramétrage de la fiche détail

Paramétrer la fiche détail de l'établissement en fonction des catégories et des champs

```
<Strong>Informations générales</Strong>
```

```
{{ name|gogo_text }}
```

```
{{ email|gogo_email(label = "Courriel:") }}
```

```
{{ address|gogo_text(label = "Adresse:") }} {{ originalAddress.addressLocality }}
```

```
Téléphone: {{ telephone }}
```

```
RNE: {{ rne }}
```

```
{{ nom_commune }}
```

```
Mise à jour: {{ date_maj_ligne }}
```

```
<Strong>Autres informations</Strong>
```

```
{% for value in groups["type_etablissement"].children %}
```

```
<div>Type d'établissement: {{ value.option.name }}</div>
```

```
{% endfor %}
```

```
{% for value in groups["nom_circonscription"].children %}
```

```
<div>Circonscription: {{ value.option.name }}</div>
```

```
{% endfor %}
```

```
{% for value in groups["pial"].children %}
```

```
<div>PIAL: {{ value.option.name }}</div>
```

```
{% endfor %}
```

```
{% for value in groups["statut_public_prive"].children %}
```

```
<div>Statut: {{ value.option.name }}</div>
```

```
{% endfor %}
```

```
Nombre d'élèves: {{ nombre_d_eleves }}
```

```
{% if college_de_secteur %} Collège de secteur: {{ college_de_secteur }} {% else%} {% endif %}
```

Informations générales

{{ name|gogo_text }}

{{ email|gogo_email(label = "Courriel:") }}

{{ address|gogo_text(label = "Adresse:") }} {{ originalAddress.addressLocality }}

{{ files|gogo_text(label = "Toile:") }}

Téléphone: {{ telephone }}

RNE: {{ rne }}

{{ nom_commune }}

Mise à jour: {{ date_maj_ligne }}

Autres informations

{% for value in groups["type_etablissement"].children %}

<div>Type d'établissement: {{ value.option.name }}</div>

{% endfor %}

{% for value in groups["nom_circonscription"].children %}

<div>Circonscription: {{ value.option.name }}</div>

{% endfor %}

{% for value in groups["pial"].children %}

<div>PIAL: {{ value.option.name }}</div>

{% endfor %}

{% for value in groups["statut_public_prive"].children %}

<div>Statut: {{ value.option.name }}</div>

{% endfor %}

Nombre d'élèves: {{ nombre_d_eleves }}

{% if college_de_secteur %} Collège de secteur: {{ college_de_secteur }} {% else%} {% endif %}

{% if groups["labellisation"].children %}

<div>Labellisation : {% for value in groups["labellisation"].children %}

{{ value.option.name }}

```

{% endfor %}
</div>
{% else%}

{% endif %}

{% if groups["projet"].children %}

<div> ??? Projets : {% for value in groups["projet"].children %}
  {{ value.option.name }}
  <details> <summary>Les objectifs???? </summary> {{ objectif|gogo_text}} </details> </a>
  <details> <summary>Les objectifs détaillés???? </summary> {{ objectifs_detaillés|gogo_text}} </details>
</a>
  <details> <summary>Plan des actions???? </summary> {{ plan_action|gogo_text}} </details> </a>
  {{ partenaires_du_projet|gogo_text(label = "Partenaires:") }}

{% endfor %}
</div>
{% else%}

{% endif %}

```

Version 3/07/2024 Intégration des données PIX avec %

```

<Strong>Informations générales</Strong>
{{ name|gogo_text }}
{{ email|gogo_email(label = "Courriel:") }}
{{ address|gogo_text(label = "Adresse:") }} {{ originalAddress.addressLocality }}

<a href="{{ files }}" target="_blank">Toile de l'établissement</a> </br>

Téléphone: {{ telephone}}
RNE: {{ rne }}
{{ nom_commune }}

{% if LUNDI_MATIN_DEBUT %}

```

```
{# Horaire de l'établissement #}
```

```
<div>
```

```
<details> <summary><Strong>Horaire de l'établissement????</Strong> </summary>
```

```
<table style="border-collapse: collapse; border: 1px solid black" >
```

```
<tr style="border-collapse: collapse; border: 1px solid black">
```

```
<th style="border-collapse: collapse; border: 1px solid black">Lundi</th>
```

```
<th style="border-collapse: collapse; border: 1px solid black">Mardi</th>
```

```
{% if MERCREDI_MATIN_DEBUT %}
```

```
<th style="border-collapse: collapse; border: 1px solid black">Mercredi</th>
```

```
{% else%} {% endif %}
```

```
□ <th style="border-collapse: collapse; border: 1px solid black">Jeudi</th>
```

```
<th style="border-collapse: collapse; border: 1px solid black">Vendredi</th>
```

```
{% if SAMEDI_MATIN_DEBUT %}
```

```
<th style="border-collapse: collapse; border: 1px solid black">Samedi</th>
```

```
{% else%} {% endif %}
```

```
</tr>
```

```
<tr style="border-collapse: collapse; border: 1px solid black">
```

```
<td style="border-collapse: collapse; border: 1px solid black">{{ LUNDI_MATIN_DEBUT.substring(0, 5) }} -
```

```
{{ LUNDI_MATIN_FIN.substring(0, 5) }} </td>
```

```
<td style="border-collapse: collapse; border: 1px solid black">{{ MARDI_MATIN_DEBUT.substring(0, 5) }} -
```

```
{{ MARDI_MATIN_FIN.substring(0, 5) }} </td>
```

```
{% if MERCREDI_MATIN_DEBUT %}
```

```
<td style="border-collapse: collapse; border: 1px solid black">{{ MERCREDI_MATIN_DEBUT.substring(0, 5) }} -
```

```
{{ MERCREDI_MATIN_FIN.substring(0, 5) }} </td>
```

```
{% else%} {% endif %}
```

```
<td style="border-collapse: collapse; border: 1px solid black">{{ JEUDI_MATIN_DEBUT.substring(0, 5) }} - {{ JEUDI_MATIN_FIN.substring(0, 5) }} </td>
```

```
<td style="border-collapse: collapse; border: 1px solid black">{{ VENDREDI_MATIN_DEBUT.substring(0, 5) }} - {{ VENDREDI_MATIN_FIN.substring(0, 5) }} </td>
```

```
{% if SAMEDI_MATIN_DEBUT %}
```

```
<td style="border-collapse: collapse; border: 1px solid black">{{ SAMEDI_MATIN_DEBUT.substring(0, 5) }} - {{ SAMEDI_MATIN_FIN.substring(0, 5) }} </td>
```

```
{% else%} {% endif %}
```

```
</tr>
```

```
<tr style="border-collapse: collapse; border: 1px solid black">
```

```
<td style="border-collapse: collapse; border: 1px solid black">{{ LUNDI_APRES_MIDI_DEBUT.substring(0, 5) }} - {{ LUNDI_APRES_MIDI_FIN.substring(0, 5) }} </td>
```

```
<td style="border-collapse: collapse; border: 1px solid black">{{ MARDI_APRES_MIDI_DEBUT.substring(0,
```

```

5) }} - {{ MARDI_APRES_MIDI_FIN.substring(0, 5) }} </td>
    {% if MERCREDI_MATIN_DEBUT %}
    <td style="border-collapse: collapse; border: 1px solid black">{{
MERCREDI_APRES_MIDI_DEBUT.substring(0, 5) }} - {{ MERCREDI_APRES_MIDI_FIN.substring(0, 5) }} </td>
    {% else%} {% endif %}
    <td style="border-collapse: collapse; border: 1px solid black">{{ JEUDI_APRES_MIDI_DEBUT.substring(0, 5)
}} - {{ JEUDI_APRES_MIDI_FIN.substring(0, 5) }} </td>
    <td style="border-collapse: collapse; border: 1px solid black">{{
VENDREDI_APRES_MIDI_DEBUT.substring(0, 5) }} - {{ VENDREDI_APRES_MIDI_FIN.substring(0, 5) }} </td>
    {% if SAMEDI_MATIN_DEBUT %}
    <td style="border-collapse: collapse; border: 1px solid black">{{ SAMEDI_APRES_MIDI_DEBUT.substring(0,
5) }} - {{ SAMEDI_APRES_MIDI_FIN.substring(0, 5) }} </td>
    {% else%} {% endif %}
</tr>
</table>

```

```

{% else%} {% endif %}
</details>
</div>

```

Mise à jour: {{ date_maj_ligne }}

Autres informations

```

{% for value in groups["type_etablissement"].children %}
<div>Type d'établissement: {{ value.option.name }}</div>
{% endfor %}

```

```

{% for value in groups["nom_circonscription"].children %}
<div>Circonscription: {{ value.option.name }}</div>
{% endfor %}

```

```

{% for value in groups["pial"].children %}
<div>PIAL: {{ value.option.name }}</div>
{% endfor %}

```

```
{% for value in groups["statut_public_prive"].children %}
```

```
<div>Statut: {{ value.option.name }}</div>
```

```
{% endfor %}
```

```
Nombre d'élèves: {{ nombre_d_eleves }}
```

```
{% if college_de_secteur %} Collège de secteur: {{ college_de_secteur }} {% else%} {% endif %}
```

```
{% if groups["labellisation"].children %}
```

```
<div>Labellisation-Certification : {% for value in groups["labellisation"].children %}
```

```
  {{ value.option.name }}
```

```
{% endfor %}
```

```
</div>
```

```
{% else%}
```

```
{% endif %}
```

```
{% if groups["labellisation"].children %}
```

```
{% if participants %}
```

```
<div> <strong>PIX : </strong>{{ certifies}} élèves certifiés soit {% set reussite =  
(certifies/participants*100)//1 %} {{reussite}} % </br>
```

```
{# <table style="border-collapse: collapse; border: 1px solid black">#}
```

```
{# <tr style=" border: 1px solid black">#}
```

```
  {# <th style=" border: 1px solid black">Participants:</th> #}
```

```
  {# <th style=" border: 1px solid black">Non Certifies:</th> #}
```

```
  {# <th style=" border: 1px solid black">Certifies:</th>#}
```

```
{# </tr>#}
```

```
{# <tr>#}
```

```
  {# <td style=" border: 1px solid black">{{ participants|gogo_text}}</td>#}
```

```
  {# <td style=" border: 1px solid black">{{ non_obtenues|gogo_text}}</td> #}
```

```
  {# <td style=" border: 1px solid black">{{ certifies|gogo_text}}</td>#}
```

```
{# </tr>#}
```

```
{# </table> #}
```

```
</div>
```

```
{% else%} {% endif %}
```

```
{% else%}
```

```
{% endif %}
```

```
{% if groups["projet"].children %}
```

```
<div> ??? Projets : {% for value in groups["projet"].children %}
```

```
  {{ value.option.name }}
```

```
  <details> <summary>Les objectifs???? </summary> {{ objectif|gogo_text}} </details> </a>
```

```
  <details> <summary>Les objectifs détaillés???? </summary> {{ objectifs_detaillés|gogo_text}} </details>
```

```
</a>
```

```
  <details> <summary>Plan des actions???? </summary> {{ plan_action|gogo_text}} </details> </a>
```

```
  {{ partenaires_du_projet|gogo_text(label = "Partenaires:")}}
```

```
{% endfor %}
```

```
</div>
```

```
{% else%}
```

```
{% endif %}
```

Catégorie

PCAA PCAA

Allemand PCAA-ALLEMAND

Sous-Catégories 33

	ABIBAC	147				
	LCE	164				
	LLCECR	157				
	SELO	158				
	SI	SI-ALL				
	LV	LV-ALLEMAND				

Ajouter une Catégorie

Ajouter un groupe

Chinois PCAA-CHINOIS

Sous-Catégories 34

	BFI	149				
	SI	160				
	SELO	SELO-CHINOIS				
	LO	LO				

Ajouter une Catégorie

Ajouter un groupe



Sous-Catégories

34  =  

    BFI	149  =  
    SI	160  =  
    SELO	SELO-CHINOIS  =  
    LO	LO  =  


Ajouter une Catégorie

Ajouter un groupe

✓    Espagnol PCAA-ESPAGNOL  =  

Sous-Catégories

35  =  

    BACHIBAC	151  =  
    LCE	161  =  
    LLCECR	162  =  
    SELO	SELO-ESPAGNOL  =  
    LV	LV-ESPAGNOL  =  

Ajouter une Catégorie

GRUPE PCAA

ESPAGNOL → * Espagnol

CHINOIS → * Chinois

ALLEMAND → * Allemand

GRUPE MODALITE_ENS

SELO-ALL → * Allemand → SELO

SI-CHINOIS → * Chinois → SI

SI-ALL → * Allemand → SI

SELO-ESPAGNOL → * Espagnol → SELO

SELO-CHINOIS → * Chinois → SELO

Abibac → * Allemand → ABIBAC

BFI-CHINOIS → * Chinois → BFI

LV-ALLEMAND → * Allemand → LV

LO → * Chinois → LO

LLCECR-ESPAGNOL → * Espagnol → LLCECR

LLCECR-ALL → * Allemand → LLCECR

LCE-ESPAGNOL → * Espagnol → LCE

LCE-ALL → * Allemand → LCE

Bachibac → * Espagnol → BACHIBAC

LV-ESPAGNOL → * Espagnol → LV

Créer toutes les catégories manquantes

Sous-couche de carte

[SAEI : Les 3 antennes 01-cole_inclusive_2024.geojson](#)

[SAEI : Les 3 antennes 02-ecole_inclusive_couleur.geojson](#)

Communes de la Réunion :

<https://geo.api.gouv.fr/departements/974/communes?format=geojson&geometry=contour>

Liste des API

API dans l'ordre d'importance

1) Liste des établissements - Données locales <https://portail.ac-reunion.fr/swami/public/api/etablisements/>

Importation du fichier vierge avec les têtes de colonne dans GRIST : [Annuaire-ETAB-ACA-ALL.csv](#)

Ce code permet d'ajouter, de corriger les établissements mis à jour.

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="utf-8">
  <title>Import Lycées La Réunion</title>
  <script src="https://docs.getgrist.com/grist-plugin-api.js"></script>
  <style>
    #importEtabBtn {
      margin: 10px;
      padding: 10px 20px;
      background-color: #4CAF50;
      color: white;
      border: none;
      border-radius: 4px;
      cursor: pointer;
      font-family: sans-serif;
      font-size: 14px;
      transition: background-color 0.3s;
    }
    #importEtabBtn:hover:not(:disabled) {
      background-color: #45a049;
    }
    #importEtabBtn:disabled {
      background-color: #cccccc;
      cursor: not-allowed;
    }
  </style>
</head>
```

```
<body>
  <div style="font-family: sans-serif; padding: 1em;">
    <h2>Import des établissements</h2>
    <p>Cliquez sur le bouton ci-dessous pour mettre à jour la table Annuaire_ETAB_ACA_ALL.</p>
    <button id="importEtabBtn">Mettre à jour l'annuaire</button>
    <p id="status" style="margin-top: 10px; color: #666;"></p>
  </div>

  <script>
    // Demande un accès complet pour lire et écrire
    grist.ready({ requiredAccess: 'full' });

    // Liste des noms de colonnes pour la cohérence
    const COLUMN_NAMES = [
      "uaj", "nom", "denominationOfficielle", "uajDenominationOfficielle",
      "type", "rue", "bassin", "code_postal", "ville", "tel", "fax", "mail", "lat", "lon"
    ];

    /**
     * Récupère les données d'établissements via l'API externe.
     */
    async function fetchEtab() {
      const apiUrl = "https://portail.ac-reunion.fr/swami/public/api/etablissements/";
      try {
        document.getElementById("status").textContent = "Récupération des données en cours...";
        const response = await fetch(apiUrl);
        if (!response.ok) {
          throw new Error(`HTTP ${response.status}`);
        }
        const data = await response.json();
        const results = data.data?.values || [];
        document.getElementById("status").textContent = `${results.length} établissements trouvés.`;
        return results;
      } catch (error) {
        document.getElementById("status").textContent = `Erreur : ${error.message}`;
        console.error("Erreur API :", error);
        return [];
      }
    }
  </script>
```

```

/**
 * Met à jour (UpdateRecord) ou insère (AddRecord) les enregistrements.
 */
async function upsertIntoGristOptimized(etabs) {
  document.getElementById("status").textContent = "Préparation des mises à jour en cours...";
  const tableId = "Annuaire_ETAB_ACA_ALL";
  const uajColumn = "uaj";

  // 1. LECTURE : Récupération des enregistrements existants (docApi.fetchTable est fiable)
  const allRecords = await grist.docApi.fetchTable(tableId, ['id', uajColumn]);

  // Création de la Map de recherche rapide
  const uajToldMap = new Map();
  if (allRecords[uajColumn]) {
    allRecords[uajColumn].forEach((uaj, index) => {
      uajToldMap.set(uaj, allRecords.id[index]);
    });
  }

  const actions = []; // Liste des actions à envoyer
  let createdCount = 0;
  let updatedCount = 0;

  for (const etab of etabs) {
    const recordId = uajToldMap.get(etab.uaj);

    // Construction de l'objet d'enregistrement (format clé-valeur)
    const recordObject = {};
    COLUMN_NAMES.forEach(col => {
      const value = etab[col];
      // Assurer que les valeurs manquantes sont correctement traitées
      if (value !== undefined && value !== null) {
        recordObject[col] = value;
      } else if (["rue", "bassin", "fax", "mail"].includes(col)) {
        recordObject[col] = ""; // Les champs texte optionnels doivent être des chaînes vides
      } else {
        recordObject[col] = null; // Autres champs (nombre/liens) sont null
      }
    });
  }
}

```

```

    if (recordId) {
        // MISE À JOUR (Action Ligne par Ligne)
        // Format: ['UpdateRecord', TableID, RowID, RecordObject]
        actions.push(
            ['UpdateRecord', tableId, recordId, recordObject]
        );
        updatedCount++;
    } else {
        // CRÉATION (Action Ligne par Ligne)
        // Format: ['AddRecord', TableID, RowID (null), RecordObject]
        actions.push(
            ['AddRecord', tableId, null, recordObject]
        );
        createdCount++;
    }
}

// 2. ENVOI DES ACTIONS
if (actions.length > 0) {
    document.getElementById("status").textContent = `Envoi de ${actions.length} opérations à Grist...
(peut prendre du temps)`;
    // Envoi de toutes les actions (créations + mises à jour ligne par ligne)
    await grist.docApi.applyUserActions(actions);
}

return { created: createdCount, updated: updatedCount };
}

/**
 * Fonction principale déclenchée par le bouton.
 */
async function importEtab() {
    const btn = document.getElementById("importEtabBtn");
    btn.disabled = true;
    btn.textContent = "Importation en cours...";

    try {
        const etabs = await fetchEtab();
        if (etabs.length > 0) {
            document.getElementById("status").textContent = "Mise à jour de la base en cours...";

```

```
const result = await upsertIntoGristOptimized(etabs);
document.getElementById("status").textContent =
  `❑ **${result.created}** établissements ajoutés, **${result.updated}** établissements mis à
jour.`;
} else {
  document.getElementById("status").textContent = "Aucun établissement trouvé à traiter.";
}
} catch (error) {
  document.getElementById("status").textContent = "❑ Une erreur critique est survenue. Vérifiez la
console (F12) et l'accès.";
  console.error("Erreur critique d'importation :", error);
} finally {
  btn.disabled = false;
  btn.textContent = "Mettre à jour l'annuaire";
}
}

document.getElementById("importEtabBtn").addEventListener("click", importEtab);
</script>
</body>
</html>
```