

Annexe technique

Fonctionnement d'un système automatisé de contestation administrative

Plateforme Captain Radar — Description logique, non exhaustive et non opératoire

Version : v2.2 | Date : 23/01/2026

Objet du document. Décrire, en termes techniques et pédagogiques, le fonctionnement d'un service numérique automatisé qui (i) collecte des informations déclaratives via un questionnaire, (ii) extrait des données d'une contravention via reconnaissance de texte (OCR) et (iii) génère une lettre-type préétablie correspondant à un scénario de contestation. Le document ne constitue pas un mode d'emploi de reproduction : les détails d'implémentation (outils, prestataires, paramétrage, endpoints, schémas internes) sont volontairement omis car non nécessaires à la compréhension du mécanisme.

1. Contexte et objectif

Cette annexe décrit, d'un point de vue strictement technique, la chaîne automatisée permettant à un utilisateur de déposer une contestation administrative d'une contravention via une plateforme numérique. L'enjeu est de caractériser un mécanisme d'industrialisation de démarches : collecte de données, extraction de champs, sélection d'un scénario pré-défini, puis génération d'une lettre-type.

1.1 Principes fonctionnels (résumé factuel)

Le fonctionnement repose sur un nombre limité de scénarios prédéfinis. Le système vérifie, via les réponses au questionnaire, que l'utilisateur entre dans l'un de ces scénarios, puis génère la lettre-type associée en y insérant les données d'identité et de dossier (numéro d'avis, dates, etc.).

Points structurants (à retenir) :

- L'utilisateur renseigne un questionnaire fermé (choix de réponses, cases à cocher, champs d'identité). Ces réponses constituent les seules données « circonstancielles » utilisées.
- L'utilisateur transmet la copie de sa contravention. Le système réalise une reconnaissance de texte (OCR) et extrait des champs administratifs (numéro d'avis, date, plaque, code infraction, etc.).
- Un moteur de règles identifie un scénario parmi un nombre limité de cas conceptualisés ; chaque scénario est associé à une lettre-type préétablie.
- La génération consiste à compléter la lettre-type par fusion de champs (mail merge) : nom/prénom, adresse, numéro d'avis, éléments déclaratifs du questionnaire, sans création d'argumentaire sur mesure.
- Aucun « diagnostic juridique individualisé » n'est produit : le système n'interprète pas une situation au-delà de critères fermés et ne rédige pas d'acte personnalisé.

2. Définitions techniques (glossaire)

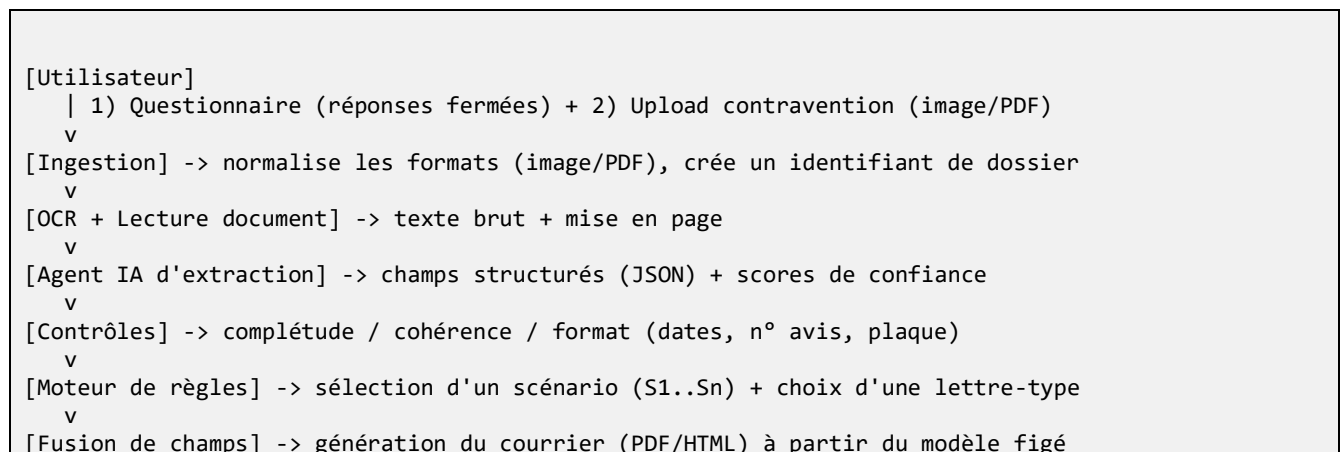
Terme	Définition
Automatisation	Exécution d'une suite d'actions par un système informatique, sans intervention humaine, à partir d'entrées déterminées (formulaire, document) et selon des règles pré-configurées.
IA (Intelligence artificielle)	Famille de méthodes statistiques qui apprennent des régularités à partir de données et produisent, en inférence, une sortie (classement, extraction, génération) à partir d'une entrée.
Agent IA	Composant logiciel qui combine perception (lecture de document), structuration (extraction en champs) et contrôle (validation/complétude) pour produire une sortie structurée à partir d'une entrée non structurée.
OCR (Optical Character Recognition)	Technique de reconnaissance de texte qui convertit une image (scan/photo) en texte exploitable, en

	détectant les zones de texte puis en reconnaissant les caractères.
Vision par ordinateur	Méthodes permettant d'analyser des images (détection de zones, redressement, classification de document, extraction d'éléments visuels).
Extraction structurée	Transformation d'un contenu non structuré (image, texte OCR) en un objet de données normalisé (ex. JSON) avec champs nommés et indicateurs de confiance.
Moteur de règles	Module qui applique des conditions déterministes (if/then) sur les réponses du questionnaire et/ou les champs extraits afin de sélectionner un scénario et une lettre-type.
Lettre-type	Modèle de courrier rédigé en amont, figé sur le fond, comportant des variables (placeholders) destinées à être remplacées par des données d'identité et de dossier.
Fusion de champs (mail merge)	Procédé consistant à remplacer automatiquement des placeholders d'un modèle par des valeurs issues d'un formulaire ou d'un document (ex. {{numero_avis}}).
Traçabilité	Journalisation d'événements techniques (horodatage, identifiant de dossier, étapes franchies, erreurs) permettant de reconstituer le traitement automatisé.

3. Vue d'ensemble de l'architecture logique

Le système peut se décrire comme une chaîne « Entrées → Normalisation → Décision → Génération ». Chaque étape transforme des données, sans interprétation libre :

3.1 Schéma fonctionnel (niveau logique)



v [Déclencheur administratif] -> validation de la demande et, le cas échéant, paiement
v [Remise à l'utilisateur / Dépôt] -> la contestation est déposée au nom du contrevenant

4. Entrées utilisateur : questionnaire, document et déclenchement

4.0 Parcours utilisateur (A-Z) — vision client

Le parcours de contestation (tunnel de vente) est un formulaire multi-étapes linéaire, déclenché depuis un bouton d'entrée (ex. « Contester une amende »). Les informations sont collectées via des questions fermées (principalement oui/non) et complétées par le dépôt de la contravention. Le présent document décrit ce parcours de constitution administrative du dossier ; tout simulateur d'information préalable, lorsqu'il existe, est hors périmètre et n'intervient pas dans la génération du courrier.

- Accès : l'utilisateur ouvre la page d'entrée et lance la démarche via un bouton d'action (CTA).
- Qualification initiale : l'utilisateur qualifie sa situation par sélection de réponses fermées (ex. boutons oui/non), sans champ libre.
- Questionnaire : l'utilisateur répond à des questions fermées (principalement oui/non), sans rédaction libre.
- Dépôt : l'utilisateur téléverse la contravention (photo/scan).
- Extraction : le système lit le document (OCR) et en extrait des champs administratifs.
- Lecture : le système tente d'extraire des informations administratives depuis la contravention. En cas d'extraction insuffisante (document illisible/incomplet), le dossier est classé techniquement non exploitable et suit un circuit administratif de remboursement ; le tunnel ne sollicite pas de correction/ré-upload au titre de l'argumentation.
- Génération : le système assemble un courrier à partir d'un modèle (lettre-type paramétrée) en injectant les champs collectés et en sélectionnant des paragraphes pré-écrits en fonction des réponses. Le client « constitue » ainsi son courrier par ses choix (cases/réponses), selon un assemblage déterministe.
- Sortie : le courrier est produit sous forme de document, prêt pour la transmission administrative selon le parcours proposé.

4.1 Questionnaire — données déclaratives (entrées contrôlées)

Le questionnaire du tunnel de contestation est volontairement standardisé : il collecte des variables factuelles via des questions fermées, afin de permettre un traitement administratif automatisé. Il ne sollicite pas un récit libre et ne requiert pas de rédaction par le client.

- Champs d'identité : état civil et coordonnées (utilisés uniquement en fusion de champs).
- Champs de qualification factuelle : réponses oui/non et informations factuelles (ex. nature du document, situation déclarée), utilisées comme variables d'assemblage.

4.1.1 Questionnaire linéaire : enchaînement fixe de questions fermées

Dans le tunnel de vente, le questionnaire suit un enchaînement linéaire de questions fermées. L'utilisateur progresse étape par étape, en choisissant des réponses (principalement oui/non) qui décrivent sa situation factuelle. Ce choix de conception vise à réduire l'ambiguïté, éviter la rédaction libre et garantir que le courrier produit résulte exclusivement de variables déclarées par l'utilisateur et d'un assemblage déterministe.

```
{
  "questions": [
    { "id": "points_loss", "type": "boolean", "required": true },
    { "id": "fixed_radar", "type": "boolean", "required": true },
    {
      "id": "company_denunciation",
      "type": "boolean",
      "required": true,
      "show_if": { "points_loss": true }
    },
    {
      "id": "identity_checked_by_officer",
      "type": "boolean",
      "required": true,
      "show_if": { "fixed_radar": false }
    },
    {
      "id": "driver_identity_known",
      "type": "boolean",
      "required": true,
      "show_if": { "points_loss": true }
    }
  ]
}
```

4.1.2 Contrôles techniques : normalisation et classement « exploitable / non exploitable »

Les informations collectées (déclaratives et extraites) peuvent faire l'objet de normalisations et contrôles techniques (format, longueur, caractères). Ces contrôles n'ont pas pour objet d'« améliorer » le dossier ni de construire un argumentaire : ils servent uniquement à déterminer si les éléments nécessaires à la constitution administrative du dossier sont exploitables. En cas d'illisibilité ou d'incomplétude technique, le dossier est classé « non exploitable » et suit un circuit administratif de remboursement ; le tunnel ne demande pas systématiquement de correction ou de re-dépôt.

- Exemple : un numéro d'avis incomplet/non exploitable conduit au classement « non exploitable » et à un remboursement administratif.
- Exemple : une immatriculation illisible n'est pas « corrigée » par le système : elle peut conduire au classement « non exploitable » et au remboursement.
- Exemple : si les réponses ne correspondent pas à un cas pris en charge, le système retourne un statut « non pris en charge » et n'émet pas de courrier.

Exemple d'entrée normalisée (JSON) :

```
{
  "case_id": "CR-2026-000812",
  "user_inputs": {
    "points_loss": true,
    "fixed_radar": true,
    "company_denunciation": false,
    "identity_checked_by_officer": false,
    "driver_identity_known": false
  },
  "identity": {
    "first_name": "Jean",
    "last_name": "DUPONT",
    "address": "12 rue Exemple, 75000 Paris",
  }
}
```

```
"email": "jean.dupont@example.com"
}
```

4.2 Dépôt de la contravention — document source

L'utilisateur transmet la copie de la contravention (photo ou scan). Le système traite le document comme une source de champs administratifs : numéro d'avis, dates, immatriculation, références, codes et libellés.

4.2.1 Dépôt du document : formats acceptés et lisibilité (sans ré-upload imposé)

Le téléversement est encadré par des contraintes de compatibilité (formats, taille). Le système traite le document comme une source d'informations administratives ; s'il ne peut pas lire suffisamment le document (qualité insuffisante, document tronqué, etc.), le dossier est classé « non exploitable » et suit un processus administratif de remboursement. Le parcours ne repose pas sur une sollicitation systématique de nouvelle transmission.

- Formats usuels acceptés : images (JPG/PNG/WebP) et PDF.
- Pré-traitement automatique : redressement/orientation/amélioration de contraste (sans altération du contenu), afin d'optimiser la lecture technique.

4.3 Déclencheur administratif (validation et, le cas échéant, paiement)

Dans une chaîne automatisée, certaines étapes peuvent être conditionnées par un évènement de validation administrative (ex. « dossier complet ») et, le cas échéant, par un évènement de transaction (ex. paiement confirmé). Techniquement, il s'agit d'un simple signal qui autorise l'exécution de la génération et/ou du dépôt.

Exemple d'évènement de déclenchement (illustratif) :

```
{ "ts": "2026-01-12T10:14:58Z", "case_id": "CR-2026-000812", "event": "PAYMENT_CONFIRMED", "amount": 39, "currency": "EUR" }
```

5. Traitement documentaire : OCR et lecture de contravention

5.1 De l'image au texte : ce que fait un OCR

Une contravention déposée en ligne est un fichier (image ou PDF). Pour un ordinateur, ce fichier est avant tout une grille de pixels (points colorés) ou une série de pages. Un OCR vise à convertir cette grille de pixels en texte exploitable.

Pipeline OCR typique (étapes successives) :

- Pré-traitement image : correction d'orientation, redressement de perspective, réduction du bruit, amélioration du contraste.
- Détection des zones de texte : localisation des blocs contenant des caractères (rectangles / polygones).
- Reconnaissance des caractères : conversion des pixels de chaque zone en séquences de caractères (A-Z, 0-9, ponctuation).

- Reconstruction : remise en ordre des lignes/colonnes, regroupement des mots, conservation approximative de la mise en page.
- Post-traitement : correction d'erreurs probables (0/O, 1/I), normalisation des formats (dates, plaques).

Exemple de sortie OCR (simplifiée) :

```
{
  "pages": [
    {
      "page_index": 0,
      "text": "AVIS DE CONTRAVENTION ... \nN° d'avis: 1234567890\nDate: 12/01/2026\nImmatriculation: AB-123-CD\n...",
      "blocks": [
        {
          "bbox": [95, 210, 420, 235],
          "text": "N° d'avis: 1234567890",
          "conf": 0.96,
        },
        {
          "bbox": [95, 245, 330, 270],
          "text": "Date: 12/01/2026",
          "conf": 0.94,
        }
      ]
    }
  ]
}
```

5.2 OCR vs IA : différence de rôle (pédagogique)

Pour simplifier : l'OCR « lit » et renvoie du texte. L'IA d'extraction « comprend la structure » et renvoie des champs. Autrement dit, l'OCR répond à « que dit le document ? », l'IA d'extraction répond à « quels champs administratifs en tirer ? ».

5.3 Ce qu'est une IA d'extraction (sans marque ni prestataire)

Une IA d'extraction est un modèle statistique utilisé en inférence : on lui fournit une entrée (image ou texte OCR) et on lui demande une sortie structurée. Dans un usage d'extraction, la sortie est contrainte par un schéma attendu (liste de champs) afin d'éviter une génération libre.

- Entrée : document + consignes de structuration (ex. « renvoyer un objet JSON avec notice_number, plate, dates »).
- Sortie : objet structuré ; chaque champ peut être accompagné d'un score de confiance.
- Encadrement : le système rejette toute sortie qui ne respecte pas le schéma ou les validations (formats).

Exemple de schéma attendu (illustratif) :

```
{
  "notice_number": "string",
  "notice_date": "date",
  "infraction_date": "date",
  "plate": "string",
  "infraction_code": "string|null",
  "fine_amount": "integer|null",
  "points_loss": "boolean|null"
}
```

6. Agent IA d'extraction : de la contravention aux champs (données structurées)

6.1 Extraction structurée et indicateurs de confiance

À partir du texte OCR (et éventuellement de la mise en page), l'agent IA produit des champs structurés et associe des indicateurs de fiabilité. Si des champs critiques sont insuffisamment fiables (ou absents), ils ne sont pas « inventés » : le dossier est classé techniquement non exploitable et suit un circuit administratif de remboursement. Le tunnel ne requiert pas, à ce stade, une validation/correction destinée à construire un argumentaire.

Exemple de sortie structurée (JSON) :

```
{
  "case_id": "CR-2026-000812",
  "document_fields": {
    "notice_number": {"value": "1234567890", "conf": 0.97},
    "notice_date": {"value": "2026-01-12", "conf": 0.93},
    "infraction_date": {"value": "2026-01-05", "conf": 0.91},
    "plate": {"value": "AB-123-CD", "conf": 0.95}
  },
  "quality": {
    "min_conf_threshold": 0.85,
    "low_conf_fields": []
  }
}
```

6.2 Contrôles de cohérence (validation automatique)

Avant d'alimenter la génération de courrier, le système exécute des validations mécaniques (formats, longueurs, contrôles de plausibilité).

Pseudo-code de validation (illustratif) :

```
def validate(fields):
    assert re.match(r"^\d{8,12}$", fields["notice_number"].value)
    assert re.match(r"^[A-Z]{2}-\d{3}-[A-Z]{2}$", normalize_plate(fields["plate"].value))
    return True
```

7. Sélection de scénario et génération de la lettre-type

7.1 Sélection de scénario par moteur de règles

Le moteur d'assemblage applique des règles déterministes sur les réponses du questionnaire et les champs extraits. Ces règles sont codifiées a priori (logique binaire/conditions) et ne produisent pas un raisonnement juridique : elles sélectionnent uniquement des blocs pré-écrits et des variables de fusion. La « sélection » provient des réponses de l'utilisateur, qui paramètrent mécaniquement l'assemblage.

Exemple de règles (illustratif) :

```
# Pseudo-code illustratif — sélection de cas (règles déterministes)
# Les variables proviennent exclusivement des réponses fermées de l'utilisateur (+ champs extraits).

if user.is_case_supported == False:
    scenario = "NON_PRIS_EN_CHARGE"
elif user.fixed_radar == True and user.identity_checked_by_officer == False:
    scenario = "CAS_A"
else:
    scenario = "CAS_B"
```

7.2 Fusion de champs : génération d'un courrier à partir d'un modèle

Le courrier est construit à partir d'un modèle unique (lettre-type paramétrée) et d'une bibliothèque de paragraphes pré-écrits. Le système remplace des placeholders par les données déclarées/extraites et active certains paragraphes selon les réponses de l'utilisateur. En pratique, le client « constitue » son courrier par ses choix (cases/réponses) : l'outil ne rédige pas au libre, n'ajoute pas d'arguments nouveaux et n'effectue pas de conseil individualisé.

Extrait de modèle (simplifié) :

```
Objet : Contestation de l'avis de contravention n° {{notice_number}}

Madame, Monsieur l'Officier du Ministère Public,

Je soussigné(e) {{full_name}}, demeurant {{address}}, conteste l'avis de contravention n°
{{notice_number}}
relatif à l'infraction relevée le {{infraction_date}}.

[CONTENU FIGÉ DU MODÈLE ASSOCIÉ AU SCÉNARIO {{scenario_id}}]
```

8. Traçabilité : journal d'exécution automatisée

Le traitement automatisé est traçable par une suite d'événements techniques horodatés (ingestion, OCR, extraction, sélection de scénario, génération, dépôt).

```
[
  {"ts": "2026-01-12T10:14:02Z", "case_id": "CR-2026-000812", "event": "FORM_SUBMITTED"},
  {"ts": "2026-01-12T10:14:35Z", "case_id": "CR-2026-000812", "event": "DOCUMENT_UPLOADED"},
  {"ts": "2026-01-12T10:14:39Z", "case_id": "CR-2026-000812", "event": "OCR_COMPLETED"},
  {"ts": "2026-01-12T10:14:41Z", "case_id": "CR-2026-000812", "event": "FIELDS_EXTRACTED"},
  {"ts": "2026-01-12T10:14:44Z", "case_id": "CR-2026-000812", "event": "SCENARIO_SELECTED", "scenario": "S_RADAR_FIXE_SANS_IDENTITE"},
  {"ts": "2026-01-12T10:14:58Z", "case_id": "CR-2026-000812", "event": "PAYMENT_CONFIRMED"},
  {"ts": "2026-01-12T10:15:12Z", "case_id": "CR-2026-000812", "event": "SUBMISSION_DONE"}
]
```

9. Délimitation fonctionnelle : automatisation administrative (pas de conseil individualisé)

9.1 Mécanismes présents

- Collecte de variables via questionnaire fermé.
- Lecture automatique d'un document par OCR et extraction de champs.
- Affectation à un scénario via règles déterministes.
- Génération d'un courrier par fusion de champs à partir d'un modèle figé.
- Dépôt de la contestation et suivi technique de statut.

9.2 Mécanismes absents (à éviter toute confusion)

- Pas d'analyse juridique individualisée (pas de raisonnement au cas par cas au-delà de critères fermés).
- Pas de rédaction sur mesure : le courrier résulte d'un modèle unique paramétré (placeholders + paragraphes pré-écrits), assemblé selon les réponses de l'utilisateur.
- Pas d'évaluation de « chances de succès » calculée sur les faits.
- Pas de discussion ou de négociation : la plateforme exécute une démarche administrative standardisée.

10. Annexes — exemples techniques supplémentaires

10.1 Mapping « scénario → lettre-type » (illustratif)

```
{
  "template_id": "TEMPLATE_MAIN",
  "clause_sets": {
    "CAS_A": ["P_01", "P_02", "P_07"],
    "CAS_B": ["P_01", "P_03", "P_08"],
    "NON_PRIS_EN_CHARGE": []
  }
}
```

10.2 Gestion d'échec technique (OCR incomplet) — classement non exploitable et remboursement

```
# Pseudo-code illustratif — traitement d'échec technique (sans correction utilisateur)
# Objectif : décider si le dossier est exploitable pour la constitution administrative.

if extraction.missing_critical_fields == True:
    return {"status": "NON_EXPLOITABLE", "action": "REFUND"}

if extraction.confidence_score < MIN_CONFIDENCE:
    return {"status": "NON_EXPLOITABLE", "action": "REFUND"}
```

```
# Sinon : dossier exploitable, poursuite de l'assemblage  
return {"status":"EXPLOITABLE","action":"CONTINUE"}
```