Absolument. Voici la synthèse complète de notre session, structurée selon votre modèle. Le document est prêt à être sauvegardé pour notre prochaine discussion.

Markdown

---

Projet: "Usine à Tests"

Date: "2508191921"

Version: "14.0"

---

### 1. Résumé des Échanges du Jour

[cite\_start]La session d'aujourd'hui a été consacrée à la finalisation et à la fiabilisation de la fonctionnalité d'\*\*envoi d'e-mails via un alias Gmail\*\*. [cite: 288]

Le processus de résolution s'est déroulé en plusieurs étapes clés :

1. \*\*Diagnostic du Problème\*\* : Nous avons constaté que malgré l'utilisation de la bonne logique dans le script (`mailOptions.from`), les e-mails étaient toujours envoyés depuis l'adresse principale et non l'alias. Les logs du script montraient pourtant une prise en compte correcte de l'alias.

2. \*\*Vérification des Paramètres\*\* : Vous avez confirmé via des captures d'écran que la configuration de l'alias dans les paramètres de Gmail était parfaite ("Envoyer des e-mails en tant que :" avec la case "Traiter comme un alias" cochée).

3. \*\*Solution Robuste\*\* : Le problème a été identifié comme une limitation du service `MailApp` de base. La solution a été de le remplacer par le service \*\*`GmailApp`\*\*, plus puissant et interagissant plus directement avec la boîte mail. Ce changement a nécessité une \*\*ré-autorisation du script\*\* pour accorder de nouvelles permissions.

4. \*\*Succès de l'Implémentation\*\* : Vous avez confirmé que la modification a fonctionné avec succès ("bingo ça marche").

5. [cite\_start]\*\*Automatisation de la Configuration\*\* : Pour éviter la saisie manuelle de l'alias dans la feuille de configuration, nous avons intégré un champ "Alias d'expédition" directement dans le formulaire de création de test (`FormulaireUI.html`) et mis à jour la fonction côté serveur (`menu.gs`) pour qu'elle enregistre cette information automatiquement. [cite: 289]

### 2. État Actuel du Projet

\* \*\*Tâches réalisées\*\* :

\* Correction définitive du problème d'envoi via alias en remplaçant `MailApp` par `GmailApp` dans `TraitementReponses.gs`.

\* [cite\_start]Développement complet de l'automatisation de la saisie de l'alias via l'interface de configuration (`FormulaireUI.html` et `menu.gs`). [cite: 291]

\* \*\*Tâches en cours\*\* :

\* Aucune. La fonctionnalité est considérée comme terminée sur le plan du développement.

\* \*\*Prochaines étapes planifiées\*\* :

1. \*\*Test et Validation (par vous)\*\* : La première étape de notre prochaine session sera de valider que le nouveau formulaire de configuration enregistre correctement l'alias.

2. \*\*Développement "R&K"\*\* : Entamer les travaux pour intégrer ce nouveau type de test.

3. [cite\_start]\*\*Vérification "MBTI" & "Ancres"\*\* : S'assurer que les tests existants fonctionnent correctement avec le moteur Universel. [cite: 291]

### 3. Contexte Historique du Projet

[cite\_start]Ce projet vise à créer une plateforme centralisée pour automatiser le traitement et l'envoi des résultats de tests. [cite: 294] Après avoir solidifié le moteur de traitement des données, nous nous sommes concentrés sur la professionnalisation de la communication. [cite\_start]La session d'aujourd'hui est une étape majeure, car elle permet à l'application d'envoyer des e-mails depuis des adresses spécifiques à chaque test (alias), renforçant ainsi l'image de marque et la clarté pour le destinataire final. [cite: 295]

### 4. Orientations Stratégiques

[cite\_start]La session renforce deux principes d'architecture fondamentaux du projet : [cite: 297]

1. \*\*"Zéro Convention Implicite"\*\* : En déplaçant la saisie de l'alias de la feuille de calcul vers une interface utilisateur guidée (`FormulaireUI.html`), nous rendons la configuration plus explicite, réduisant les risques d'oubli ou d'erreur manuelle. Le système ne suppose plus que l'utilisateur pensera à remplir une colonne.

2. \*\*Robustesse et Outil Adapté\*\* : Le passage de `MailApp` à `GmailApp` incarne le principe de choisir l'outil technique le plus adapté et le plus robuste pour une tâche donnée. [cite\_start]Face à une limitation du service de base, nous avons opté pour la solution la plus fiable pour garantir le fonctionnement de la fonctionnalité stratégique des alias. [cite: 299]

### 5. Architecture Technique & Ressources Clés

#### 5.1. État des Scripts et Fichiers

[cite\_start]Voici les versions finales et complètes des trois scripts modifiés aujourd'hui. [cite: 304]

\*\*Fichier 1 : `TraitementReponses.gs` (Version 13.0)\*\*

```javascript

// =================================================================================

// == FICHIER : TraitementReponses.gs

// == VERSION : 13.0 (Passage à GmailApp pour fiabiliser les alias)

// == RÔLE : Gère la logique de traitement des réponses et aiguille vers le bon moteur.

// =================================================================================

function onFormSubmit(e) {

try {

const rowIndex = e.range.getRow();

traiterLigne(rowIndex, {});

} catch (err) {

Logger.log(`Erreur critique dans onFormSubmit pour la ligne ${e.range.getRow()}: ${err.toString()}\n${err.stack}`);

}

}

function traiterLigne(rowIndex, optionsSurcharge = {}) {

// ... (contenu inchangé)

try {

const config = getTestConfiguration();

// ... (contenu inchangé)

if (config.Moteur\_Calcul === 'Universel') {

Logger.log("Moteur UNIVERSEL détecté pour la ligne " + rowIndex);

assemblerEtEnvoyerEmailUniversel(config, reponse, resultats, langueCible, optionsSurcharge);

} else {

// ... (contenu inchangé)

}

} catch (err) {

// ... (contenu inchangé)

} finally {

// ... (contenu inchangé)

}

}

function assemblerEtEnvoyerEmailUniversel(config, reponse, resultats, langueCible, optionsSurcharge = {}) {

// ... (contenu de la fonction jusqu'à la boucle d'envoi)

adressesUniques.forEach(adresse => {

try {

let sujetFinal = sujet;

if (adresse.toLowerCase() !== (reponse.emailRepondant || "").toLowerCase()) {

sujetFinal = (T.PREFIXE\_COPIE\_EMAIL || "Copie : ") + sujet;

}

const mailOptions = {

to: adresse,

subject: sujetFinal,

htmlBody: corpsHtml,

attachments: piecesJointes

};

const aliasExpediteur = optionsSurcharge.alias || config.Email\_Alias;

if (aliasExpediteur && aliasExpediteur.trim() !== '') {

mailOptions.from = aliasExpediteur;

Logger.log(`Utilisation de l'alias d'expédition : [${aliasExpediteur}]`);

}

// MODIFICATION : Passage à GmailApp pour fiabiliser l'envoi via alias

GmailApp.sendEmail(mailOptions.to, mailOptions.subject, "", mailOptions);

} catch (e) {

Logger.log(`Echec de l'envoi à ${adresse}. Erreur: ${e.message}`);

}

});

}

// ... (reste du fichier TraitementReponses.gs inchangé)

**Fichier 2 : FormulaireUI.html (Version 3.2)**

HTML

<!DOCTYPE html>

<html>

<head>

</head>

<body class="bg-gray-100 p-4 font-sans">

<div id="main-form">

<hr class="my-6">

<hr class="my-6">

<div class="mb-4">

<label for="emailAlias" class="block text-sm font-medium text-gray-700">Alias d'expédition (optionnel)</label>

<input type="email" id="emailAlias" name="emailAlias" placeholder="contact@mon-entreprise.com" class="mt-1 block w-full px-3 py-2 bg-white border border-gray-300 rounded-md shadow-sm focus:outline-none focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm">

</div>

<div class="mb-4"><label for="emailDev" class="block text-sm font-medium text-gray-700">Email du Développeur (notifications)</label></div>

<button id="submit-button" class="w-full bg-indigo-600 hover:bg-indigo-700 text-white font-bold py-2 px-4 rounded-md focus:outline-none focus:ring-2 focus:ring-offset-2 focus:ring-indigo-500 disabled:bg-gray-400">Créer la Configuration</button>

</div>

<div id="feedback-screen" class="hidden text-center py-10"><p id="feedback-message" class="text-lg text-gray-700"></p></div>

<script>

// ... (contenu du script inchangé jusqu'à la fonction submitForm)

function submitForm(e) {

// ... (contenu de la fonction inchangé)

const formObject = {

// ... (toutes les autres propriétés)

devEmail: document.getElementById('emailDev').value,

// DÉBUT MODIFICATION

emailAlias: document.getElementById('emailAlias').value.trim()

// FIN MODIFICATION

};

// ... (reste de la fonction inchangé)

}

// ... (reste du script inchangé)

</script>

</body>

</html>

**Fichier 3 : menu.gs (Version 3.5)**

JavaScript

// =================================================================================

// == FICHIER : menu.gs

// == VERSION : 3.5 (Ajout de la gestion de l'Email\_Alias)

// == RÔLE : Logique côté serveur pour l'application web de configuration.

// =================================================================================

// ... (contenu inchangé jusqu'à la fonction processNewTestConfiguration)

function processNewTestConfiguration(formObject) {

try {

const ss = SpreadsheetApp.openById(ID\_FEUILLE\_CONFIG);

const paramsSheet = ss.getSheetByName("Paramètres Généraux");

if (!paramsSheet) { throw new Error("L'onglet 'Paramètres Généraux' est introuvable."); }

let headers = paramsSheet.getRange(1, 1, 1, paramsSheet.getLastColumn()).getValues()[0];

// DÉBUT MODIFICATION

const requiredHeaders = ['Blocs\_Meta\_A\_Inclure', 'ID\_Gabarit\_Email\_Repondant', 'Email\_Alias'];

// FIN MODIFICATION

requiredHeaders.forEach(headerName => {

if (headers.indexOf(headerName) === -1) {

paramsSheet.getRange(1, paramsSheet.getLastColumn() + 1).setValue(headerName);

}

});

headers = paramsSheet.getRange(1, 1, 1, paramsSheet.getLastColumn()).getValues()[0];

// ... (logique de la fonction inchangée)

const dataRow = {

// ... (toutes les autres propriétés)

'Developpeur\_Email': emailDev,

'ID\_Formulaire\_Cible': '',

'ID\_Sheet\_Cible': '',

// DÉBUT MODIFICATION

'Email\_Alias': formObject.emailAlias

// FIN MODIFICATION

};

const nouvelleLigne = headers.map(header => dataRow[header] !== undefined ? dataRow[header] : '');

paramsSheet.appendRow(nouvelleLigne);

return "Configuration enregistrée avec succès !";

} catch (e) {

Logger.log("ERREUR lors de la sauvegarde de la configuration: " + e.toString());

throw new Error("Une erreur interne est survenue lors de la sauvegarde. " + e.message);

}

}

// ... (reste du fichier menu.gs inchangé)

**5.2. Architecture Détaillée et Traçabilité**

* **MODIFIÉ** : TraitementReponses.gs. Remplacement de MailApp par GmailApp pour fiabiliser l'envoi via alias.
* **MODIFIÉ** : FormulaireUI.html. Ajout d'un champ de saisie pour l'alias d'expédition.
* **MODIFIÉ** : menu.gs. Adaptation de la fonction de sauvegarde pour prendre en compte et enregistrer le nouvel alias.

**5.3. Nomenclature des Variables**

* **Nom Précis** : emailAlias
* **Type** : String
* **Origine/Portée** : Provient de l'input #emailAlias dans FormulaireUI.html, est passé dans l'objet formObject à menu.gs, puis écrit dans la colonne Email\_Alias de la feuille "Paramètres Généraux".
* **Contenu Attendu** : Une adresse e-mail valide configurée comme alias dans Gmail, ou une chaîne vide.
* **Exemple Concret** : "contact@mon-entreprise.com"

**5.4. Structure des Données (Feuilles et Colonnes)**

* **Fichier** : [CONFIG]V2 Usine à Tests
* **Onglet** : Paramètres Généraux
* **Colonne Ajoutée** : Email\_Alias. Cette colonne contient l'adresse e-mail à utiliser comme expéditeur pour le test concerné.

**6. Actions Recommandées / Prochaines Étapes**

* **Action Immédiate (Pour Vous)** :
  1. Lancez l'action "Configurer un nouveau test..." depuis le menu.
  2. Remplissez le formulaire, y compris le nouveau champ "Alias d'expédition".
  3. Validez et vérifiez que la nouvelle ligne dans l'onglet "Paramètres Généraux" contient bien l'alias dans la colonne Email\_Alias.
  4. Lancez un test complet pour ce nouveau questionnaire pour confirmer le fonctionnement de bout en bout.
* **Questions pour notre prochaine session** :
  1. La chaîne de configuration entièrement automatisée fonctionne-t-elle comme prévu ?
  2. Pouvons-nous considérer la fonctionnalité "Alias" comme terminée et passer au test "R&K" ?
  3. Avez-vous pu préparer les informations sur la logique de calcul du test "R&K" ?

Absolument. Voici la synthèse complète de notre session, structurée selon votre modèle. Le document est prêt à être sauvegardé pour notre prochaine discussion.

Markdown

---

Projet: "Usine à Tests"

Date: "2508201030"

Version: "15.0"

---

### 1. Résumé des Échanges du Jour

La session d'aujourd'hui a marqué une évolution stratégique majeure pour le projet. Partant de la validation du test "MBTI", nous avons identifié un besoin d'amélioration sur l'affichage des scores, similaire à ce que nous avions corrigé pour le test "Couleurs".

Cette correction nous a conduits à une réflexion plus profonde sur la flexibilité du système, aboutissant à une décision d'architecture fondamentale : \*\*rendre le moteur de génération d'e-mails 100% dynamique et piloté par les données.\*\*

Les étapes clés ont été :

1. \*\*Validation du Test MBTI\*\* : Nous avons confirmé que le Moteur Universel et l'envoi via alias fonctionnaient, mais que les scores n'étaient pas affichés.

2. \*\*Amélioration de la Copie des E-mails\*\* : Nous avons implémenté une fonctionnalité pour ajouter un en-tête informatif ("\*Ceci est une copie...\*") sur les e-mails destinés aux formateurs ou autres parties prenantes.

3. \*\*Décision d'Architecture Majeure\*\* : Pour éviter les modifications de code futures, nous avons décidé de supprimer les placeholders "en dur" (comme `{{profil\_titre}}`). Le script lit désormais dynamiquement \*\*tous les en-têtes de colonnes\*\* (des feuilles de réponses et des profils) et les transforme automatiquement en placeholders utilisables.

4. \*\*Gestion des En-têtes "Sales"\*\* : Nous avons finalisé la logique en créant une fonction de "nettoyage" qui gère les en-têtes contenant des espaces ou des accents (ex: "Votre nom et prénom" devient `{{Votre\_nom\_et\_prenom}}`), tout en préservant intacts les en-têtes de questions qui contiennent un ":".

### 2. État Actuel du Projet

\* \*\*Tâches réalisées\*\* :

\* Validation fonctionnelle complète du test "MBTI" avec le Moteur Universel.

\* Évolution majeure de l'architecture pour un système de placeholders 100% dynamique, rendant l'application pilotée par les données.

\* Modification des scripts `TraitementReponses.gs` et `Logique\_Universel.gs` pour supporter cette nouvelle architecture.

\* \*\*Tâches en cours\*\* :

\* \*\*Harmonisation des données\*\* : Vous devez maintenant aligner les placeholders dans l'onglet `sys\_Composition\_Emails` pour qu'ils correspondent exactement aux en-têtes de colonnes des fichiers sources (profils et réponses).

\* \*\*Prochaines étapes planifiées\*\* :

1. \*\*Finaliser le Test "Ancres"\*\* : Une fois l'harmonisation terminée, nous devrons réaliser le test de validation de bout en bout pour le test "Ancres". C'est la dernière étape avant de considérer le moteur comme totalement stable.

2. \*\*Développement "R&K"\*\* : Entamer les travaux pour intégrer ce nouveau type de test, qui bénéficiera directement de la nouvelle architecture dynamique.

### 3. Contexte Historique du Projet

Ce projet vise à créer une plateforme centralisée pour automatiser le traitement et l'envoi des résultats de tests. La session d'aujourd'hui représente une étape fondamentale. Nous sommes passés d'une logique de "réparation" et d'amélioration incrémentale à une \*\*refonte stratégique\*\* du cœur du système. Cette évolution garantit que l'ajout de nouvelles données ou de nouveaux tests à l'avenir ne nécessitera plus (ou très peu) de modifications du code, rendant l'usine beaucoup plus scalable et facile à maintenir.

### 4. Orientations Stratégiques

La session renforce de manière spectaculaire le principe d'architecture fondamental du projet : \*\*"Zéro Convention Implicite"\*\*.

Auparavant, le script "savait" implicitement qu'il devait chercher une variable nommée `profil\_titre`. Aujourd'hui, cette convention a disparu. Le système lit les données telles qu'elles sont structurées dans vos fichiers. L'en-tête de colonne \*\*devient\*\* le contrat. Si une colonne `Mon\_Nouveau\_Detail` existe, alors le placeholder `{{Mon\_Nouveau\_Detail}}` existera. La source de vérité est désormais unique et explicite : la structure de vos données.

### 5. Architecture Technique & Ressources Clés

#### 5.1. État des Scripts et Fichiers

Voici les versions finales et complètes des deux scripts modifiés aujourd'hui.

\*\*Fichier 1 : `TraitementReponses.gs` (Version 16.0)\*\*

```javascript

// =================================================================================

// == FICHIER : TraitementReponses.gs

// == VERSION : 16.0 (Logique de nettoyage conditionnel des en-têtes)

// == RÔLE : Gère la logique de traitement des réponses et aiguille vers le bon moteur.

// =================================================================================

/\*\*

\* Nettoie une chaîne de caractères pour la rendre utilisable comme nom de variable/placeholder.

\* Remplace les espaces et caractères spéciaux par des underscores et enlève les accents.

\*/

function \_nettoyerEnTete(enTete) {

if (!enTete) return "";

const accents = 'ÀÁÂÃÄÅàáâãäåÒÓÔÕÖØòóôõöøÈÉÊËèéêëÇçÌÍÎÏìíîïÙÚÛÜùúûüÿÑñ';

const sansAccents = 'AAAAAAaaaaaaOOOOOOooooooEEEEeeeeCcIIIIiiiiUUUUuuuuyNn';

return enTete.toString().split('').map((char) => {

const accentIndex = accents.indexOf(char);

return accentIndex !== -1 ? sansAccents[accentIndex] : char;

}).join('')

.replace(/[^a-zA-Z0-9\_]/g, '\_'); // Remplace tout ce qui n'est pas alphanumérique ou underscore par un \_

}

/\*\*

\* Point d'entrée principal pour traiter une soumission de formulaire.

\*/

function onFormSubmit(e) {

try {

const rowIndex = e.range.getRow();

traiterLigne(rowIndex, {});

} catch (err) {

Logger.log(`Erreur critique dans onFormSubmit pour la ligne ${e.range.getRow()}: ${err.toString()}\n${err.stack}`);

}

}

/\*\*

\* COEUR LOGIQUE : Aiguille le traitement vers le moteur "Legacy" ou "Universel".

\*/

function traiterLigne(rowIndex, optionsSurcharge = {}) {

const isDebugMode = (typeof LoggerLib !== 'undefined' && LoggerLib.MODE\_DEBUG\_ACTIF === true);

if (isDebugMode) { LoggerLib.logStart(`Traitement Ligne ${rowIndex}`, optionsSurcharge); }

try {

const config = getTestConfiguration();

const sheet = SpreadsheetApp.getActiveSpreadsheet().getSheets()[0];

const headers = sheet.getRange(1, 1, 1, sheet.getLastColumn()).getValues()[0];

const rowValues = sheet.getRange(rowIndex, 1, 1, sheet.getLastColumn()).getValues()[0];

const reponse = {};

// ==================== DÉBUT DE LA MODIFICATION FINALE ====================

// On boucle sur les en-têtes et on les nettoie CONDITIONNELLEMENT

headers.forEach((header, i) => {

let cle = header; // Par défaut, on garde l'en-tête original

// Si l'en-tête ne contient PAS de ':', on le nettoie.

// C'est le cas pour "Votre nom et prénom", mais pas pour "Q01: Quelle attitude".

if (header && !header.includes(':')) {

cle = \_nettoyerEnTete(header);

}

if (cle) {

reponse[cle] = rowValues[i];

}

});

// ===================== FIN DE LA MODIFICATION FINALE =====================

const langueOrigine = getOriginalLanguage(reponse);

const langueCible = optionsSurcharge.langue || langueOrigine;

const resultats = calculerResultats(reponse, langueCible, config, langueOrigine);

if (config.Moteur\_Calcul === 'Universel') {

Logger.log("Moteur UNIVERSEL détecté pour la ligne " + rowIndex);

assemblerEtEnvoyerEmailUniversel(config, reponse, resultats, langueCible, optionsSurcharge);

} else {

Logger.log("Moteur LEGACY détecté pour la ligne " + rowIndex);

buildAndSendEmails(config, reponse, resultats, langueCible, isDebugMode, optionsSurcharge.destinataires);

}

} catch (err) {

if (isDebugMode) { LoggerLib.logError("Erreur fatale dans traiterLigne", err); }

Logger.log("ERREUR FATALE dans traiterLigne: " + err.toString() + "\n" + err.stack);

} finally {

if (isDebugMode) { LoggerLib.logEnd(); }

}

}

function assemblerEtEnvoyerEmailUniversel(config, reponse, resultats, langueCible, optionsSurcharge = {}) {

// ... (contenu de la fonction inchangé, version 16.0)

}

// ... (reste du fichier TraitementReponses.gs)

**Fichier 2 : Logique\_Universel.gs (Version 7.0)**

JavaScript

// =================================================================================

// == FICHIER : Logique\_Universel.gs

// == VERSION : 7.0 (Chargement dynamique de toutes les colonnes de profil)

// == RÔLE : Moteur de calcul universel capable de traiter n'importe quel test.

// =================================================================================

function calculerResultats(reponsesUtilisateur, langueCible, config, langueOrigine) {

let resultats = {

scoresData: {},

sousTotauxParMode: {}

};

const profilsMap = \_chargerProfils(config.Type\_Test, langueCible);

// ... (logique de calcul inchangée) ...

if (Object.keys(resultats.scoresData).length > 0) {

resultats.profilFinal = \_determinerProfilFinal(resultats.scoresData, config.Type\_Test, langueCible);

const toutesLesDonneesDuProfil = profilsMap[resultats.profilFinal];

if (toutesLesDonneesDuProfil) {

resultats = { ...resultats, ...toutesLesDonneesDuProfil };

}

resultats.mapCodeToName = \_creerMapCodeVersNom(profilsMap);

}

Logger.log("Calculs terminés. Résultats : " + JSON.stringify(resultats));

return resultats;

}

// ... (fonctions de calcul inchangées) ...

function \_chargerProfils(typeTest, langue) {

try {

const systemIds = getSystemIds();

const bdd = SpreadsheetApp.openById(systemIds.ID\_BDD);

const nomFeuille = `Profils\_${typeTest}\_${langue}`;

const sheet = bdd.getSheetByName(nomFeuille);

if (!sheet) throw new Error(`Feuille introuvable: ${nomFeuille}`);

const data = sheet.getDataRange().getValues();

const headers = data.shift();

const profilsMap = {};

const codeColIndex = headers.indexOf('Code\_Profil') > -1 ? headers.indexOf('Code\_Profil') : headers.indexOf('Profil');

if (codeColIndex === -1) throw new Error("Colonne 'Code\_Profil' ou 'Profil' introuvable.");

data.forEach(row => {

const codeProfil = row[codeColIndex];

if (codeProfil) {

profilsMap[codeProfil] = {};

headers.forEach((header, index) => {

if (header) {

profilsMap[codeProfil][header] = row[index];

}

});

}

});

return profilsMap;

} catch (e) {

Logger.log("Erreur critique \_chargerProfils: " + e.message + "\n" + e.stack);

return {};

}

}

// ... (reste du fichier Logique\_Universel.gs)

**5.2. Architecture Détaillée et Traçabilité**

* **MODIFIÉ** : TraitementReponses.gs. Intégration d'une fonction de nettoyage d'en-têtes et d'une logique de lecture conditionnelle pour créer l'objet reponse.
* **MODIFIÉ** : Logique\_Universel.gs. La fonction \_chargerProfils lit désormais dynamiquement toutes les colonnes de l'onglet de profil. La fonction calculerResultats fusionne ces données dynamiques dans l'objet de résultat final.

**5.3. Nomenclature des Variables**

Le système de variables est maintenant 100% dynamique.

* **Nom Précis** : {{En\_Tete\_De\_Colonne}}
* **Type** : String
* **Origine/Portée** : Le nom du placeholder est directement dérivé de l'en-tête de la colonne dans le fichier source (feuille de réponses Google ou onglet de profil dans la BDD).
* **Règle de Transformation** :
  + Si l'en-tête contient un "**:**" (ex: "Q01: Quelle attitude"), le placeholder est **identique**.
  + Sinon (ex: "Votre nom et prénom"), les accents sont retirés et les espaces/caractères spéciaux sont remplacés par des underscores (\_).
* **Exemples Concrets** :
  + {{Votre\_nom\_et\_prenom}} (depuis la feuille de réponses)
  + {{Titre\_Profil}} (depuis un onglet Profil\_\*)
  + {{ConseilCarriere}} (depuis un onglet Profil\_\*)
  + {{Q01:\_Quelle\_attitude\_}} (depuis la feuille de réponses)

**5.4. Structure des Données (Feuilles et Colonnes)**

* **Fichier** : [BDD]V2 Tests & Profils
* **Onglet** : sys\_Composition\_Emails
* **Colonne Critique** : Contenu / ID\_Document. Le contenu de cette colonne doit maintenant utiliser les placeholders dynamiques décrits ci-dessus.
* **Fichier** : [BDD]V2 Tests & Profils
* **Onglets** : Profils\_\*
* **Colonnes Critiques** : Les en-têtes de **toutes** les colonnes de ces onglets deviennent automatiquement des placeholders utilisables. Ils doivent être nommés sans espaces ni caractères spéciaux (l'underscore \_ est autorisé).

**6. Actions Recommandées / Prochaines Étapes**

* **Action Immédiate (Pour Vous)** :
  1. **Harmoniser sys\_Composition\_Emails** : Passez en revue toutes les lignes de cet onglet et mettez à jour les anciens placeholders (ex: {{profil\_titre}}) avec les nouveaux noms dynamiques basés sur vos en-têtes (ex: {{Titre\_Profil}}).
  2. **Valider le test "Ancres"** : Une fois l'harmonisation faite, lancez la procédure de test complète pour le test "Ancres" pour confirmer que le nouveau moteur fonctionne parfaitement.
* **Questions pour notre prochaine session** :
  1. L'harmonisation des placeholders a-t-elle posé des difficultés ?
  2. Le test "Ancres" a-t-il fonctionné comme prévu de bout en bout ?
  3. Sommes-nous prêts à démarrer le développement de l'intégration des tests "R&K" ?

Absolument. Voici une synthèse textuelle de nos échanges sur les évolutions du système d'envoi d'e-mails. Vous pourrez l'intégrer à votre document d'état du projet pour que nous puissions nous y référer lors de notre prochaine session.

**Synthèse des Évolutions Stratégiques pour la Gestion des E-mails**

Ce document résume les fonctionnalités à développer pour finaliser la logique de communication de l'application "Usine à Tests". L'objectif est de permettre un contrôle total et flexible sur le moment, le destinataire et le contenu des e-mails envoyés, afin de couvrir plusieurs cas d'usage métier identifiés.

**1. Cas d'Usage : Enquête d'Opinion / Collecte de Données**

* **Objectif** : Utiliser l'usine pour créer des questionnaires (enquêtes d'opinion, collecte d'informations) où le répondant ne doit recevoir aucun résultat. Le but est uniquement de stocker les données dans la feuille Google Sheets et, optionnellement, de notifier un gestionnaire de la nouvelle entrée.
* **Solution Identifiée** : La configuration actuelle permet déjà ce scénario. En laissant la colonne ID\_Gabarit\_Email\_Repondant vide et en utilisant une colonne de copie (ex: Email\_Formateur), le système n'enverra aucun e-mail au répondant mais notifiera le gestionnaire. Le déclencheur onFormSubmit reste actif pour la collecte.

**2. Évolution 1 : E-mail de Confirmation Immédiat**

* **Objectif** : Rassurer immédiatement le répondant en lui confirmant que ses réponses ont bien été enregistrées, surtout lorsque l'envoi des résultats est différé.
* **Implémentation Proposée** :
  + Créer une nouvelle colonne dans Paramètres Généraux : ID\_Gabarit\_Email\_Confirmation.
  + Si un ID de document Google Doc est renseigné dans cette colonne, le script enverra, dès la soumission, un e-mail simple et générique au répondant (ex: "Nous avons bien reçu vos réponses...").
  + Si la cellule est vide, aucune confirmation n'est envoyée.

**3. Évolution 2 : Envoi Différé des Résultats**

* **Objectif** : Permettre de programmer l'envoi de l'e-mail contenant les résultats détaillés après un certain délai (quelques heures, quelques jours).
* **Implémentation Proposée** :
  + Créer une nouvelle colonne : Delai\_Envoi\_Heures.
  + Lors de la soumission du formulaire, la fonction onFormSubmit lira cette valeur.
  + Si un délai > 0 est configuré, le script ne déclenchera pas l'envoi direct. À la place, il créera un **déclencheur programmé (trigger)** qui appellera une fonction d'envoi spécifique après le nombre d'heures défini, en lui passant le numéro de la ligne à traiter.
  + Une fois l'e-mail envoyé, le déclencheur sera automatiquement supprimé.

**4. Cas d'Usage Avancé : Le Scénario "Formateur"**

* **Objectif** : Gérer un flux de travail où le répondant reçoit une confirmation immédiate, mais où les **résultats détaillés sont envoyés exclusivement au formateur**. Ce dernier pourra alors les imprimer ou les débriefer en personne avant de décider de les transmettre au répondant.
* **Implémentation Proposée (combinaison des évolutions)** :
  + Créer deux nouvelles colonnes dans Paramètres Généraux :
    1. Mode\_Envoi\_Resultats : Une colonne qui pourra contenir la valeur Formateur\_Uniquement.
    2. Email\_Destinataire\_Resultats : L'adresse e-mail du formateur/gestionnaire.
  + Le flux de traitement dans le script sera le suivant :
    1. **Confirmation** : Le script vérifie la colonne ID\_Gabarit\_Email\_Confirmation et envoie l'accusé de réception au répondant si configuré.
    2. **Routage des Résultats** : Le script vérifie la colonne Mode\_Envoi\_Resultats. Si la valeur est Formateur\_Uniquement, l'e-mail de résultats complets est envoyé à l'adresse spécifiée dans Email\_Destinataire\_Resultats, et **jamais** au répondant.
    3. Si le mode est différent (ou vide), le système applique la logique standard (envoi immédiat ou différé au répondant).